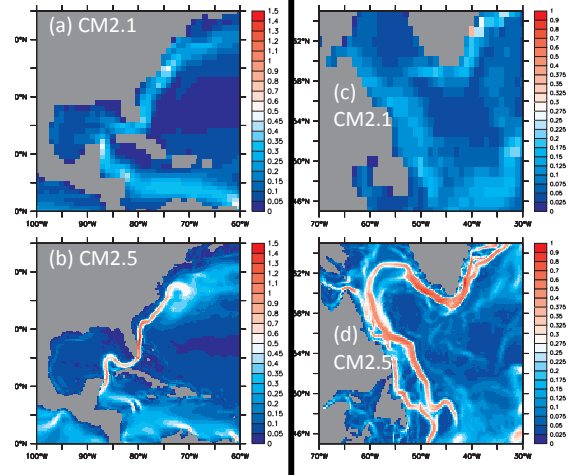
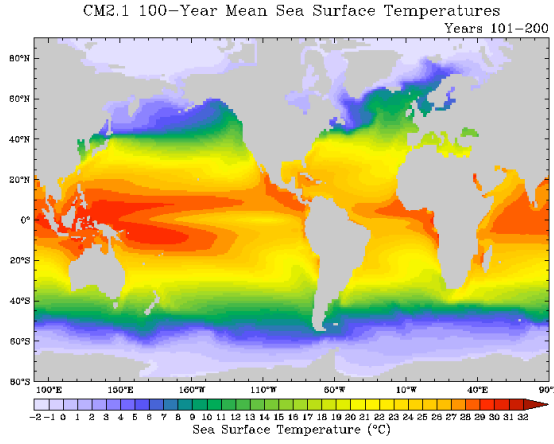


ELEMENTS OF THE MODULAR OCEAN MODEL (MOM)

2012 RELEASE WITH UPDATES

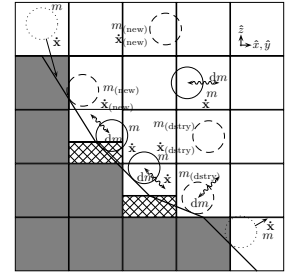
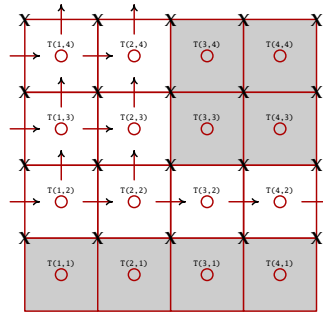
STEPHEN M. GRIFFIES
NOAA Geophysical Fluid Dynamics Laboratory
Princeton, USA

WITH CONTRIBUTIONS FROM
NOAA/GFDL AND PRINCETON UNIVERSITY SCIENTISTS AND ENGINEERS
AND THE INTERNATIONAL MOM COMMUNITY



$$\begin{aligned} [\partial_t + (f + M) \hat{\mathbf{z}} \wedge] (dz \rho \mathbf{u}) &= \rho dz S^{(u)} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] \\ &\quad - dz (\nabla_s p + \rho \nabla_s \Phi) + dz \rho \mathbf{F} \\ &\quad - [\rho (w^{(z)} \mathbf{u} - \kappa \mathbf{u}_z)]_{s=s_{k-1}} \\ &\quad + [\rho (w^{(z)} \mathbf{u} - \kappa \mathbf{u}_z)]_{s=s_k} \end{aligned}$$

$$\begin{aligned} \partial_t (dz \rho C) &= dz \rho S^{(C)} - \nabla_s \cdot [dz \rho (\mathbf{u} C + \mathbf{F})] \\ &\quad - [\rho (w^{(z)} C + F^{(s)})]_{s=s_{k-1}} \end{aligned}$$



This document is freely distributed. It provides information regarding the fundamentals and practices of the Modular Ocean Model. This document should be referenced as the following.

ELEMENTS OF THE MODULAR OCEAN MODEL (MOM) (2012 RELEASE WITH UPDATES)

GFDL OCEAN GROUP TECHNICAL REPORT No. 7

Stephen M. Griffies

NOAA/Geophysical Fluid Dynamics Laboratory

632 + xiii pages

Information about how to download and run MOM can be found at the GFDL Flexible Modeling System (FMS) web site accessible from www.gfdl.noaa.gov/fms.

This document was prepared using \LaTeX as described by [Lamport \(1994\)](#) and [Goossens et al. \(1994\)](#).

Cover images

- Sea surface temperature anomalies over years 101-200 from the coupled climate model CM2.1 documented in [Griffies et al. \(2005\)](#), [Gnanadesikan et al. \(2006\)](#), [Delworth et al. \(2006\)](#), [Wittenberg et al. \(2006\)](#), and [Stouffer et al. \(2006a\)](#). CM2.1 is provided as a test case with MOM.
- Annual mean surface current speed from the nominally one-degree ocean model used in CM2.1, and the nominally 1/4-degree ocean model used in CM2.5. This figure is taken after [Delworth et al. \(2012\)](#).
- Finite volume based momentum and tracer equations forming the basis for MOM, and as derived in Chapter 2.
- Schematic layout of tracer cells with fluxes crossing the cell faces. This layout is used for both the B-grid and C-grid arrangements. The B-grid is standard in MOM, and the C-grid is under development. See Chapters 9 and 14 for details.
- Schematic of the interactions between the Lagrangian blob submodel available in MOM and bottom topography, as may occur in an overflow situation ([Bates et al., 2012a,b](#)).

MOM AND THIS DOCUMENT

THE MODULAR OCEAN MODEL (MOM) IS A HYDROSTATIC GENERALIZED LEVEL COORDINATE NUMERICAL OCEAN CODE WITH MASS CONSERVING NON-BOUSSINESQ OR VOLUME CONSERVING BOUSSINESQ KINEMATICS. THE MODEL EQUATIONS ARE DISCRETIZED WITH GENERALIZED HORIZONTAL COORDINATES ON THE SPHERE USING EITHER AN ARAKAWA B-GRID OR C-GRID.¹ MOM HAS A BROAD SUITE OF PHYSICAL PARAMETERIZATIONS, DIAGNOSTIC FEATURES, TEST CASES, AND DOCUMENTATION. IT HAS BEEN UTILIZED FOR RESEARCH AND OPERATIONS FROM THE COASTS TO THE GLOBE. MOM IS INSTITUTIONALLY SANCTIONED BY NOAA'S GEOPHYSICAL FLUID DYNAMICS LABORATORY (GFDL), WHERE DEVELOPMENT IS CENTERED. ADDITIONAL DEVELOPMENT AND USE OCCURS THROUGH HUNDREDS OF INTERNATIONAL SCIENTISTS AND ENGINEERS COMPRISING THE MOM COMMUNITY. MOM IS FREE SOFTWARE DISTRIBUTED UNDER GPLv2 AND IT IS PART OF AN OPEN SOURCE COMMUNITY.

The 2014 release of MOM is the latest in a 50+ year history of numerical ocean codes developed at GFDL. In addition to GFDL leadership, MOM code development and use occurs through a broad network of scientists and engineers who contribute numerical algorithms, physical parameterizations, diagnostics, bug fixes, test cases, documentation, and user feedback. The development and use of MOM thus comprises a vital international open source community.

Ocean climate modeling has evolved tremendously over the years since Kirk Bryan first illustrated the compelling nature of a nonlinear wind driven ocean circulation in [Bryan \(1963\)](#) using a numerical model. We know far more about the ocean than in 1963, and we have far more realistic numerical tools to investigate the ocean using some of the most powerful computers on the planet. Furthermore, the problems associated with anthropogenic climate change prompt an increasing relevance and importance to the results produced by ocean models. That is, climate science is not limited to the domain of curiosity driven research. Instead, the science points to the nontrivial consequences of an ongoing uncontrolled planetary-wide experiment. Hence, there has never been a more critical time for numerical models to be fully detailed with rational and thorough descriptions, and supported by theory and observations.

Generations of ocean and climate scientists studied the ocean circulation by using the Cox code ([Cox, 1984](#)) in both idealized and realistic simulations. Cox's code formed the basis for the first version of MOM ([Pacanowski et al., 1991](#)) (see [Section 1.1](#) for a brief history of MOM). Over the years since [Cox \(1984\)](#), the name "MOM" has become synonymous with ocean climate models. The MOM release of 2014 is hence the result of decades of contributions by hundreds of scientists and engineers. Each contribution, however large or small, adds valuable experience and features that allows MOM to be a numerical tool worthy of the trust and utility required to make it suitable for both research and operations. A trustworthy and useful numerical tool is the result of robust numerical methods, a wide range of state-of-the-science physical parameterizations, and extensive diagnostics, combined with thorough and pedagogical documentation, a huge suite of proven applications, and decades of experience by generations of scientists and engineers. By this definition, MOM is among the world's most useful and valuable ocean codes.

This document provides an account of the theory and methods forming the fundamentals of MOM, with a focus on the most recent release, MOM5. Further documentation is available as part of the MOM distribution where details are given for how to configure the code for a particular model experiment. All of this documentation aims to strengthen the intellectual basis for MOM as well as its practical usability. It is with sincere humility and honor that I remain part of the MOM community, both as one interested in the science resulting from its simulations, and as one who nurtures and supports the science forming the foundations of the code itself. I hope that this document enables yet another generation of scientists and engineers, young and old, new and experienced, to wrap their heads around a truly significant piece of code, and in turn to offer feedback to support the integrity, transparency, utility, and evolution of MOM.

Stephen.Griffies@noaa.gov NOAA/GFDL, Princeton, USA

¹The C-grid version of MOM5 is new as of June 2012, and not yet available for general use. It is thus anticipated that the C-grid option in MOM will mature rapidly over 2012 and beyond, and become the standard choice, particularly for coastal and mesoscale eddying applications.

Contents

1	INTRODUCING THE MODULAR OCEAN MODEL	1
1.1	A brief history of MOM	2
1.2	Releases of MOM since 2003	4
1.3	The MOM6 Project	5
1.4	Elements of MOM5	5
1.5	A flow diagram for the MOM algorithm	18
1.6	Papers and reports providing documentation of MOM	21
1.7	Remainder of this document	22
I	Formulation of the ocean equations	23
2	FUNDAMENTAL EQUATIONS	25
2.1	Fluid kinematics	26
2.2	Mass conservation and the tracer equation	37
2.3	Thermodynamical tracers	39
2.4	Material time changes over finite regions	42
2.5	Basics of the finite volume method	44
2.6	Mass and tracer budgets over finite regions	45
2.7	Special considerations for tracers	51
2.8	Forces from pressure	53
2.9	Linear momentum budget	58
2.10	The Boussinesq budgets	62
3	THE HYDROSTATIC PRESSURE FORCE	65
3.1	Hydrostatic pressure forces at a point	65
3.2	Pressure gradient body force	66
3.3	Pressure gradient body force in B-grid MOM	71
3.4	Pressure gradient body force in C-grid MOM	74
4	PARAMETERIZATIONS WITH GENERALIZED LEVEL COORDINATES	75
4.1	Friction	75
4.2	Diffusion and skew diffusion	79
5	DEPTH AND PRESSURE BASED VERTICAL COORDINATES	85
5.1	Depth based vertical coordinates	85
5.2	Pressure based coordinates	92

6	EQUATION OF STATE AND RELATED QUANTITIES	97
6.1	Introduction	97
6.2	Linear equation of state	99
6.3	The two realistic equations of state	99
7	DYNAMICAL OCEAN EQUATIONS WITH A NONCONSTANT GRAVITY FIELD	103
7.1	Gravitational force: conventional approach	103
7.2	Gravitational force: general approach	105
8	TIDAL FORCING FROM THE MOON AND SUN	109
8.1	Tidal constituents and tidal forcing	109
8.2	Formulation in non-Boussinesq models	110
8.3	Implementation in MOM	110
II	Numerical formulations	113
9	B AND C GRID DISCRETIZATIONS	115
9.1	B and C grids used in MOM	115
9.2	Describing the horizontal grid	117
9.3	The Murray (1996) tripolar grid	121
9.4	Specifying fields and grid distances within halos	123
10	QUASI-EULERIAN ALGORITHMS FOR HYDROSTATIC MODELS	135
10.1	Pressure and geopotential at tracer points	136
10.2	Initialization issues	139
10.3	Vertical dimensions of grid cells	139
10.4	Summary of vertical grid cell increments	141
10.5	Surface height and bottom pressure diagnosed	147
10.6	Vertically integrated volume/mass budgets	148
10.7	Compatibility between tracer and mass	150
10.8	Diagnosing the dia-surface velocity component	150
10.9	Vertically integrated horizontal momentum	156
11	TIME STEPPING SCHEMES	159
11.1	Split between fast and slow motions	160
11.2	Time stepping the model equations as in MOM4.0	160
11.3	Introduction to time stepping in MOM	166
11.4	Basics of staggered time stepping in Boussinesq MOM	167
11.5	Predictor-corrector for the barotropic system	167
11.6	The Griffies (2004) scheme	169
11.7	Algorithms motivated from predictor-corrector	169
11.8	Algorithms enforcing compatibility	174
12	DISCRETE SPACE-TIME CORIOLIS FORCE	177
12.1	The Coriolis force and inertial oscillations	177
12.2	Time stepping for the B-grid version of MOM	179
12.3	Time stepping for the C-grid version of MOM	182
13	TIME-IMPLICIT TREATMENT OF VERTICAL MIXING AND BOTTOM DRAG	183
13.1	General form of discrete vertical diffusion	184
13.2	Discretization of vertical fluxes	184
13.3	A generic form: Part A	185
13.4	A generic form with implicit bottom drag	186

14	MECHANICAL ENERGY CONVERSIONS AND ADVECTIVE MASS TRANSPORT	189
14.1	Basic considerations	190
14.2	Energetic conversions in the continuum	191
14.3	How we make use of energetic conversions	193
14.4	Thickness weighted volume and mass budgets	194
14.5	Thickness and mass per area for the momentum	195
14.6	B-grid Boussinesq pressure work conversions	196
14.7	C-grid Boussinesq pressure work conversions	203
14.8	B-grid non-Boussinesq pressure work conversions	206
14.9	C-grid non-Boussinesq pressure work conversions	210
14.10	Effective Coriolis force and mechanical energy	212
14.11	B-grid kinetic energy advection	214
14.12	C-grid kinetic energy advection	218
15	ADVECTION VELOCITY AND HORIZONTAL REMAPPING FOR THE B-GRID	221
15.1	General considerations	221
15.2	Remapping operators for horizontal fluxes	222
15.3	Remapping operator for vertical fluxes	224
15.4	Remapping error	225
15.5	Subtleties at the southern-most row	228
16	OPEN BOUNDARY CONDITIONS FOR THE B-GRID	229
16.1	Introduction	230
16.2	Types of open boundary conditions	231
16.3	Implementation of sea level radiation conditions	234
16.4	OBC for tracers	238
16.5	The namelist obc_nml	241
16.6	Topography generation - Preparation of boundary data	243
III	Subgrid scale parameterizations for vertical processes	247
17	SURFACE AND PENETRATIVE SHORTWAVE HEATING	249
17.1	General considerations and model implementation	249
17.2	The Paulson and Simpson (1977) irradiance function	250
17.3	Shortwave penetration based on chlorophyll-a	251
17.4	Diagnosing shortwave heating in MOM	252
18	KPP FOR THE SURFACE OCEAN BOUNDARY LAYER (OBL)	255
18.1	Elements of the K-profile parameterization (KPP)	256
18.2	Surface ocean boundary momentum fluxes	261
18.3	Surface ocean boundary buoyancy fluxes	262
18.4	Surface layer and Monin-Obukhov similarity	268
18.5	Specifying the KPP parameterization	272
19	VERTICAL CONVECTIVE ADJUSTMENT SCHEMES	283
19.1	Introduction	283
19.2	Summary of the vertical adjustment options	283
19.3	Concerning a double application of vertical adjustment	284
19.4	Implicit vertical mixing	284
19.5	Convective adjustment	284

20 MIXING RELATED TO TIDAL ENERGY DISSIPATION	287
20.1 Formulation	287
20.2 Mixing from internal wave breaking	288
20.3 Dianeutral diffusivities from bottom drag	292
21 MIXING RELATED TO SPECIFIED MINIMUM DISSIPATION	295
21.1 Formulation	295
22 PARAMETERIZATION OF FORM DRAG	297
22.1 Regarding the TEM approach	297
22.2 What is available in MOM	298
 IV Subgrid scale parameterizations for lateral processes	 299
23 Neutral Physics	301
23.1 Introduction	303
23.2 Notation	306
23.3 Discretization	309
23.4 Implementation	315
23.5 Diffusion and Skew-Diffusion Tensors	320
23.6 Tracer Gradients	328
23.7 Quantities related to density gradients	329
23.8 Specification of the diffusivity	333
23.9 Summary of the notation	342
24 RESTRATIFICATION BY SUBMESOSCALE EDDIES	347
24.1 Basics of the scheme	347
24.2 Skew tracer flux components	349
24.3 Eddy induced transport	350
24.4 Eddy advection implementation	352
24.5 Cautionary remarks on <code>compute_psi_legacy</code>	352
24.6 Horizontal diffusion associated with submesoscale processes	353
25 LATERAL FRICTION METHODS	355
25.1 Introduction	356
25.2 Lateral friction options in MOM	356
25.3 Continuum formulation for the friction operator	357
25.4 Lateral friction operator for B-grid MOM	359
25.5 Lateral friction operator for C-grid MOM	367
25.6 Boundary conditions	371
26 EDDY-TOPOGRAPHY INTERACTION VIA NEPTUNE	373
26.1 Introduction	373
26.2 Basics of the parameterization in MOM	374
26.3 Topostrophy diagnostic	375
 V <i>Ad hoc</i> subgrid scale parameterizations	 377
27 OVERFLOW SCHEMES	379
27.1 Motivation for overflow schemes	380
27.2 The sigma transport scheme	380
27.3 The Campin and Goosse (1999) scheme	385
27.4 Neutral depth over extended horizontal columns	389

27.5	Sigma friction	391
28	RIVER DISCHARGE INTO THE OCEAN MODEL	393
28.1	Introduction	393
28.2	General considerations	394
28.3	Steps in the algorithm	395
29	CROSS-LAND MIXING	397
29.1	Introduction	397
29.2	Tracer and mass/volume compatibility	398
29.3	Tracer mixing in a Boussinesq fluid with fixed boxes	398
29.4	Mixing of mass/volume	399
29.5	Tracer and mass mixing	401
29.6	Formulation with multiple depths	402
29.7	Suppression of B-grid null mode	404
30	CROSS-LAND INSERTION	405
30.1	Introduction	405
30.2	Algorithm details	406
30.3	An example: insertion to three cells in MOM4.0	407
30.4	An example: insertion to just the top cell in MOM4.0	409
30.5	Updates for generalized level coordinates	410
31	THE B-GRID COMPUTATIONAL MODE	411
31.1	Checkerboard mode	411
31.2	Filter for sea surface height	412
31.3	Filter for bottom pressure	412
VI	Diagnostic capabilities	413
32	METHODS FOR DIAGNOSING MASS TRANSPORT	415
32.1	Brief on notation	416
32.2	Meridional-overturning streamfunction	416
32.3	Mass transport through tracer cell faces	421
32.4	Vertically integrated transport	425
33	KINETIC ENERGY DIAGNOSTICS	427
33.1	Formulation of kinetic energy diagnostics	427
34	EFFECTIVE DIANEUTRAL DIFFUSIVITY	429
34.1	Potential energy and APE in Boussinesq fluids	430
34.2	Effective dianeutral mixing	431
34.3	Modifications for time dependent cell thicknesses	434
34.4	An example with vertical density gradients	435
34.5	An example with vertical and horizontal gradients	440
35	SPURIOUS DISSIPATION FROM NUMERICAL ADVECTION	449
35.1	Formulation of the method for Boussinesq fluid	449
35.2	Formulation for MOM	451
35.3	Comparing to physical mixing	453

36	DIANEUTRAL TRANSPORT AND ASSOCIATED BUDGETS	455
36.1	Introduction to the diagnostic methods	458
36.2	Density layer mass budgets and watermass formation	459
36.3	Pieces required to locally compute dianeutral transport	464
36.4	The dianeutral transport	470
36.5	Layer calculation of the watermass transformation $\mathcal{G}(\gamma)$	472
36.6	Kinematic method to compute the material time derivative	475
36.7	Process method to compute the material time derivative	477
36.8	Finite volume estimate of the advective-form material time derivative	480
36.9	Comments on the MOM diagnostic calculation	489
36.10	Kinematic method diagnosed in MOM	493
36.11	Process method diagnosed in MOM	502
36.12	Budget for locally referenced potential density	525
36.13	Diagnosing mass budgets for density layers	526
36.14	Inferring transformation from surface buoyancy fluxes	534
36.15	Specifying the density classes for layer diagnostics	537
36.16	Known limitations	539
37	MIXED LAYER DEPTH DIAGNOSTICS	541
37.1	The mixed layer depth	541
37.2	Tracer budgets within the mixed layer	542
38	SUBDUCTION DIAGNOSTICS	543
38.1	Kinematics of flow across a surface	543
38.2	MOM subduction diagnostic calculation	548
39	DIAGNOSING THE CONTRIBUTIONS TO SEA LEVEL EVOLUTION	551
39.1	Mass conservation for seawater and tracers	553
39.2	Kinematic equations for sea level evolution	559
39.3	The non-Boussinesq steric effect	563
39.4	Evolution of global mean sea level	570
39.5	Vertical diffusion and global mean sea level	573
39.6	Neutral diffusion and global mean sea level	574
39.7	Parameterized quasi-Stokes transport and global mean sea level	577
39.8	MOM sea level diagnostics: Version I	580
39.9	MOM sea level diagnostics: Version II	592
40	GYRE AND OVERTURNING CONTRIBUTIONS TO TRACER TRANSPORT	595
40.1	Formulation	595
40.2	Enabling the diagnostic	596
41	BALANCING THE HYDROLOGICAL CYCLE IN OCEAN-ICE MODELS	599
41.1	Transfer of water between sea ice and ocean	599
41.2	Balancing the hydrological cycle	599
41.3	Water mass flux from salt mass flux	600
42	DIAGNOSING THE MOMENTUM BUDGET	603
42.1	A split-explicit algorithm to time step momentum	604
42.2	Momentum budget diagnostics	608
	BIBLIOGRAPHY	613

List of Figures

1.1	Bipolar Arctic grid lines	7
1.2	Bottom topography comparing full and partial cells	8
1.3	Comparing geopotential and z^* vertical coordinates	9
1.4	Schematic of Lagrangian blobs	12
1.5	Flow diagram for the MOM algorithm	19
2.1	Schematic of a generalized surface interior to the ocean	30
2.2	Schematic of the ocean bottom surface	33
2.3	Schematic of the ocean upper surface	36
2.4	Schematic of an ocean grid cell	47
2.5	Schematic of an ocean grid cell next to bottom	49
2.6	Schematic of an ocean grid cell next to ocean surface	50
2.7	Schematic of mass convergence-divergence	54
2.8	Schematic of pressure acting on a cube	54
2.9	Schematic of pressure in two dimensions	56
3.1	Illustrating grid cells in a vertical slice	72
4.1	Relation between slopes of surfaces	82
5.1	Comparison of partial step and full step topography	87
5.2	Constant depth surfaces with partial step	88
5.3	Constant sigma surfaces	91
7.1	Coordinates on a sphere	104
9.1	Placement of fields onto the B-grid	117
9.2	Placement of fields onto the C-grid	118
9.3	Four basic grid points for B and C grids	119
9.4	Grid cells with land-sea masking	120
9.5	Grid distances between points and vertices	122
9.6	Cell distances	123
9.7	Distances between grid points	124
9.8	Bipolar grid lines	125
9.9	Tracer and velocity cells on bipolar grid	125
9.10	North and east vectors on tracer cell faces within the bipolar grid	126
9.11	Basic elements of halos	127
9.12	Zonally periodic array	128
9.13	Quarter-cell distances at the bipolar fold	131

9.14 Tracer cell distances at the bipolar fold	133
9.15 Velocity cell distances at the bipolar fold	133
9.16 Grid distances between tracer points at the bipolar fold	134
10.1 Vertical column of tracer cells	138
11.1 Illustrating problems with leap-frog time stepping	165
14.1 Grid cells through a vertical slice with land-sea masking	195
14.2 Computation of discrete pressure	201
14.3 Tracer cell distances	203
14.4 Velocity cell distances	214
15.1 Schematic of the remapping function REMAP_ET_TO_EU	224
15.2 Tracer and velocity cell quarter distances	225
15.3 Tracer and velocity cell spacings	226
15.4 Tracer cell distances	226
15.5 Velocity cell distances	227
16.1 Schematic of open boundary conditions	231
18.1 KPP boundary layer schematic	259
18.2 Figure 1 from Large et al. (1994)	261
18.3 Figure 2 from Large et al. (1994)	274
18.4 Figure B1 from Large et al. (1994)	275
18.5 Alternative similarity functions	276
20.1 Problems with MOM4 implementation of Lee et al. (2006)	293
23.1 Stencils and indices for a centered triad group.	308
23.2 Stencils and indices for horizontal and vertical face centered triad groups.	309
23.3 The sine taper function.	321
23.4 the tanh taper function.	323
23.5 Diffusivity module namelist parameters and their relationships.	334
23.6 Illustrating the baroclinic zone for use in computing the diffusivity	341
25.1 Stencil for the discrete frictional functional	361
25.2 Notation for the quadrants surrounding a velocity point	364
25.3 Array of C-grid velocity vectors	368
25.4 Stencil for $u_{i,j}$ and $v_{i,j}$ contributions to lateral C-grid friction	369
25.5 C-grid layout for the deformation rates	372
27.1 Schematic of sigma transport pathways	381
27.2 Schematic of the Campin and Goosse overflow method	386
27.3 Specifying where a step occurs in the topography	388
27.4 Comparison of Campin and Goosse overflow method to Beckmann and Döscher	391
28.1 Schematic of river discharge algorithm	395
29.1 Schematic of cross-land mixing	399
30.1 Schematic of cross-land insertion	408
30.2 Example of cross-land insertion	410
32.1 Relating the overturning streamfunction to the transport	419

34.1 Sample vertical density profile	435
34.2 Vertical diffusive flux	436
34.3 Example of effective diffusivity	438
34.4 Example of effective diffusivity	439
34.5 Sorting a density profile	441
34.6 Vertical diffusive flux and sorted density	442
34.7 Vertical diffusive flux and sorted density	443
34.8 Sorting the density and the potential energy	445
34.9 Sorting the density field and the effective diffusivity	447
36.1 Mass balance for a density layer	460
36.2 Surfaces of constant generalized vertical coordinate	467
36.3 Schematic of an ocean grid cell	476
38.1 Surfaces of constant generalized vertical coordinate	544
39.1 Schematic of an ocean basin and the processes impacting sea level	561
39.2 Schematic ocean basin and the boundary and internal ocean processes impacting sea level	568
39.3 Schematic of how the Gent et al. (1995) scheme impacts sea level	579

INTRODUCING THE MODULAR OCEAN MODEL

Contents

1.1	A brief history of MOM	2
1.2	Releases of MOM since 2003	4
1.2.1	First release of MOM4.0: October 2003	4
1.2.2	First release of MOM4p1: Early 2007	4
1.2.3	MOM4p1 release December 2009	4
1.2.4	MOM5 release 2012 and afterward	4
1.3	The MOM6 Project	5
1.4	Elements of MOM5	5
1.4.1	FMS and parallel programming	5
1.4.2	Features of the dynamical core	6
1.4.2.1	Generalized orthogonal horizontal coordinates	6
1.4.2.2	Partial bottom steps	6
1.4.2.3	Generalized level coordinates	7
1.4.2.4	Explicit barotropic solver	10
1.4.2.5	Time stepping schemes	10
1.4.2.6	Pressure gradient calculation	11
1.4.3	Dynamically interacting Lagrangian parcels	11
1.4.4	Tracer features	11
1.4.4.1	Equation of state	11
1.4.4.2	Conservative temperature	12
1.4.4.3	Freezing temperature for frazil	12
1.4.4.4	Tracer advection	12
1.4.4.5	Tracer packages	13
1.4.5	Subgrid scale parameterizations	14
1.4.5.1	Penetration of shortwave radiation	14
1.4.5.2	Horizontal friction	15
1.4.5.3	Convective adjustment schemes	15
1.4.5.4	Neutral physics	15
1.4.5.5	Restratification effects from submesoscale eddies	16
1.4.5.6	Parameterization of form drag	16
1.4.5.7	Tidal mixing parameterizations	16
1.4.5.8	An array of vertical mixing schemes	16
1.4.5.9	Overflow schemes	17
1.4.6	Diagnostics and the FMS diagnostic manager	17

1.4.7	Open boundary conditions	17
1.4.8	Test cases	17
1.5	A flow diagram for the MOM algorithm	18
1.6	Papers and reports providing documentation of MOM	21
1.7	Remainder of this document	22

The Modular Ocean Model (MOM) is a numerical representation of the ocean's hydrostatic primitive equations employing either Boussinesq (volume conserving) or non-Boussinesq (mass conserving) kinematics. It is formulated using a quasi-Eulerian algorithm employing generalized level coordinate technology that facilitates the use of a suite of vertical coordinates. It is designed primarily as a tool for studying the ocean climate as well as regional and coastal phenomena. There is a wide array of subgrid scale parameterizations (SGS) available for use in a variety of global to coastal applications. An extensive suite of diagnostic capabilities allows the researcher to probe into mechanisms underlying simulation features. MOM is developed by an international team of ocean scientists and engineers participating in the MOM project, with the main algorithm development and software engineering provided by NOAA's Geophysical Fluid Dynamics Laboratory (GFDL) in Princeton, USA. The model is freely available under the GNU General Public License (<http://www.gnu.org/licenses/gpl.html>) and can be downloaded after registration at

<http://www.gfdl.noaa.gov/fms>

The purpose of this document is to present a rationalized account of the theory and practice of MOM as an ocean model tool for use in studying the ocean climate system. To achieve this purpose, this document incorporates salient features of the following MOM related documents:

- The MOM3 Manual of [Pacanowski and Griffies \(1999\)](#)
- Fundamentals of Ocean Climate Models by [Griffies \(2004\)](#)
- A Technical Guide to MOM4.0 by [Griffies et al. \(2004\)](#)

There are additional elements in this document that are unique to more recent versions of MOM.

Note that MOM encompasses a relatively large body of code. Besides the code directly related to the ocean model itself, there are allied codes required to support the use of MOM on various computational platforms, including parallel machines; codes required to perform input/output operations; codes for coupling to other component models, etc. The present document is concerned exclusively with that code associated with the ocean equations.

1.1 A brief history of MOM

The Modular Ocean Model evolved from numerical ocean models developed in the 1960's-1980's by Kirk Bryan and Mike Cox at GFDL. Most notably, the first internationally released and supported primitive equation ocean model was developed by Mike Cox ([Cox \(1984\)](#)). Although somewhat common today, it was actually quite revolutionary in 1984 to freely release, support, and document code for use in numerical ocean modeling. The Cox-code provided scientists worldwide with a powerful tool to investigate basic and applied questions about the ocean and its interactions with other components of the climate system. Previously, rational investigations of such questions focused on idealized models and analytical methods. Many researchers embraced the Cox-code, thus fostering a wide community of users and developers that further enhanced the features and robustness of the code. This community approach has been fundamental to all versions of the Cox-code and subsequent releases of MOM, with the underlying assumption that the scientific integrity of the code progresses more rapidly through input from a wide suite of researchers employing the code for a variety of scientific and operational applications. Quite simply, the Cox-code started what has today become a right-of-passage for every high-end numerical model of dynamical earth systems.

Upon the untimely passing of Mike Cox in 1989 ([Bryan, 1991](#)), Ron Pacanowski, Keith Dixon, and Tony Rosati at GFDL rewrote the Cox-code with an eye on new ideas of modular programming using Fortran

77. The result was the first version of MOM (Pacanowski et al. (1991)). Version 2 of MOM (Pacanowski (1995)) introduced the memory window idea, which was a generalization of the vertical-longitudinal slab approach used in the Cox-code and MOM1. Both of these methods were driven by the desires of modelers to run large experiments on machines with relatively small memories. The memory window provided enhanced flexibility to incorporate higher order numerics, whereas slabs used in the Cox-code and MOM1 restricted the numerics to second order accuracy. MOM3 (Pacanowski and Griffies (1999)) even more fully exploited the memory window with a substantial number of new physics and numerics options.

MOM4 has origins dating back to a transition from vector to parallel computers at GFDL, starting in 1999. Other related codes successfully made the transition some years earlier (e.g., The Los Alamos Parallel Ocean Program (POP) and the OCCAM model from Southampton, UK). New computer architectures generally allow far more memory than previously available, thus removing many of the reasons for the slabs and memory window approaches used in earlier versions of MOM. Additionally, the loop structure can be quite opaque with the memory windows, making it relatively difficult to introduce new algorithms, especially for the novice. Hence, for MOM4.0, the memory window was jettisoned in favor of a horizontal 2D domain decomposition. The project to convert MOM3 to MOM4.0 took roughly four years of coding and testing.

After gaining some experience on parallel machines with MOM4.0, and after developing the IPCC AR4 coupled climate model CM2.1 at GFDL (Griffies et al., 2005; Delworth et al., 2006; Gnanadesikan et al., 2006), development focused on a generalized level coordinate version of MOM, allowing the code to be used with depth based Boussinesq vertical coordinates or pressure based non-Boussinesq vertical coordinates. This effort led to the MOM4p1 project. During development and use of MOM4p1, a wide suite of new diagnostics were developed in support of the evolving applications toward climate and biogeochemistry modeling. Additionally, MOM4p1 has incorporated tools required for use in regional and coastal applications (Herzfeld et al., 2011).

MOM4p1 continued to evolve from its initial release in 2007 toward the end of 2011. The most recent release took place in 2012, which represents the first release of MOM5. For many applications, the 2012 release of MOM is quite similar to the December 2009 release of MOM4p1. However, the 2012 MOM release has two notable enhancements to the underlying model framework.

- The 2012 MOM release has a C-grid layout for the horizontal gridding of the discrete model fields. The C-grid has many advantages for fine resolution models and for representing land/sea boundaries (see Section 9.1). Hence, there is much interest at GFDL and within the MOM community to allow MOM to support both the B-grid and C-grid. It is anticipated that the bulk of the fine resolution modeling with MOM at GFDL will transition from the B-grid to the C-grid during late 2012 and beyond. Note that the C-grid available in the initial release of MOM5 is a proto-type, with extensive testing remaining to be performed over the course of 2012 and beyond. Users intent on applying the C-grid for their purposes should recognize the early stages of this code.
- The 2012 MOM release is coupled to a dynamically active Lagrangian submodel as documented by Bates et al. (2012a,b). The interactive Lagrangian parcels provide a fundamentally new means to represent/parameterize vertical convection and gravity driven downslope processes. It is anticipated that much effort will be devoted over the next few years towards development and understanding of the utility of solving a coupled set of Eulerian and Lagrangian equations that interact through the exchange of mass, tracer, and momentum.
- Further work has continued to refine the many physical parameterizations in MOM.
- The 2012 MOM release has significantly new diagnostic facilities allowing researchers to probe mechanisms for water mass transformation and steric changes to sea level, amongst the growing suite of other diagnostic features.

The Cox-code and each version of MOM have an associated manual or user guide. Besides describing elements of the code and its practical use, these manuals aim to rationalize model methods, algorithms, and parameterizations. Absent such documentation, the code could present itself as a black box, thus greatly hindering its utility to the curious and skeptical scientific researcher. As the code grows and evolves, it is a nontrivial task to keep code and documentation consistent. Hence, visions for complete and updated

documentation are unrealized, with elements of the documentation incomplete and/or not fully consistent with the code. Nonetheless, the present document, as well as the earlier MOM documents, should provide ample opportunity to understand many details of the code, thus facilitating its use for simulating the ocean.

1.2 Releases of MOM since 2003

There have been many releases of MOM since the original MOM1 code in 1991. We focus here on the releases of MOM4.

1.2.1 First release of MOM4.0: October 2003

When physical scientists aim to rewrite code based on software engineering motivations, more than software issues are addressed. During the writing of MOM4, numerous algorithmic issues were also addressed, which added to the development time. Hence, the task of rewriting MOM3 into MOM4.0 took roughly four years to complete, taking place from 1999 to 2003. Such represents a very useful lesson. Namely, even if one presumes from the start that the code will be rewritten only with an eye towards computational architecture questions, such questions inevitably raise questions about fundamentals of algorithms and parameterizations. When introducing such additional questions, the timeline for rewriting code grows extensively. There is a general rule in code/model development that must be honestly acknowledged when scoping out the timelines for a project:

IT ALWAYS TAKES LONGER TO DEVELOP CODE AND MODEL CONFIGURATIONS THAN ORIGINALLY ANTICIPATED, EVEN WHEN UNDERSTANDING THAT IT TAKES LONGER THEN ANTICIPATED.

This rule has been proven valid multiple times with the development of MOM versions, and various model configurations, over its multiple decades of history.

1.2.2 First release of MOM4p1: Early 2007

Griffies spent much of 2005 in Hobart, Australia as a NOAA representative at the CSIRO Marine and Atmospheric Research Laboratory, as well as with researchers at the University of Tasmania. This period saw focused work to upgrade MOM4.0 to include certain features of generalized level coordinates. By allowing for the use of a suite of vertical coordinates, MOM4p1 is algorithmically more flexible than any previous version of MOM. This work, however, did not fundamentally alter the overall computational structure relative to the last release of MOM4.0 (the MOM4p0d release in May 2005). In particular, MOM4p1 is closer in “look and feel” to MOM4p0d than MOM4p0a is to MOM3.1. Given this similarity, it was decided to retain the MOM4 name for the MOM4p1 release, rather switch to MOM5.

1.2.3 MOM4p1 release December 2009

The MOM4p1 release of December 2009 represents a major upgrade to the code, especially those areas related to open boundary conditions of use for regional applications (Chapter 16 and Herzfeld et al. (2011)), various physical parameterizations, diagnostics, and computational infrastructure. This public release also provides the community with a *test case* consisting of the CM2.1 configuration used by GFDL for the IPCC AR4 assessment, as documented by Griffies et al. (2005), Gnanadesikan et al. (2006), Delworth et al. (2006), Wittenberg et al. (2006), and Stouffer et al. (2006a). Although CM2.1 for the AR4 assessment actually used MOM4.0, the setup in the CM2.1-MOM4p1 test case is backwards compatible, meaning that the climate state is the same.

1.2.4 MOM5 release 2012 and afterward

The most recent release of MOM occurred in 2012, and it is referred to as MOM5. This code is released via GitHub, with the source code continually updated. Hence, the user is able to access the most recent code without having to await bundling the code into a formal release.

As noted earlier, MOM5 includes a C-grid option as well as a dynamically interacting Lagrangian sub-model. It is notable that both the C-grid and Lagrangian submodel are less mature than other portions of MOM. Hence, extensive further tests and development are required. Therefore, we offer the Lagrangian code with the following caveat:

THE LAGRANGIAN BLOB SUBMODEL RELEASED WITH MOM REMAINS IN THE EARLY RESEARCH/DEVELOPMENT STAGE. IT HAS NOT YET BEEN PORTED TO THE C-GRID. FURTHERMORE, IT IS NOT FULLY SUPPORTED FOR PRODUCTION WORK.

THE C-GRID OPTION RELEASED WITH MOM IN 2012 REMAINS IN THE EARLY RESEARCH/DEVELOPMENT STAGE. IT HAS NOT YET BEEN FULLY TESTED.

1.3 The MOM6 Project

In addition to MOM, GFDL has supported the development of a generalized layer ocean model under the leadership of Bob Hallberg and Alistair Adcroft. This project has been termed GOLD, for Generalized Ocean Layer Dynamics. For certain applications, the choice for vertical coordinate needs to be more general than that available with the generalized level capability of MOM5 and earlier. In particular, the questions of spurious diapycnal mixing, first identified by Griffies et al. (2000b) and more recently summarized by Ilicak et al. (2012), motivated much of the GOLD effort, as did difficulties representing gravity driven downslope flows (Winton et al., 1998). GOLD has matured recently through development of an IPCC class earth system model using isopycnal vertical coordinates (the ESM2G model documented in Dunne et al. (2012, 2013)). Arguably GOLD represents the state-of-the-science in isopycnal layer models, and its dynamical core provides the framework for doing any vertical coordinate or hybrid coordinate.

Starting June 2012, Adcroft, Griffies and Hallberg have embarked on a major effort to merge key physical parameterizations and the dynamical core from GOLD into MOM. This project, known as MOM6, is timely for many reasons. The key motivator is that GFDL is initiating development of a new climate model, CM4, that includes a mesoscale eddy permitting ocean configuration. To ensure success of CM4, GFDL is focusing its presently diverse climate model development pathways. Thus, all the ocean model developers at GFDL will focus on a single ocean code trunk for use in the new climate model, as well as for other applications.

Progress towards MOM6 will occur in stages. Much of the initial efforts involve an upgrade to the computational framework in GOLD to facilitate merging in elements of MOM5's physical parameterizations and diagnostics. The associated code restructuring is relevant regardless of the merger with GOLD, given the need for MOM6 to address elements of the changing paradigm in computational platforms appearing on the near-term horizon. Throughout development towards MOM6, we will continue to support key capabilities of MOM5 as well as GOLD. At strategic points in this development, we will solicit input from the MOM community to examine the code and to provide assistance in upgrading selected portions where non-GFDL expertise is required. There will be frequent updates to the MOM community as development progresses towards MOM6. At GFDL, we are incredibly excited about the prospects of working towards a unified ocean model code base. We trust the community will be also be excited when the project matures.

1.4 Elements of MOM5

In this section, we outline certain features of MOM as of 2012. Note that much of the following discussion holds also for MOM4.0 and earlier releases of MOM4p1.

1.4.1 FMS and parallel programming

The tools required for parallel programming with MOM are provided by the GFDL Flexible Modeling System (FMS). FMS provides the foundation upon which MOM is coded. That is, MOM is based on FMS. There are dozens of scientists and engineers at GFDL focused on meeting the evolving needs of climate

scientists pushing the envelope of computational tools for studying climate. This situation is favorable to the oceanographer who is less interested in the computer science required to run a high-end model, and more interested in coding his or her new idea into MOM in a manner that is clear, flexible, and robust across various computational platforms.

One of the early decisions made towards porting MOM3 to MOM4.0 concerned the elimination of the memory window in MOM3 and MOM2. Instead, MOM4.0 and later releases employ arrays ordered (i, j, k) for straightforward processor domain decomposition over the horizontal (i, j) directions. This array layout provides the orientation of data structures used to parallelize MOM and other codes based on GFDL FMS.

For those unfamiliar with parallel programming, yet wish to code something new in MOM, it is recommended that study be placed on certain of the existing MOM modules. By doing so, one can garner a working understanding of the methods used to pass data across processor boundaries, thus ensuring that simulation results are independent of the details of processor layout.

1.4.2 Features of the dynamical core

This section outlines certain features of the dynamical core in MOM.

1.4.2.1 Generalized orthogonal horizontal coordinates

MOM4.0 and later releases are written using generalized horizontal coordinates, with the coordinates assumed to be locally orthogonal. The formulation in this document follows this approach as well. For global ocean climate modeling, MOM comes with test cases using the tripolar grid of [Murray \(1996\)](#).

Code for reading in the grid and defining MOM specific grid factors is found in the module

`ocean_core/ocean_grids.`

MOM comes with preprocessing code suitable for generating grid specification files of various complexity, including the [Murray \(1996\)](#) tripolar grid that has a bipolar Arctic region (see Figure 1.1). Note that the horizontal grid in MOM is static (time independent), whereas the vertical grid is generally time dependent. Hence, there is utility in separating the horizontal from the vertical grids.

1.4.2.2 Partial bottom steps

MOM4.0 and later releases employ the partial bottom step technology of [Pacanowski and Gnanadesikan \(1998\)](#) to facilitate the representation of bottom topography. Each of the generalized level coordinates in MOM make use of this technology. Code associated with partial bottom steps is located in the module

`ocean_core/ocean_topog.`

It is common in older (those dating from before 1997) z -models for model grid cells at a given discrete level to have the same thickness. In these models, it is difficult to resolve weak topographic slopes without including uncommonly fine vertical and horizontal resolution. This limitation can have important impacts on the model's ability to represent topographically influenced advective and wave processes. The partial step methods of [Adcroft et al. \(1997\)](#) and [Pacanowski and Gnanadesikan \(1998\)](#) have greatly remedied this problem via the implementation of more realistic representations of the solid earth lower boundary. Here, the vertical thickness of a grid cell at a particular discrete level does not need to be the same. This added freedom allows for a smoother, and more realistic, representation of topography by adjusting the bottom grid cell thickness to more faithfully contour the topography. Figure 1.2 illustrates the bottom realized with the ocean component of CM2.1, CM3, and ESM2M along the equator. Also shown is a representation using an older *full step* method with the same horizontal and vertical resolution. The most visible differences between full step and partial step topography are in regions where the topographic slope is not large, whereas the differences are minor in steeply sloping regions.

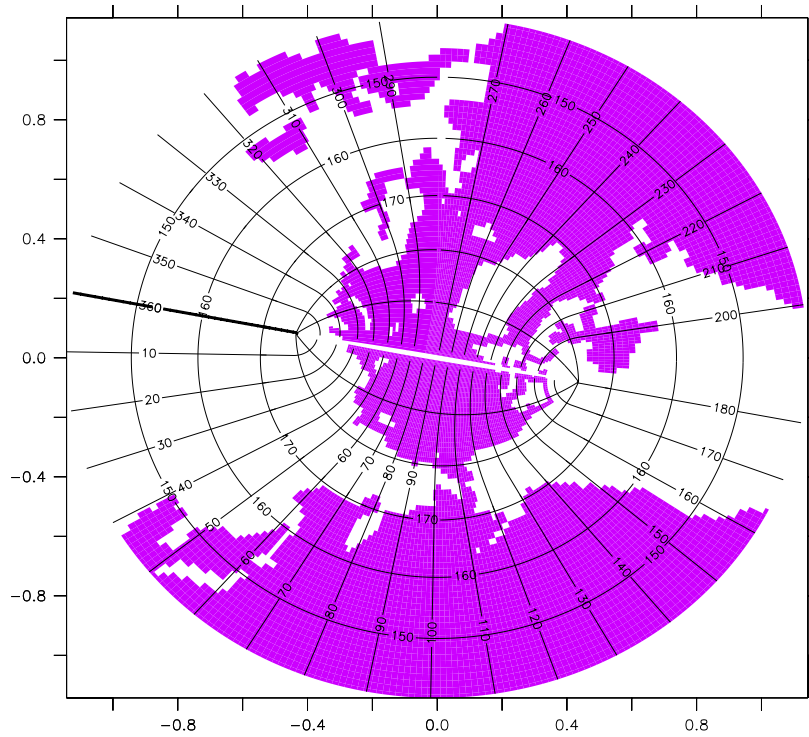


Figure 1.1: Illustration of the bipolar Arctic as prescribed by Murray (1996) (see his Figure 7) and realized in the ocean component of CM2.1, CM3, and ESM2M. The transition from the bipolar Arctic to the spherical grid occurs at 65°N . We denote horizontal grid cells by (i, j) indices. As in the spherical coordinate region of the grid, lines of constant i -index move in a generalized eastward direction within the bipolar region. They start from the bipolar south pole at $i = 0$, which is identified with $i = ni$, where ni is the number of points along a latitude circle and $ni = 360$ for a one degree horizontal resolution. The bipolar north pole is at $i = ni/2$, which necessitates that ni be an even number. Both poles are centered at a velocity point when using the B-grid in MOM. Lines of constant j move in a generalized northward direction. The bipolar prime-meridian is situated along the j -line with $j = nj$, where $nj = 200$ in OM3. This line defines the *bipolar fold* that bisects the tracer grid. Care must be exercised when mapping fields across this fold. As noted by Griffies et al. (2004), maintaining the exact identity of fields computed redundantly along the fold is essential for model stability. Note that the cut across the bipolar fold is a limitation of the graphics package, and does not represent a land-sea boundary in the model domain. This figure is taken after Figure 1 of Griffies et al. (2005).

1.4.2.3 Generalized level coordinates

Various vertical coordinates have been implemented in MOM. We have focused attention on vertical coordinates based on functions of depth or pressure, which means in particular that MOM *does not* support thermodynamic or isopycnal based vertical coordinates.¹

The following list summarizes vertical coordinates presently implemented in MOM. Extensions to other vertical coordinates are straightforward, given the framework available for the coordinates already present. Full details of the vertical coordinates are provided in Chapter 5.

- Geopotential coordinate as in MOM4.0, including the undulating free surface at $z = \eta$ and bottom partial cells approximating the bottom topography at $z = -H$

$$s = z. \quad (1.1)$$

¹The Hallberg Isopycnal Model (HIM) is available from GFDL for those wishing to use layered models and it is available at <http://www.gfdl.noaa.gov/fms/>.

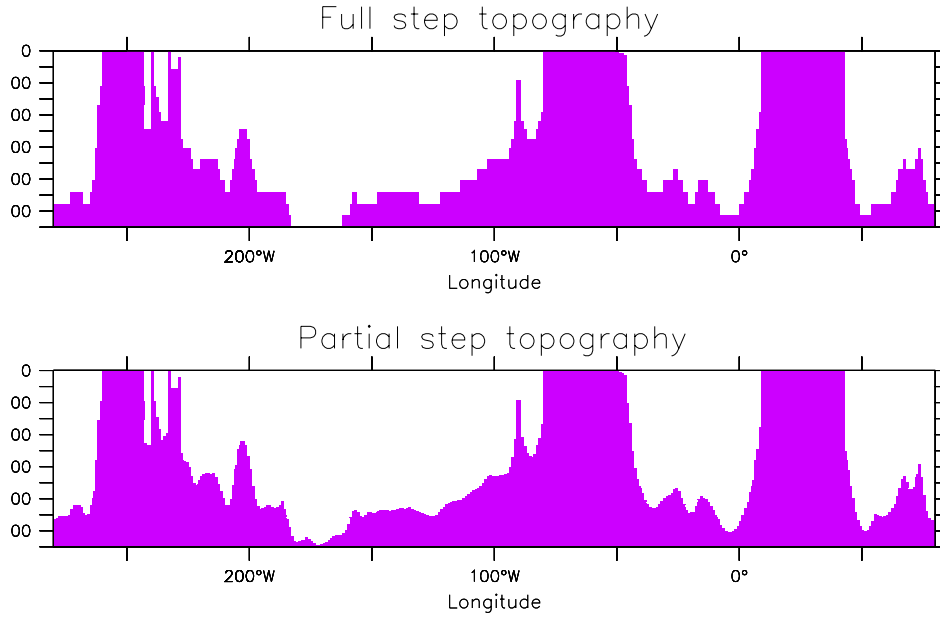


Figure 1.2: Bottom topography along the equator for the tracer cells. This figure illustrates the difference between the older full step representation of the bottom topography (upper) and the partial step representation used in CM2.1, CM3, and ESM2M (lower). Note the large differences especially in regions where the topographic slope is modest and small. This figure is taken after Figure 4 of [Griffies et al. \(2005\)](#).

This is the vertical coordinate used in the GFDL IPCC AR4 coupled climate model CM2.1 documented by [Griffies et al. \(2005\)](#); [Delworth et al. \(2006\)](#); [Gnanadesikan et al. \(2006\)](#).

- Quasi-horizontal rescaled height coordinate of [Stacey et al. \(1995\)](#) and [Adcroft and Campin \(2004\)](#)

$$\begin{aligned}
 s &= z^* \\
 &= H \left(\frac{z - \eta}{H + \eta} \right).
 \end{aligned}
 \tag{1.2}$$

This is the vertical coordinate used in the ocean component of the GFDL IPCC AR5 coupled climate model CM3 documented by [Griffies et al. \(2011\)](#) and [Donner et al. \(2011\)](#). It is also the vertical coordinate used in the earth system model ESM2M documented by [Dunne et al. \(2012, 2013\)](#). Note that tests at GFDL indicate that CM2.1 with the z^* vertical coordinate exhibits the same climate as CM2.1 with geopotential vertical coordinate.

In equation (1.2), $z = \eta(x, y, t)$ is the deviation of the ocean free surface from a state of rest at $z = 0$, and $z = -H(x, y)$ is the ocean bottom. Whereas a geopotential ocean model places all free surface undulations into the top model grid cell, a z^* model distributes the undulations throughout the ocean column. All grid cells thus have a time dependent thickness with z^* . Surfaces of constant z^* differ from geopotential surfaces according to the ratio η/H , which is generally quite small. Hence, surfaces of constant z^* are quasi-horizontal, thus minimizing difficulties of accurately computing the horizontal pressure gradient (see [Griffies et al., 2000a](#), for a review). The z^* vertical coordinate is analogous to the “eta” coordinate sometimes used for atmospheric models ([Black, 1994](#)).

We chose z^* for CM3 and ESM2M because of the enhanced flexibility when considering two key applications of climate models. The first application concerns large surface height deviations associated with tides and/or increased loading from sea ice (e.g., a global cooling simulation). The z^* model allows for the free surface to fluctuate to values as large as the local ocean depth, $|\eta| < H$, whereas the geopotential model is subject to the more stringent constraint $|\eta| < \Delta z_1$, with Δz_1 the thickness of

the top grid cell with a resting ocean. The ocean models in CM2.1 and CM3 set a minimum depth to $H \geq 40\text{m}$, whereas $\Delta z_1 = 10\text{m}$ (note that there is no *wetting and drying* algorithm in MOM). This flexibility with z^* is further exploited if considering even finer vertical grid resolution. Figure 1.3 illustrates this flexibility.

The second application where z^* is useful concerns increased land ice melt that adds substantially to the sea level, as in the idealized studies of [Stouffer et al. \(2006b\)](#), [Kopp et al. \(2010\)](#), and [Yin et al. \(2010b\)](#). Placing all of the surface expansion into the top model grid cell, as with the free surface geopotential model, greatly coarsens the vertical grid resolution in this important portion of the ocean, whereas the z^* model does not suffer from this problem since the expansion is distributed throughout the column.

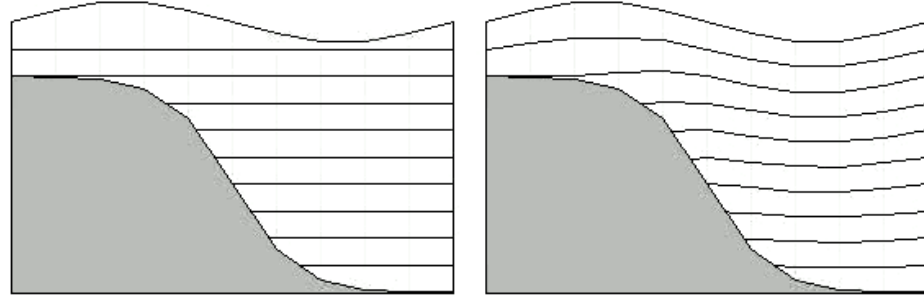


Figure 1.3: Illustrating the differences between geopotential vertical coordinate (left panel) and z^* vertical coordinate (right panel). In the upper ocean grid cell, the free surface with the geopotential vertical coordinate can generally penetrate through the bottom of the top cell lower boundary, in which case there is a problem with the simulation. In contrast, for the z^* vertical coordinate, all vertical cells undulate in time, with motion of the free surface spread throughout the ocean depth. Note that the undulations of the cell interfaces with z^* are scaled according to η/H , which is generally quite small. The undulations shown in this schematic are thus highly exaggerated for visualization purposes.

- Depth based terrain following “sigma” coordinate, popular for coastal applications (e.g., [Blumberg and Mellor, 1987](#))

$$\begin{aligned} s &= \sigma^{(z)} \\ &= \frac{z - \eta}{H + \eta}. \end{aligned} \quad (1.3)$$

This coordinate has not been for research applications by GFDL researchers.

- The pressure coordinate

$$s = p \quad (1.4)$$

was shown by [Huang et al. \(2001\)](#), [DeSzoeke and Samelson \(2002\)](#), [Marshall et al. \(2004\)](#), and [Losch et al. \(2004\)](#) to be a useful way to transform Boussinesq z -coordinate models into non-Boussinesq pressure coordinate models.

- Quasi-horizontal rescaled pressure coordinate

$$\begin{aligned} s &= p^* \\ &= p_b^o \left(\frac{p - p_a}{p_b - p_a} \right), \end{aligned} \quad (1.5)$$

where p_a is the pressure applied at the ocean surface from the atmosphere and/or sea ice, p_b is the hydrostatic pressure at the ocean bottom, and p_b^o is a time independent reference bottom pressure. This coordinate is the pressure coordinate analog to the z^* coordinate.

- Pressure based terrain following coordinate

$$s = \sigma^{(p)} = \left(\frac{p - p_a}{p_b - p_a} \right). \quad (1.6)$$

This coordinate is the pressure coordinate analog to the $\sigma^{(z)}$ coordinate.

We now highlight the following points regarding these vertical coordinates.

- All depth based vertical coordinates implement the volume conserving, Boussinesq, ocean primitive equations.
- All pressure based vertical coordinates implement the mass conserving, nonBoussinesq, ocean primitive equations.
- There has little effort focused on reducing pressure gradient errors in the terrain following coordinates (Section 3.2). Researchers intent on using terrain following coordinates may find it necessary to implement one of the more sophisticated pressure gradient algorithms available in the literature, such as that from [Shchepetkin and McWilliams \(2002\)](#).
- Use of neutral physics parameterizations (Section 4.2.3 and Chapter 23) with terrain following coordinates is not recommended with the present implementation. There are formulation issues that have not been addressed, since the main focus of neutral physics applications at GFDL centres on vertical coordinates that are quasi-horizontal.
- Most of the vertical coordinate dependent code is in the module

`ocean_core/ocean_thickness`

where the thickness of a grid cell is updated according to the vertical coordinate choice. The developer intent on introducing a new vertical coordinate may find it suitable to emulate the steps taken in this module for other vertical coordinates. The remainder of the model code is generally transparent to the specific choice of vertical coordinate, and such has facilitated a straightforward upgrade of the code from MOM4.0 to later releases.

- The restart file for `ocean_core/ocean_thickness` is not compatible across vertical coordinates, given particular distinctions between the various vertical coordinates. Hence, one should *not* modify the vertical coordinate in the middle of a simulation without re-initializing the thickness module.

1.4.2.4 Explicit barotropic solver

MOM4.0 and later releases employ a split-explicit time stepping scheme where fast two-dimensional dynamics is sub-cycled within the slower three dimensional dynamics. The method follows ideas detailed in Chapter 12 of [Griffies \(2004\)](#), which are based on [Killworth et al. \(1991\)](#) and [Griffies et al. \(2001\)](#). Chapter 10 presents the details for MOM, and the code is on the module

`ocean_core/ocean_barotropic.`

1.4.2.5 Time stepping schemes

The time tendency for tracer and baroclinic velocity can be discretized two ways.

1. The first approach uses the traditional leap-frog method for the inviscid/dissipationless portion of the dynamics, along with a Robert-Asselin time filter. This method is available in MOM4.0. However, its use is strongly discouraged given that it is unstable when used without time filters, and since the time filters preclude conservation of tracer.²

²The method from [Leclair and Madec \(2009\)](#) aims to overcome the limitations of tracer conservation with a time filtered leap frog scheme. Their method has not been implemented in MOM.

2. The preferred method discretizes the time tendency with a two-level forward step, which eliminates the need to time filter. Tracer and velocity are staggered in time, thus providing second order accuracy in time. For certain model configurations, this scheme has been found to be twice as efficient as the leap-frog based scheme since one can take twice the time step with the two-level approach. Furthermore, without the time filtering needed with the leap-frog, the new scheme conserves total tracer to within numerical roundoff. This scheme is discussed in [Griffies et al. \(2005\)](#) and [Griffies \(2004\)](#) (see Chapter 12), as well as in Chapter 10 of this document.

The code implementing these ideas in MOM can be found in

```
ocean_core/ocean_velocity
ocean_tracers/ocean_tracer
```

As discussed in Chapter 12, there are various methods available for time stepping the Coriolis force in MOM. The most commonly used method for global climate simulations with the B-grid version of MOM is the semi-implicit approach in which half the force is evaluated at the present time and half at the future time. An Adams-Bashforth scheme is used for the C-grid version of MOM.

1.4.2.6 Pressure gradient calculation

The pressure gradient calculation has been updated in MOM4p1 later releases to allow for the use of generalized vertical coordinates. A description of the formulation is given in Chapter 3, and the code is in the module

```
ocean_core/ocean_pressure.
```

Notably, none of the sophisticated methods described by [Shchepetkin and McWilliams \(2002\)](#) are implemented in MOM, and so terrain following vertical coordinates may suffer from unacceptably large pressure gradients errors. Researchers are advised to perform careful tests prior to using these coordinates.

1.4.3 Dynamically interacting Lagrangian parcels

The one feature that most distinguishes the 2012 release of MOM relative to earlier releases is the ability to enable an interactive Lagrangian parcel scheme, whereby the parcels, or “blobs”, are dynamically coupled to the traditional Eulerian grid cell properties. That is, the Lagrangian and Eulerian submodels conservatively exchange seawater mass, tracer mass, and momentum. Figure 1.4 provides a schematic of this coupled system.

There are two general physical applications that motivate considering the added degrees of freedom afforded with a Lagrangian submodel, with both applications associated with vertically unstable water.

- Representation of convection in a hydrostatic model;
- Representation of gravitationally driven bottom downslope flows.

[Bates et al. \(2012a,b\)](#) presents the formulation of how the Lagrangian submodel is coupled to the traditional Eulerian grid cells of MOM. The Lagrangian blobs comes with the following caveat:

THE LAGRANGIAN BLOB SUBMODEL RELEASED WITH MOM REMAINS IN THE EARLY RESEARCH/DEVELOPMENT STAGE.

1.4.4 Tracer features

In this section we outline features available for tracers in MOM.

1.4.4.1 Equation of state

As discussed in Chapter 6, the equation of state in MOM has been updated to TEOS-10 as detailed in [IOC et al. \(2010\)](#). The code for computing density and related fields is found in the module

```
ocean_core/ocean_density.
```

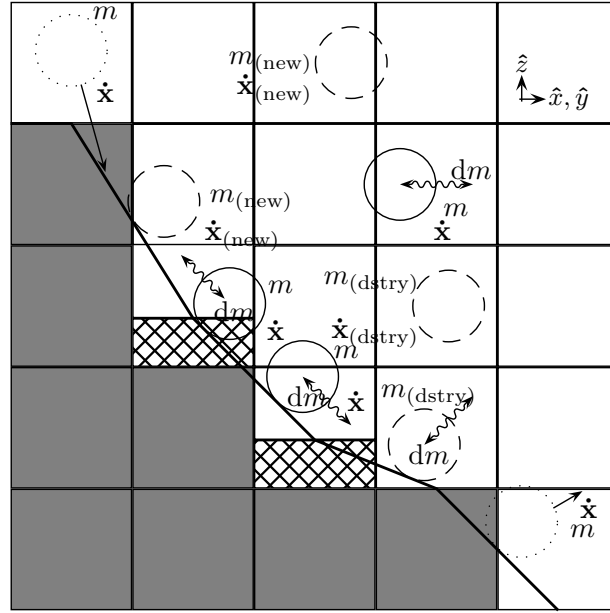


Figure 1.4: A vertical-horizontal section near the ocean bottom to illustrate Lagrangian parcels or blobs in MOM interacting with Eulerian grid cell properties. The cross-hatched region denotes partial step topography, and the entrainment and detrainment rates illustrate the decay and growth of a blob as it moves downslope with an acceleration $\dot{\mathbf{x}}$.

1.4.4.2 Conservative temperature

MOM time steps the conservative temperature described by [McDougall \(2003\)](#) to provide a measure of heat in the ocean (see Section 2.3.2). This variable is about 100 times more conservative than the traditional potential temperature variable. An option exists to set either conservative temperature or potential temperature prognostic, with the alternative temperature variable carried as a diagnostic tracer. This code for computing conservative temperature is within the module

`ocean_tracers/ocean_tempsalt.`

1.4.4.3 Freezing temperature for frazil

Accurate methods for computing the freezing temperature of seawater are provided by [Jackett et al. \(2006\)](#) and [IOC et al. \(2010\)](#). These methods allow, in particular, for the computation of the freezing point at arbitrary depth, which is important for ice shelf modelling. These methods have been incorporated into the frazil module

`ocean_tracers/ocean_frazil,`

with heating due to frazil formation treated as a diagnostic tracer.

1.4.4.4 Tracer advection

MOM comes with the following array of tracer advection schemes. Note that centred schemes are stable *only* for the leap-frog version of MOM. We thus partition the advection schemes according to the corresponding time stepping schemes. The code for tracer advection schemes are in the module

`ocean_tracers/ocean_tracer_advect.`

Examples of the advection scheme simulation features are provided in the Torus test case that comes with the MOM distribution.

- Tracer advection schemes available for either time stepping method include the following.
 1. First order upwind
 2. Quicker scheme is third order upwind biased and based on the [Leonard \(1979\)](#). [Holland et al. \(1998\)](#) and [Pacanowski and Griffies \(1999\)](#) discuss implementations in ocean climate models. This scheme does not have flux limiters, so it is not monotonic.
 3. Quicker-MOM3: The Quicker scheme in MOM4p1 differs slightly from that in MOM3, and so the MOM3 algorithm has also been ported to MOM4p1 and later releases.
 4. Multi-dimensional third order upwind biased approach of [Hundsdorfer and Trompert \(1994\)](#), with Super-B flux limiters.³ The scheme is available in MOM4p1 and later releases using either time stepping scheme.
 5. Multi-dimensional third order upwind biased approach of [Hundsdorfer and Trompert \(1994\)](#), with flux limiters of [Sweby \(1984\)](#).⁴ It is available in MOM4p1 and later releases with either time stepping scheme. This scheme was used in the ocean component of the CM2.1 climate model ([Griffies et al., 2005](#); [Delworth et al., 2006](#); [Gnanadesikan et al., 2006](#)).
 6. The second moment scheme of [Prather \(1986\)](#) has been implemented in MOM . It is available without limiters, or with the limiters of [Prather \(1986\)](#) and [Merryfield and Holloway \(2003\)](#).
 7. The multi-dimensional piece-wise parabolic method (MDPPM) has been implemented in MOM.⁵ This is the scheme used for most of the newer (post 2010) climate models developed at GFDL such as ESM2M ([Dunne et al., 2012, 2013](#)).

Both the Super-B and Sweby schemes are non-dispersive, preserve shapes in three dimensions, and preclude tracer concentrations from moving outside of their natural ranges in the case of a purely advective process. They are modestly more expensive than the Quicker scheme, and it do not significantly alter the simulation relative to Quicker in those regions where the flow is well resolved. The Sweby limiter code was used for the ocean climate model documented by [Griffies et al. \(2005\)](#). The MDPPM scheme can likewise ensure monotonicity with one of the three possible limiters.

- Tracer advection schemes available just for the leap-frog time stepping method include the following.
 1. Second order centred differences
 2. Fourth order centred differences: This scheme assumes the grid is uniformly spaced (in metres), and so is less than fourth order accurate when the grid is stretched, in either the horizontal or vertical.
 3. Sixth order centred differences: This scheme assumes the grid is uniformly spaced (in metres), and so is less than sixth order accurate when the grid is stretched, in either the horizontal or vertical. This scheme is experimental, and so *not* supported for general use.

1.4.4.5 Tracer packages

MOM comes with an array of tracer packages of use for understanding water mass properties and for building more sophisticated tracer capabilities, such as for ocean ecosystem models. Modules for these tracers are in the directories

```
ocean_tracers
ocean_bgc
ocean_shared/generic_tracers.
```

Various of the tracer options include the following.

³This scheme was ported to MOM4.0 by Alistair Adcroft, based on his implementation in the MITgcm. The online documentation of the MITgcm at <http://mitgcm.org> contains useful discussions and details about this advection scheme.

⁴This scheme was ported to MOM4.0 by Alistair Adcroft, based on his implementation in the MITgcm. The online documentation of the MITgcm at <http://mitgcm.org> contains useful discussions and details about this advection scheme.

⁵This scheme was ported to MOM4p1 by Alistair Adcroft, based on his implementation in the MITgcm. The online documentation of the MITgcm at <http://mitgcm.org> contains useful discussions and details about this advection scheme.

- Idealized passive tracer module with internally generated initial conditions. These tracers are ideal for testing various advection schemes, for example, as well as to diagnose pathways of transport.
- An ideal age tracer, with various options for specifying the initial and boundary conditions.
- The OCMIP2 protocol tracers (CO_2 , CFC, biotic).
- iBGC: A simple ocean biogeochemistry model.
- BLING: An intermediate complexity ocean biogeochemistry model. This model has been written in a *generic* format to allow for its use with both MOM and GFDL's model code GOLD. BLING is documented in the paper by [Galbraith et al. \(2011\)](#).
- TOPAZ: A comprehensive model of oceanic ecosystems and biogeochemical cycles is a state of the art model that considers 22 tracers including three phytoplankton groups, two forms of dissolved organic matter, heterotrophic biomass, and dissolved inorganic species for C , N , P , Si , Fe , CaCO_3 and O_2 cycling. The model includes such processes as gas exchange, atmospheric deposition, scavenging, N_2 fixation and water column and sediment denitrification, and runoff of C , N , Fe , O_2 , alkalinity and lithogenic material. The phytoplankton functional groups undergo co-limitation by light, nitrogen, phosphorus and iron with flexible physiology. Loss of phytoplankton is parameterized through the size-based relationship of [Dunne et al. \(2013\)](#). Particle export is described through size and temperature based detritus formation and mineral protection during sinking with a mechanistic, solubility-based representation alkalinity addition from rivers, CaCO_3 sedimentation and sediment preservation and dissolution. This model has been written in a *generic* format to allow for its use with both MOM and GFDL's isopycnal model GOLD. Further documentation of TOPAZ is provided by [Dunne et al. \(2013\)](#).

1.4.5 Subgrid scale parameterizations

Simulations in the ocean require the use of subgrid scale (SGS) parameterizations to allow the impacts from unresolved scales to impact the resolved scales. The development of robust and scientifically based SGS parameterizations is an active area of theoretical oceanography. Given the large uncertainty associated with parameterizations, MOM has chosen to implement a wide suite of methods so that researchers can have access to a variety of approaches that may best fit the particular application. The downside of such variety is that it requires knowledge by the user to best make use of the huge number of options. Some guidance for the use of SGS parameterizations is available from the test cases that come with MOM, and some is provided by querying the online MOM user community. Nonetheless, the best approach is for the MOM researcher to penetrate into the literature in order to make well educated decisions about SGS parameterizations for a particular application.

In this section we outline some features of the subgrid scale parameterizations available in MOM.

1.4.5.1 Penetration of shortwave radiation

Chapter 17 describes the computational method used in MOM for implementing the penetration of shortwave radiation into the ocean. The following modules are available for determining the details of how shortwave radiation penetrates into the ocean.

```
ocean_param/sources/ocean_shortwave
ocean_param/sources/ocean_shortwave_csiro
ocean_param/sources/ocean_shortwave_gfdl
ocean_param/sources/ocean_shortwave_jerlov
```

Please refer to each module for full documentation. In brief, these modules provide the following options.

- `ocean_shortwave`: This module drives the other shortwave modules.

- `ocean_shortwave_csiro`: This module implements an exponential decay for the penetrative short-wave radiation. This module was prepared at CSIRO Marine and Atmospheric Research in Australia.
- `ocean_shortwave_jerlov`: This module implements yet another exponential decay formulation (actually, a double exponential) for the penetrative shortwave radiation.
- `ocean_shortwave_gfdl`: This module implements the optical model of [Morel and Antoine \(1994\)](#) as well as that of [Manizza et al. \(2005\)](#).
 - [Sweeney et al. \(2005\)](#) compile a seasonal climatology of chlorophyll based on measurements from the NASA SeaWiFS satellite, and this climatology is available with the distribution of MOM. They used this data to develop two parameterizations of visible light absorption based on the optical models of [Morel and Antoine \(1994\)](#) and [Ohlmann \(2003\)](#). The two models yield quite similar results when used in global ocean-only simulations, with very small differences in heat transport and overturning.
 - The [Morel and Antoine \(1994\)](#) method for attenuating shortwave radiation was employed in the CM2 coupled climate model, as discussed by [Griffies et al. \(2005\)](#). MOM4p1 and later releases have updated the implementation of this algorithm relative to MOM4.0 by including the time dependent nature of the vertical position of a grid cell. The MOM4.0 implementation used the vertical position appropriate only for the case of a static ocean free surface.
 - In more recent model development, especially that associated with interactive biogeochemistry, GFDL modelers have preferred the scheme from [Manizza et al. \(2005\)](#) rather than [Morel and Antoine \(1994\)](#).

1.4.5.2 Horizontal friction

MOM has a suite of horizontal friction schemes, such as Smagorinsky laplacian and biharmonic schemes described in [Griffies and Hallberg \(2000\)](#) and the anisotropic laplacian scheme from [Large et al. \(2001\)](#) and [Smith and McWilliams \(2003\)](#). Code for these schemes is found in the modules

```
ocean_param/lateral/ocean_lapgen_friction
ocean_param/lateral/ocean_bihgen_friction.
```

1.4.5.3 Convective adjustment schemes

There are various convective methods available for producing a gravitationally stable column, with the code found in the module

```
ocean_param/vertical/ocean_convect.
```

The scheme used most frequently at GFDL for idealized modeling is that due to [Rahmstorf \(1993\)](#). Chapter 19 details this scheme and other adjustment methods. Note that for realistic climate and regional modeling, convective adjustment is not recommended. Instead, preference is given towards the use of a large vertical diffusivity, such as that promoted by [Klinger et al. \(1996\)](#). Consequently, the convective adjustment schemes remain in MOM largely for idealized simulations and legacy purposes.

1.4.5.4 Neutral physics

The parameterization of mesoscale eddies remains amongst the most active areas of theoretical research impacting ocean models. MOM comes with a suite of options available for treating neutral physics in the ocean interior as well as in boundary regions. A discussion of the methods is given in Chapter 23. The code related to this material is available in the directory

```
ocean_param/neutral.
```

1.4.5.5 Restratification effects from submesoscale eddies

There is an option available for parameterizing the restratification effects from submesoscale eddies, as proposed by [Fox-Kemper et al. \(2008b\)](#) and [Fox-Kemper et al. \(2011\)](#). The MOM formulation is given in Chapter 24, and the code is available in the module

`ocean_param/lateral/ocean_submesoscale.`

1.4.5.6 Parameterization of form drag

MOM has various options associated with the parameterization of form drag arising from unresolved mesoscale eddies, as proposed by [Greatbatch and Lamb \(1990\)](#), [Aiki et al. \(2004\)](#), and [Ferreira and Marshall \(2006\)](#). The code is available in the module

`ocean_param/vertical/ocean_form_drag,`

and documentation is given in Chapter 22. The form drag parameterization schemes have not been thoroughly used at GFDL.

1.4.5.7 Tidal mixing parameterizations

The tidal mixing parameterization of [Simmons et al. \(2004\)](#) has been implemented as a means to parameterize the diapycnal mixing effects from breaking internal gravity waves, especially those waves influenced by rough bottom topography. Additionally, this scheme has been combined with that used by [Lee et al. \(2006\)](#), who discuss the importance of barotropic tidal energy on shelves for dissipating energy and producing tracer mixing. Chapter 20 presents the model formulation, and

`ocean_param/vertical/ocean_vert_tidal`

contains the code.

1.4.5.8 An array of vertical mixing schemes

MOM comes with a wide array of vertical mixing schemes, including the following.

- Constant background diffusivity proposed by [Bryan and Lewis \(1979\)](#)

`ocean_param/vertical/ocean_vert_mix`

- Richardson number dependent scheme from [Pacanowski and Philander \(1981\)](#)

`ocean_param/vertical/ocean_vert_pp`

- The KPP scheme from [Large et al. \(1994\)](#)

`ocean_param/vertical/ocean_vert_kpp`

`ocean_param/vertical/ocean_vert_kpp_mom4p0`

The module `ocean_vert_kpp` maintains code provides some code updates relative to MOM4.0, such as to allow for the use of generalized vertical coordinates; features found useful in fresh inland seas; and modifications introduced by [Danabasoglu et al. \(2006\)](#). The module `ocean_vert_kpp_mom4p0` maintains code compatibility with the implementation of MOM4.0 necessary to allow for backwards compatibility with the CM2.1 coupled model documented in [Griffies et al. \(2005\)](#).

- GENERAL OCEAN TURBULENCE MODEL (GOTM): Coastal simulations require a suite of vertical mixing schemes beyond those available in most ocean climate models. GOTM ([Umlauf et al., 2005](#)) is a public domain Fortran90 free software used by a number of coastal ocean modellers

<http://www.gotm.net/>

GOTM includes many sophisticated turbulence closure schemes, and is updated periodically. It thus provides users of MOM4p1 and later releases access to most updated methods for computing vertical diffusivities and vertical viscosities. GOTM has been coupled to MOM by scientists at CSIRO in Australia in collaboration with German and GFDL scientists.

The MOM wrapper for GOTM is

```
ocean_param/vertical/ocean_vert_gotm
```

with the GOTM source code in the directory

```
ocean_param/gotm.
```

1.4.5.9 Overflow schemes

MOM comes with various methods of use for parameterizing, or at least facilitating the representation of, dense water moving into the abyss. These schemes are documented in Chapter 27, with the following modules implementing these methods

```
ocean_param/lateral/ocean_sigma_transport
ocean_param/lateral/ocean_mixdownslope
ocean_param/sources/ocean_overflow
ocean_param/sources/ocean_overexchange.
```

1.4.6 Diagnostics and the FMS diagnostic manager

MOM has traditionally had a plethora of diagnostic features. Indeed, perhaps one of the most appealing features of MOM is the exceptional range of online diagnostics available for help in understanding all aspects of the simulation, from details of the subgrid scale parameterizations to water mass transformation rates. A thorough discussion of various MOM diagnostic features is available in Part VI of this document.

The diagnostic manager is the central tool from the GFDL FMS code repository employed for writing diagnostics in MOM. The diagnostic manager allows users to decide at runtime whether to save a particular diagnostic field, and what particular space and time sampling to use. To access the diagnostic manager facility, the programmer must register a field to be sampled in the appropriate MOM location. This registration process is straightforward, with thousands of fields presently registered in MOM. But inevitably there will be a need to add a new diagnostic field. In this case, it is trivial to register this new field by emulating what has been done for other fields already in MOM. Furthermore, if the new field is deemed of use to the broader MOM community, then please suggest that it be included in future MOM releases.

1.4.7 Open boundary conditions

Much of the appeal of recent MOM releases is related to its enhanced facilities for regional ocean modeling, with Herzfeld et al. (2011) documenting a suite of tests that exercise these features. Central to this utility is the open boundary condition module

```
ocean_core/ocean_obc
```

which is documented in Chapter 16 as well as Herzfeld et al. (2011).

1.4.8 Test cases

All of the test cases have been revised as well as the addition of some new tests. These test cases are provided for computations and numerical evaluation, and as starting points for those wishing to design and implement their own research models.

1.5 A flow diagram for the MOM algorithm

This document aims to provide a full rationalization of how MOM updates the ocean state. Prior to delving into these extensive details, the reader may find it useful to see a coarse-grained perspective provided by Figure 1.5. Note that the particular order for the calculations are in some cases quite important, as they follow from the staggered time stepping methods detailed in Chapter 11. We now summarize the basic steps used in the MOM algorithm.

- Drive MOM using either the

`driver/ocean_solo.F90`

module when running MOM as an ocean-only model, or the module

`coupler/coupler_main.F90`

when coupling MOM to another component, such as sea ice or the atmosphere.

- Initialize ocean related fields for the start of a new time step, with these initialization calls coordinated by the module

`ocean_core/ocean_model.F90.`

Some of these fields are needed for prognostic calculations, and some are required for diagnostics.

- Accumulate surface and bottom boundary fluxes for use in forcing the ocean fluid. The surface fluxes are computed in

`ocean_core/ocean_sbc.F90`

and the bottom fluxes are computed in

`ocean_core/ocean_bbc.F90.`

- Compute the vertical mixing coefficients associated with subgrid scale (SGS) parameterizations. These coefficients are determined according to the chosen parameterization, and they are computed in modules contained in the directory

`ocean_param/.`

- Compute any sources or sponge data for tracer fields using modules in the directories

`ocean_tracers/`

`ocean_sources/.`

If there are biogeochemical tracers, then further sources will be computed in the biogeochemical modules contained in

`ocean_bgc/`

`ocean_shared/generic_tracers/.`

- Compute tracer tendencies associated with the first suite of subgrid scale parameterizations. These processes are computed using the ocean cell thicknesses as updated on the previous time step. Modules associated with this step are generally located in the directory

`ocean_param/.`

- Accumulate mass sources for use in updating the sea level or bottom pressure. This task is handled by code in

`ocean_core/ocean_barotropic.F90`

`ocean_core/ocean_thickness.F90.`

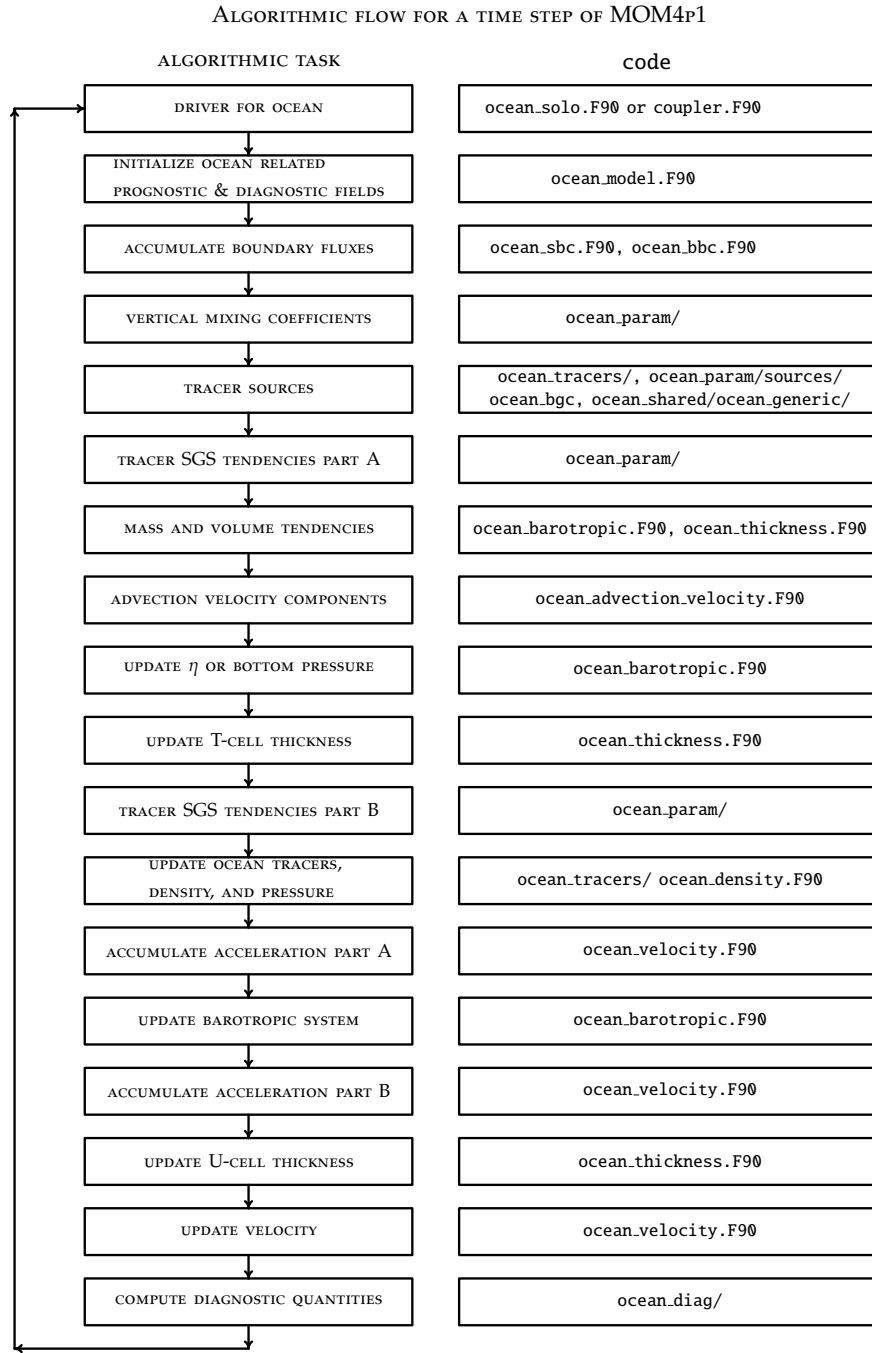


Figure 1.5: A flow diagram for the algorithm used in MOM to time step the ocean equations. Note that there are a few more steps required when enabling interactive Lagrangian blobs, with these additional steps detailed in [Bates et al. \(2012a,b\)](#).

- Diagnose the velocity components for use in tracer and velocity advection. The vertical component is diagnosed through the continuity equation, whereas the horizontal components are based on interpolating the velocity components to the tracer and velocity cell faces. This task is handled by code in

ocean_core/ocean_advection_velocity.F90.

- Update the sea surface height η (for Boussinesq depth based vertical coordinates) or bottom pressure (for non-Boussinesq pressure based vertical coordinates) to a new time step. This task is handled by code in

`ocean_core/ocean_barotropic.F90.`

- Update the vertical elements to the tracer cell using code in

`ocean_core/ocean_thickness.F90.`

- Compute tracer tendencies associated with the second suite of subgrid scale parameterizations. These processes are computed using the updated ocean cell thicknesses. Modules associated with this step are generally located in

`ocean_param/.`

- Update the tracer concentrations using tendencies associated with SGS processes, advection, and boundary fluxes. Perform the update first using time-explicit processes, and then update vertical and boundary processes using time-implicit methods. This task is handled by routines in

`ocean_tracers/ocean_tracer.F90.`

With the updated tracer concentrations, we then update the density in the module

`ocean_core/ocean_density.F90`

using the equation of state $\rho = \rho(S(\tau+1), \Theta(\tau+1), p(\tau))$, with the hydrostatic pressure computed from the previous time step. Pressure and density derivatives are then updated using all fields consistently at the τ time step.

- Accumulate the first portion of the acceleration, here associated with velocity advection, horizontal pressure gradients, horizontal friction, momentum sources, and parameterized form drag. Each of these accelerations are computed in a time-explicit manner, and they are coordinated by a routine in

`ocean_core/ocean_velocity.F90.`

- Update the two-dimensional vertically integrated momentum by time stepping the forced shallow water equations using routines in

`ocean_core/ocean_barotropic.F90.`

- Accumulate the second portion of the acceleration, here associated with time-explicit portion of the Coriolis force and time-explicit portion of vertical friction. These calculations are coordinated by a routine in

`ocean_core/ocean_velocity.F90.`

- Update the vertical elements to the velocity cell using code in

`ocean_core/ocean_thickness.F90.`

- Update the three dimensional velocity, including time implicit portions of the Coriolis forcing, vertical friction, and boundary forcing. This calculation is located in

`ocean_core/ocean_velocity.F90.`

- Complete a time step by computing some optional diagnostic quantities, such as energy analyses, global tracer budgets, etc. This calculation is coordinated by routines in the directory

`ocean_diag/.`

- Return ocean fields to the driver, including in particular the surface ocean properties that are used to compute boundary fluxes exchanged with the sea ice and atmosphere models.

1.6 Papers and reports providing documentation of MOM

The following is an incomplete list of documents that provide some further documentation of MOM than that provided here. We list in particular those papers and reports that provide some details of use to help understand aspects of the code and its use in applications.

- **MOM manuals and ocean model monograph:** As mentioned at the start of this chapter, the present document aims to incorporate many of the salient features from previous MOM related technical guides and monographs. The following lists these other documents.
 - The MOM3 Manual of [Pacanowski and Griffies \(1999\)](#)
 - Fundamentals of Ocean Climate Models by [Griffies \(2004\)](#)
 - A Technical Guide to MOM4.0 by [Griffies et al. \(2004\)](#)
 - Elements of MOM4p1 by [Griffies \(2009\)](#).
- **IPCC AR4 related papers:** The main application of MOM at GFDL relates to the study of global climate. MOM4.0 is largely the product of developing the IPCC AR4 climate models CM2.0 and CM2.1. MOM4p1 is largely the product of developing the IPCC AR5 models CM3 and ESM2M. The following papers document these models, with much in these papers of use for understanding MOM.
 - The paper by [Griffies et al. \(2005\)](#) provides a formulation of the ocean climate model used in the GFDL CM2 climate model for the study of global climate variability and change. The ocean code is based on MOM4.0.
 - The paper by [Gnanadesikan et al. \(2006\)](#) describes the ocean simulation characteristics from the coupled climate model CM2.
 - The paper by [Delworth et al. \(2006\)](#) describes the coupled climate model CM2.
 - The paper by [Wittenberg et al. \(2006\)](#) focuses on the tropical simulations in the CM2 coupled climate model.
 - The paper by [Stouffer et al. \(2006a\)](#) presents some idealized climate change simulations with the coupled climate model CM2.
- **IPCC AR5 related papers:** Recent development of MOM at GFDL, in particular MOM4p1, is largely associated with development of the IPCC AR5 models.
 - **ESM2M:** Development of the earth system model ESM2M ([Dunne et al., 2012, 2013](#)) prompted many developments in MOM. It is anticipated that further papers will be written that focus on physical aspects of the ocean component.
 - **CM3:** The climate model CM3 was developed using MOM configured very similarly to the CM2.1 ocean component. Nonetheless, the paper by [Griffies et al. \(2011\)](#) is of use to document various aspects of the ocean simulation that may be of use to those wishing to understand a bit more about MOM.
- **Regional modeling:** The paper by [Herzfeld et al. \(2011\)](#) documents the use of MOM for regional modeling.
- **Eddying global modeling:** The paper by [Delworth et al. \(2012\)](#) documents the use of MOM as a component to an eddying coupled climate model, with the ocean resolution no coarser than 25km and the atmospheric resolution roughly 50km. Further development with this model is focused on the C-grid aspects available in the 2012 release of MOM.
- **Lagrangian submodeling:** This work largely remains ongoing, with the most extensive documentation given by [Bates \(2011\)](#) as well as [Bates et al. \(2012a,b\)](#).

1.7 Remainder of this document

This document aims to provide the reader with a reasonably full accounting of the theoretical foundations of MOM, along with a thorough understanding of its use as a tool to study the ocean system. This document is split into chapters, with chapters in turn grouped into parts. An attempt is made to identify that portion of the code associated with each of the chapters.

It is inevitable that certain topics will be either incomplete or totally absent. Such represents more a limitation of those contributing to this document than a statement about the relative importance of a topic. We value all input on the document and will aim to improve the presentation with future drafts.

Formulation of the ocean equations

Descriptive methods provide a foundation for physical oceanography. Indeed, many observational oceanographers are masters at weaving a physical story of the ocean. Once a grounding in observations and experimental science is established, it is the job of the theorist to rationalize the phenomenology. For physical oceanography, these fundamentals largely rest in the classical mechanics of continuous fluids combined with continuum linear irreversible thermodynamics. That is, for a fundamental understanding, it is necessary to combine the descriptive, and more generally the experimental, approaches with theoretical methods based on mathematical physics. Together, the descriptive/experimental and theoretical methods render deep understanding of physical phenomena, and provide rational, albeit imperfect, predictions of unobserved phenomena, including the state of future ocean climate.

Many courses in physics introduce the student to mathematical tools required to garner a quantitative understanding of physical phenomena. Mathematical methods add to the clarity, conciseness, and precision of our description of physical phenomena, and so enhance our ability to unravel the essential physical processes involved with a phenomenon.

The purpose of this part is to mathematically formulate the fundamental equations providing the rational basis of the MOM ocean code. It is assumed that the reader has a basic understanding of calculus and fluid mechanics. Much of the presentation starts from first principles, and so should be accessible to those unfamiliar with physical oceanography.

FUNDAMENTAL EQUATIONS

Contents

2.1	Fluid kinematics	26
2.1.1	Mass conserving fluid parcels	26
2.1.2	Volume conserving fluid parcels	28
2.1.3	Mass conservation for finite domains	28
2.1.4	Dia-surface transport	29
2.1.4.1	Basic formulation	29
2.1.4.2	Alternative expressions for the dia-surface velocity component	31
2.1.5	Solid earth kinematic boundary condition	32
2.1.5.1	Orthogonal coordinates	32
2.1.5.2	Generalized vertical coordinates	33
2.1.6	Upper surface kinematic condition	35
2.1.6.1	Orthogonal coordinates	35
2.1.6.2	Generalized vertical coordinates	37
2.2	Mass conservation and the tracer equation	37
2.2.1	Eulerian form of mass conservation	37
2.2.2	Mass conservation for parcels	38
2.2.3	Mass conservation for constituents: the tracer equation	39
2.3	Thermodynamical tracers	39
2.3.1	Potential temperature	40
2.3.2	Potential enthalpy	40
2.4	Material time changes over finite regions	42
2.5	Basics of the finite volume method	44
2.6	Mass and tracer budgets over finite regions	45
2.6.1	General formulation	45
2.6.2	Budget for an interior grid cell	47
2.6.3	Cells adjacent to the ocean bottom	49
2.6.4	Cells adjacent to the ocean surface	50
2.7	Special considerations for tracers	51
2.7.1	Compatibility between vertically integrated mass and tracer budgets	51
2.7.2	Fresh water budget	52
2.7.3	The ideal age tracer	52
2.7.4	Budgets without dia-surface fluxes	53
2.8	Forces from pressure	53

2.8.1	The accumulation of contact pressure forces	55
2.8.2	Pressure gradient body force in hydrostatic fluids	57
2.9	Linear momentum budget	58
2.9.1	General formulation	59
2.9.2	An interior grid cell	59
2.9.3	Cell adjacent to the ocean bottom	60
2.9.4	Cell adjacent to the ocean surface	61
2.9.5	Horizontal momentum equations for hydrostatic fluids	62
2.10	The Boussinesq budgets	62

The purpose of this chapter is to formulate the kinematic and dynamic equations that form the basis for MOM. Much of this material is derived from lectures of [Griffies \(2005\)](#) at the 2004 GODAE School on Operational Oceanography. The proceedings of this school have been put together by [Chassignet and Verron \(2005\)](#), and this book contains many pedagogical reviews of ocean modelling. Additional discussions can be found in [Griffies \(2004\)](#), and [Griffies and Adcroft \(2008\)](#). The material here should be accessible to those having some exposure to the basics of mathematical physics, yet there is little assumed about knowledge of fluid mechanics.

2.1 Fluid kinematics

Kinematics is the study of the intrinsic properties of motion, without concern for dynamical laws. The purpose of this section is to derive some of the basic equations of fluid kinematics applied to the ocean. As considered here, fluid kinematics is concerned with balances of mass for infinitesimal fluid parcels or finite regions of the ocean. It is also concerned with the behaviour of a fluid as it interacts with geometrical boundaries of the domain, such as the land-sea and air-sea boundaries of an ocean basin.

2.1.1 Mass conserving fluid parcels

Consider an infinitesimal parcel of seawater contained in a volume of size¹

$$dV = dx dy dz \quad (2.1)$$

with a mass given by

$$dM = \rho dV. \quad (2.2)$$

In these equations, ρ is the *in situ* mass density of the parcel and $\mathbf{x} = (x, y, z)$ is the Cartesian coordinate of the parcel with respect to an arbitrary origin. As the parcel moves through space-time, we measure its velocity

$$\mathbf{v} = \frac{d\mathbf{x}}{dt} \quad (2.3)$$

by considering the time changes in its position.²

The time derivative d/dt introduced in equation (2.3) measures time changes of a fluid property as one follows the parcel. That is, we place ourselves in the parcel's moving frame of reference. This time derivative is thus directly analogous to that employed in classical particle mechanics ([Landau and Lifshitz, 1976](#); [Marion and Thornton, 1988](#)). Describing fluid motion from the perspective of an observer moving with fluid parcels affords us with a *Lagrangian* description of fluid mechanics. For many purposes, it is

¹A parcel of fluid is macroscopically small yet microscopically large. That is, from a macroscopic perspective, the parcel's thermodynamic properties may be assumed uniform, and the methods of continuum mechanics are applicable to describing the mechanics of an infinite number of these parcels. However, from a microscopic perspective, these fluid parcels contain many molecules, and so it is safe to ignore the details of molecular interactions. Regions of a fluid with length scales on the order of 10^{-3} cm satisfy these properties of a fluid parcel. See Section 2.2 of [Griffies \(2004\)](#) for further discussion.

²The three dimensional velocity vector is written $\mathbf{v} = (\mathbf{u}, w)$ throughout these notes, with $\mathbf{u} = (u, v)$ the horizontal components and w the vertical component.

useful to take a complementary perspective in which we measure fluid properties from a fixed space frame, and so allow fluid parcels to stream by the observer. The fixed space frame affords one with an *Eulerian* description of fluid motion. To relate the time tendencies of scalar properties measured in the moving and fixed frames, we perform a coordinate transformation, the result of which is (see Section 2.3.3 of [Griffies \(2004\)](#) for details)

$$\frac{d}{dt} = \partial_t + \mathbf{v} \cdot \nabla, \quad (2.4)$$

where

$$\partial_t = \frac{\partial}{\partial t} \quad (2.5)$$

measures time changes at a fixed space point. The *transport* term

$$\mathbf{v} \cdot \nabla = \mathbf{u} \cdot \nabla_s + w^{(s)} \partial_z \quad (2.6)$$

reveals the fundamentally nonlinear character of fluid dynamics. In this relation, ∇_s is the horizontal gradient operator taken on surfaces of constant generalized vertical coordinate s , and $w^{(s)}$ measures the transport of fluid crossing these surfaces. We provide further discussion of this expression in Section 2.1.4. In general, the operator $\mathbf{v} \cdot \nabla$ is known as the *advection* operator in geophysical fluids, whereas it is often termed *convection* in the classical fluids literature.³

It is convenient, and conventional, to formulate the mechanics of fluid parcels that conserve mass. Choosing to do so allows many notions from classical particle mechanics to transfer over to continuum mechanics of fluids, especially when formulating the equations of motion from a Lagrangian perspective. We thus focus on kinematics satisfied by mass conserving fluid parcels. In this case, the mass of a parcel changes only if there are sources within the continuous fluid, so that

$$\frac{d}{dt} \ln(dM) = \mathcal{S}^{(M)} \quad (2.7)$$

where $\mathcal{S}^{(M)}$ is the rate at which mass is added to the fluid, per unit mass. Mass sources are often assumed to vanish in textbook formulations of fluid kinematics. However, they can be nonzero in certain cases for ocean modelling in which mass is moved from one region to another through certain subgrid scale parameterizations, such as the cross land schemes of Chapters 29 and 30. So it is convenient to carry mass sources around in our formulation of the equations used by MOM.

Equation (2.7) expresses mass conservation for fluid parcels in a Lagrangian form. To derive the Eulerian form of mass conservation, start by substituting the mass of a parcel given by equation (2.2) into the mass conservation equation (2.7) to derive

$$\frac{d}{dt} \ln \rho = -\nabla \cdot \mathbf{v} + \mathcal{S}^{(M)}. \quad (2.8)$$

That is, the density of a parcel increases when the velocity field converges onto the parcel. To reach this result, we first note the expression

$$\frac{d}{dt} \ln(dV) = \nabla \cdot \mathbf{v}, \quad (2.9)$$

which says that the infinitesimal volume of a fluid parcel increases in time if the velocity of the parcel diverges from the location of the parcel. Imagine the parcel expanding in response to the diverging velocity field.

Upon deriving the material evolution of density as given by equation (2.8), rearrangement renders the Eulerian form of mass conservation

$$\rho_{,t} + \nabla \cdot (\rho \mathbf{v}) = \rho \mathcal{S}^{(M)}. \quad (2.10)$$

A comma is used here as shorthand for the partial time derivative taken at a fixed point in space

$$\rho_{,t} = \frac{\partial \rho}{\partial t}. \quad (2.11)$$

³*Convection* in geophysical fluid dynamics generally refers to the rapid vertical motions that act to stabilize fluids that are gravitationally unstable (e.g., [Marshall and Schott, 1999](#)).

We use an analogous notation for other partial derivatives throughout these notes. Rewriting mass conservation in terms of the density time tendency

$$\rho_{,t} = -\nabla \cdot (\rho \mathbf{v}) + \rho \mathcal{S}^{(M)}, \quad (2.12)$$

reveals that at each point in the fluid, the mass density increases if the linear momentum per volume of the fluid parcel,

$$\mathbf{p} = \rho \mathbf{v}, \quad (2.13)$$

converges to the point.

2.1.2 Volume conserving fluid parcels

Fluids that are comprised of parcels that conserve their mass, as considered in the previous discussion, satisfy *non-Boussinesq* kinematics. In ocean climate modelling, it has been traditional to exploit the large degree to which the ocean fluid is incompressible, in which case the volume of fluid parcels is taken as constant. These fluids are said to satisfy *Boussinesq* kinematics.

For the Boussinesq fluid, conservation of volume for a fluid parcel leads to

$$\frac{d}{dt} \ln(dV) = \mathcal{S}^{(V)}, \quad (2.14)$$

where $\mathcal{S}^{(V)}$ is the volume source per unit volume present within the fluid. It is numerically the same as the mass source $\mathcal{S}^{(M)}$ defined in equation (2.7). This statement of volume conservation is equivalent to the mass conservation statement (2.7) if we assume the mass of the parcel is given by

$$dM = \rho_o dV, \quad (2.15)$$

where ρ_o is a constant reference density.

Using equation (2.9) in the Lagrangian volume conservation statement (2.14) leads to the following *constraint* for the Boussinesq velocity field

$$\nabla \cdot \mathbf{v} = \mathcal{S}^{(V)}. \quad (2.16)$$

Where the volume source vanishes, the three dimensional velocity field is non-divergent

$$\nabla \cdot \mathbf{v} = 0 \quad \text{for Boussinesq fluids with } \mathcal{S}^{(V)} = 0. \quad (2.17)$$

2.1.3 Mass conservation for finite domains

Now consider a finite sized region of ocean extending from the free surface at $z = \eta(x, y, t)$ to the solid earth boundary at $z = -H(x, y)$, and allow the fluid within this region to respect the mass conserving kinematics of a non-Boussinesq fluid. The total mass of fluid inside the region is given by

$$M = \int dx dy \int_{-H}^{\eta} \rho dz. \quad (2.18)$$

Conservation of mass for this region implies that the time tendency

$$\partial_t M = \int dx dy \partial_t \left(\int_{-H}^{\eta} dz \rho \right) \quad (2.19)$$

changes due to imbalances in the flux of seawater passing across the domain boundaries, and from sources within the region.⁴ For a region comprised of a vertical fluid column, the only means of affecting the mass are through fluxes crossing the ocean free surface, convergence of mass brought in by horizontal ocean

⁴We assume no water enters the domain through the solid-earth boundaries.

currents through the vertical sides of the column, and sources within the column. These considerations lead to the balance

$$\partial_t M = \int dx dy \left(Q_m + \int_{-H}^{\eta} dz \rho \mathcal{S}^{(M)} - \nabla \cdot \int_{-H}^{\eta} dz \rho \mathbf{u} \right). \quad (2.20)$$

The term $Q_m dx dy$ represents the mass flux of material (mass per unit time) crossing the free surface.⁵ We provide a more detailed accounting of this flux in Section 2.1.6. Equating the time tendencies given by equations (2.19) and (2.20) leads to a mass balance within each vertical column of fluid

$$\partial_t \left(\int_{-H}^{\eta} dz \rho \right) + \nabla \cdot \mathbf{U}^\rho = Q_m + \int_{-H}^{\eta} dz \rho \mathcal{S}^{(M)}, \quad (2.21)$$

where

$$\mathbf{U}^\rho = \int_{-H}^{\eta} dz \rho \mathbf{u} \quad (2.22)$$

is a shorthand notation for the vertically integrated horizontal momentum per volume.

Setting density factors in the mass conservation equation (2.21) to the constant reference density ρ_o renders the volume conservation equation

$$\partial_t \eta + \nabla \cdot \mathbf{U} = Q_m / \rho_o + \int_{-H}^{\eta} dz \mathcal{S}^{(V)} \quad (2.23)$$

appropriate for a Boussinesq fluid, where fluid parcels conserve volume rather than mass. In this equation

$$\mathbf{U} = \int_{-H}^{\eta} dz \mathbf{u} \quad (2.24)$$

is the vertically integrated horizontal velocity.

2.1.4 Dia-surface transport

A surface of constant generalized vertical coordinate, s , is of importance when establishing the balances of mass, tracer, and momentum within a layer of fluid whose upper and lower bounds are determined by surfaces of constant s . Fluid transport through this surface is said to constitute the *dia-surface* transport. This transport plays a fundamental role in generalized vertical coordinate models such as MOM. Additionally, when considering the flow of fluid and tracer properties across the ocean surface and bottom, the notions of dia-surface transport are useful.

Generalized vertical coordinates provide the ocean theorist and modeler with a powerful set of tools to describe ocean flow, which in many situations is far more natural than the more traditional geopotential coordinates (x, y, z) that we have been using thus far. Therefore, it is important for the student to gain some exposure to the fundamentals of these coordinates, as they are ubiquitous in ocean modelling today.

2.1.4.1 Basic formulation

At an arbitrary point on a surface of constant generalized vertical coordinate (see Figure 2.1), the flux of fluid in the direction normal to the surface is given by

$$\text{SEAWATER FLUX IN DIRECTION } \hat{\mathbf{n}} = \mathbf{v} \cdot \hat{\mathbf{n}}, \quad (2.25)$$

⁵Water crossing the ocean surface is typically quite fresh, such as for precipitation or evaporation. However, rivers and ice melt can generally contain a nonzero salinity. Additionally, for most climate model applications, the mass of salt particles exchanged across the liquid ocean interface upon melting and freezing of sea ice is ignored when considering the mass balance of the liquid ocean fluid.

with

$$\hat{\mathbf{n}} = \frac{\nabla s}{|\nabla s|} \quad (2.26)$$

the surface unit normal direction. Introducing the material time derivative $ds/dt = s_{,t} + \mathbf{v} \cdot \nabla s$ leads to the equivalent expression

$$\mathbf{v} \cdot \hat{\mathbf{n}} = |\nabla s|^{-1} (d/dt - \partial_t) s. \quad (2.27)$$

That is, the normal component to a fluid parcel's velocity is proportional to the difference between the material time derivative of the surface and its partial time derivative.

Since the surface is generally moving, the net flux of seawater penetrating the surface is obtained by subtracting the velocity of the surface $\mathbf{v}^{(\text{ref})}$ in the $\hat{\mathbf{n}}$ direction from the velocity component $\mathbf{v} \cdot \hat{\mathbf{n}}$ of the fluid parcels. We thus define the dia-surface velocity component according to

$$\begin{aligned} w^{(s)} &\equiv \text{FLUX OF SEAWATER THROUGH SURFACE} \\ &= \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}). \end{aligned} \quad (2.28)$$

The velocity $\mathbf{v}^{(\text{ref})}$ is the velocity of a reference point fixed on the surface, which is defined so that

$$(\partial_t + \mathbf{v}^{(\text{ref})} \cdot \nabla) s = 0. \quad (2.29)$$

Consequently,

$$\hat{\mathbf{n}} \cdot \mathbf{v}^{(\text{ref})} = -s_{,t} |\nabla s|^{-1}, \quad (2.30)$$

so that the normal component of the surface's velocity vanishes when the surface is static.

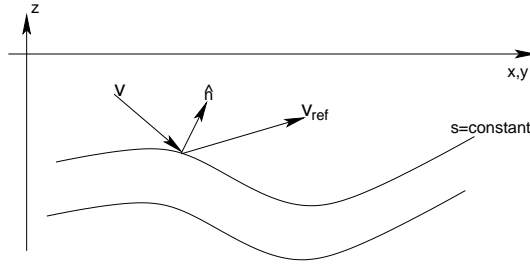


Figure 2.1: Surfaces of constant generalized vertical coordinate living interior to the ocean. An upward normal direction $\hat{\mathbf{n}}$ is indicated on one of the surfaces. Also shown is the orientation of a fluid parcel's velocity \mathbf{v} and the velocity $\mathbf{v}^{(\text{ref})}$ of a reference point living on the surface.

Expression (2.30) then leads to the following expression for the net flux of seawater crossing the surface

$$\begin{aligned} w^{(s)} &= \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) \\ &= |\nabla s|^{-1} (\partial_t + \mathbf{v} \cdot \nabla) s \\ &= |\nabla s|^{-1} \frac{ds}{dt}. \end{aligned} \quad (2.31)$$

Hence, the material time derivative of the generalized surface vanishes if and only if no water parcels cross it. This important result is used throughout ocean theory and modelling. It measures the volume of seawater crossing a generalized surface, per time, per area. The area normalizing the volume flux is that area $dA_{(\hat{\mathbf{n}})}$ of an infinitesimal patch on the surface of constant generalized vertical coordinate with outward unit normal $\hat{\mathbf{n}}$. When surfaces of constant generalized vertical coordinate are monotonically stacked in the vertical, this area factor can be written (see equation (6.58) of Griffies (2004))

$$dA_{(\hat{\mathbf{n}})} = |z_{,s} \nabla s| dA, \quad (2.32)$$

where

$$dA = dx dy \quad (2.33)$$

is the horizontal projection of the area element. Hence, the volume per time of fluid passing through the generalized surface is

$$\begin{aligned} (\text{VOLUME/TIME}) \text{ THROUGH SURFACE} &= \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) dA_{(\hat{\mathbf{n}})} \\ &= |z_{,s}| (ds/dt) dx dy, \end{aligned} \quad (2.34)$$

and the magnitude of this flux is

$$|\hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) dA_{(\hat{\mathbf{n}})}| \equiv |w^{(z)}| dx dy. \quad (2.35)$$

We introduced the expression

$$w^{(z)} = z_{,s} \frac{ds}{dt}, \quad (2.36)$$

which measures the volume of fluid passing through the surface, per unit area $dA = dx dy$ of the horizontal projection of the surface, per unit time. That is,

$$\begin{aligned} w^{(z)} &\equiv \frac{\hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) dA_{(\hat{\mathbf{n}})}}{dA} \\ &= \frac{(\text{VOLUME/TIME}) \text{ OF FLUID THROUGH SURFACE}}{\text{AREA OF HORIZONTAL PROJECTION OF SURFACE}}. \end{aligned} \quad (2.37)$$

The quantity $w^{(z)}$ is the dia-surface velocity component that appears in the budget equations for mass, tracer, and momentum in the generalized level formulation of MOM.

2.1.4.2 Alternative expressions for the dia-surface velocity component

To gain some experience with the dia-surface velocity component, it is useful to write it in the equivalent forms

$$\begin{aligned} w^{(z)} &= z_{,s} \frac{ds}{dt} \\ &= z_{,s} \nabla s \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) \\ &= (\dot{\mathbf{z}} - \nabla_s z) \cdot \mathbf{v} - z_{,t} \\ &= w - (\partial_t + \mathbf{u} \cdot \nabla_s) z \end{aligned} \quad (2.38)$$

where the penultimate step used the identity (2.40), and where

$$\begin{aligned} \mathbf{S} &= \nabla_s z \\ &= -z_{,s} \nabla_z s \end{aligned} \quad (2.39)$$

is the slope of the s surface as projected onto the horizontal directions. For example, if the slope vanishes, then the dia-surface velocity component measures the flux of fluid moving vertically relative to the motion of the generalized surface. When the surface is static and flat, then the dia-surface velocity component is simply the vertical velocity component $w = dz/dt$.

When interpreting the dia-surface velocity component below, we find it useful to note that relation (2.30) leads to

$$z_{,s} \nabla s \cdot \mathbf{v}^{(\text{ref})} = z_{,t}. \quad (2.40)$$

To reach this result, we used the identity $s_{,t} z_{,s} = -z_{,t}$, with $z_{,t}$ the time tendency for the depth of a particular constant s surface.

The expression (2.36) for $w^{(z)}$ brings the material time derivative (2.4) into the following equivalent forms

$$\frac{d}{dt} = \left(\frac{\partial}{\partial t} \right)_z + \mathbf{u} \cdot \nabla_z + w \left(\frac{\partial}{\partial z} \right) \quad (2.41)$$

$$= \left(\frac{\partial}{\partial t} \right)_s + \mathbf{u} \cdot \nabla_s + \frac{ds}{dt} \left(\frac{\partial}{\partial s} \right) \quad (2.42)$$

$$= \left(\frac{\partial}{\partial t} \right)_s + \mathbf{u} \cdot \nabla_s + w^{(z)} \left(\frac{\partial}{\partial z} \right), \quad (2.43)$$

where

$$\partial_s = z_{,s} \partial_z \quad (2.44)$$

relates the vertical coordinate partial derivatives. The form given by equation (2.43) motivates some to refer to $w^{(z)}$ as a vertical velocity component that measures the rate at which fluid parcels penetrate the surface of constant generalized coordinate (see Appendix A to McDougall (1995)). Indeed, such is part of the motivation for using the (z) superscript notation. However, we must be careful to distinguish $w^{(z)}$ from the generally different vertical velocity component $w = dz/dt$, which measures the water flux crossing constant geopotential surfaces.

We close with a few points of clarification for the case where no fluid parcels cross the generalized surface. Such occurs, in particular, in the case of adiabatic flows with $s = \rho$ an isopycnal coordinate. In this case, the material time derivative (2.43) only has a horizontal two-dimensional advective component $\mathbf{u} \cdot \nabla_s$. This result *should not* be interpreted to mean that the velocity of a fluid parcel is strictly horizontal. Indeed, it generally is not, as the form (2.41) should make clear. Rather, it means that the transport of fluid properties occurs along surfaces of constant s , and such transport is measured by the convergence of horizontal advective fluxes as measured along surfaces of constant s . We revisit this point in Section 2.6.2 when discussing tracer transport (see in particular Figure 2.7).

2.1.5 Solid earth kinematic boundary condition

We now apply the discussion of dia-surface transport from Section 2.1.4 to perhaps the simplest surface; namely, the time independent solid earth boundary. This surface is commonly assumed to be impenetrable to fluid.⁶ The expression for fluid transport at the lower surface leads to the *solid earth kinematic boundary condition*. In addition to deriving the bottom kinematic boundary condition, we introduce some mathematical techniques useful when working with non-orthogonal generalized vertical coordinates, as used in many ocean models such as MOM.

2.1.5.1 Orthogonal coordinates

As there is no fluid crossing the solid earth lower boundary, a no-normal flow condition is imposed at the solid earth boundary at the depth

$$z = -H(x, y). \quad (2.45)$$

To develop a mathematical expression for the boundary condition, note that the outward unit normal pointing from the ocean into the underlying rock is given by⁷ (see Figure 2.2)

$$\hat{\mathbf{n}}_H = - \left(\frac{\nabla(z+H)}{|\nabla(z+H)|} \right). \quad (2.46)$$

Furthermore, we assume that the bottom topography can be represented as a continuous function $H(x, y)$ that does not possess “overturns.” That is, we do not consider caves or overhangs in the bottom boundary

⁶This assumption may be broken in some cases. For example, when the lower boundary is a moving sedimentary layer in a coastal estuary, or when there is seeping ground water. We do not consider such cases here.

⁷The three dimensional gradient operator $\nabla = (\partial_x, \partial_y, \partial_z)$ reduces to the two dimensional horizontal operator $\nabla_z = (\partial_x, \partial_y, 0)$ when acting on functions that depend only on the horizontal directions. To reduce notation clutter, we do not expose the z subscript in cases where it is clear that the horizontal gradient is all that is relevant.

where the topographic slope becomes infinite. Such would make it difficult to consider the slope of the bottom in our formulations. This limitation is common for ocean models.⁸

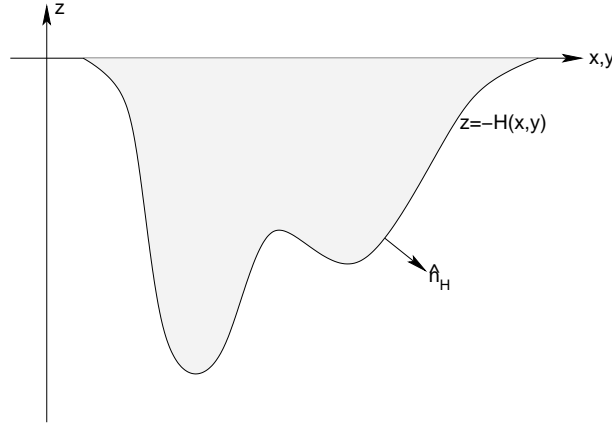


Figure 2.2: Schematic of the ocean's bottom surface with a smoothed undulating solid earth topography at $z = -H(x, y)$ and outward normal direction $\hat{\mathbf{n}}_H$. Undulations of the solid earth can reach from the ocean bottom at 5000m-6000m to the surface over the course of a few kilometers (slopes on the order of 0.1 to 1.0). These ranges of topographic variation are far greater than the surface height (see Figure 2.3). It is important for simulations to employ numerics that facilitate an accurate representation of the ocean bottom.

A no-normal flow condition on fluid flow at the ocean bottom implies

$$\mathbf{v} \cdot \hat{\mathbf{n}}_H = 0 \quad \text{at } z = -H(x, y). \quad (2.47)$$

Expanding this constraint into its horizontal and vertical components yields

$$\mathbf{u} \cdot \nabla H + w = 0 \quad \text{at } z = -H(x, y). \quad (2.48)$$

Furthermore, introducing a material time derivative (2.4) allows us to write this boundary condition as

$$\frac{d(z + H)}{dt} = 0 \quad \text{at } z = -H(x, y). \quad (2.49)$$

Equation (2.49) expresses in a material or Lagrangian form the impenetrable nature of the solid earth lower surface, whereas equation (2.48) expresses the same constraint in an Eulerian form.

2.1.5.2 Generalized vertical coordinates

We now consider the form of the bottom kinematic boundary condition in generalized vertical coordinates. Chapter 6 of Griffies (2004) develops a calculus for generalized vertical coordinates. Experience with these methods is useful to nurture an understanding for ocean modelling in generalized vertical coordinates. Most notably, these coordinates, when used with the familiar horizontal coordinates (x, y) , form a non-orthogonal triad, and thus lead to some relationships that may be unfamiliar. To proceed in this section, we present some salient results of the mathematics of generalized vertical coordinates, and reserve many of the derivations for Griffies (2004).

When considering generalized vertical coordinates for ocean models, we assume that the surfaces in question do not overturn on themselves. This constraint means that the Jacobian of transformation between the generalized vertical coordinate

$$s = s(x, y, z, t) \quad (2.50)$$

⁸For hydrostatic models, the solution algorithms rely on the ability to integrate vertically from the ocean bottom to the top, uninterrupted by rock in between. Non-hydrostatic models do not employ such algorithms, and so may in principle allow for arbitrary bottom topography, including overhangs.

and the geopotential coordinate z , must be one signed. That is, the specific thickness

$$\frac{\partial z}{\partial s} = z_{,s} \quad (2.51)$$

is of the same sign throughout the ocean fluid. The name *specific thickness* arises from the property that

$$dz = z_{,s} ds \quad (2.52)$$

is an expression for the thickness of an infinitesimal layer of fluid bounded by two constant s surfaces.

Deriving the bottom kinematic boundary condition in s -coordinates requires a relation between the vertical velocity component used in geopotential coordinates, $w = dz/dt$, and the pseudo-velocity component ds/dt . For this purpose, we refer to some results from Section 6.5.5 of Griffies (2004). As in that discussion, we derive the isomorphic relations

$$\dot{z} = (\partial_t + \mathbf{u} \cdot \nabla_s + \dot{s} \partial_s) z \quad (2.53)$$

$$\dot{s} = (\partial_t + \mathbf{u} \cdot \nabla_z + \dot{z} \partial_z) s, \quad (2.54)$$

where

$$\dot{z} = \frac{dz}{dt} \quad (2.55)$$

$$\dot{s} = \frac{ds}{dt} \quad (2.56)$$

are useful shorthands for the vertical velocity components, motivated from similar notation used in classical particle mechanics. Note that the partial time derivative appearing in each of the expressions is taken with the corresponding space variables held fixed. That is, ∂_t in equation (2.53) is taken with s held fixed, whereas ∂_t in equation (2.54) is taken with z held fixed.

Rearrangement of equations (2.53) and (2.54) leads to

$$\dot{z} = z_{,s} (d/dt - \partial_t - \mathbf{u} \cdot \nabla_z) s. \quad (2.57)$$

This expression is relevant when measurements are taken on surfaces of constant geopotential, or depth. To reach this result, we made use of the triple product identities

$$z_{,t} = -s_{,t} z_{,s} \quad (2.58)$$

$$z_{,x} = -s_{,x} z_{,s} \quad (2.59)$$

$$z_{,y} = -s_{,y} z_{,s}. \quad (2.60)$$

A derivation of these identities is given in Section 6.5.4 of Griffies (2004). These relations should be familiar to those having studied thermodynamics, where the analogous expressions are known as the Maxwell relations (e.g., Callen, 1985).

We now apply relation (2.57) to the ocean bottom, which is generally not a surface of constant depth. It is thus necessary to transform the constant depth gradient ∇_z to a horizontal gradient taken along the bottom. To do so, proceed as in Section 6.5.3 of Griffies (2004) and consider the time-independent coordinate transformation

$$(\bar{x}, \bar{y}, \bar{z}, \bar{t}) = (x, y, -H(x, y), t). \quad (2.61)$$

The horizontal gradient taken on constant depth surfaces, ∇_z , and the horizontal gradient along the bottom, $\nabla_{\bar{z}}$, are thus related by

$$\nabla_{\bar{z}} = \nabla_z - (\nabla H) \partial_z. \quad (2.62)$$

Using this result in equation (2.57) yields

$$s_{,z} (w + \mathbf{u} \cdot \nabla H) = (d/dt - \partial_t - \mathbf{u} \cdot \nabla_{\bar{z}}) s \quad \text{at } z = -H. \quad (2.63)$$

The left hand side vanishes due to the kinematic boundary condition (2.48), which then leads to

$$ds/dt = (\partial_t + \mathbf{u} \cdot \nabla_{\bar{z}}) s \quad \text{at } s = s(x, y, z = -H(x, y), t). \quad (2.64)$$

The value of the generalized coordinate at the ocean bottom can be written in the shorthand form

$$s_{\text{bot}}(x, y, t) = s(x, y, z = -H, t) \quad (2.65)$$

which leads to

$$\frac{d(s - s_{\text{bot}})}{dt} = 0 \quad \text{at } s = s_{\text{bot}}. \quad (2.66)$$

This relation is analogous to equation (2.49) appropriate to z -coordinates. Indeed, it is actually a basic statement of the impenetrable nature of the solid earth lower boundary, which is true regardless the vertical coordinates.

The various mathematical steps that led to the very simple result (2.66) could have been dispensed with if we already understood some notions of generalized vertical coordinates. Nonetheless, the steps introduced some of the formalism required to work with generalized vertical coordinates, and as such provide a useful testing ground for later manipulations where the answer is less easy to anticipate. This strategy is highly recommended to the student working with new formalisms. That is, first test your mathematical skills with problems where the answer is either known, or can be readily judged correct with basic physical understanding. After garnering experience and confidence, one may then approach genuinely new problems using the methods.

2.1.6 Upper surface kinematic condition

To formulate budgets for mass, tracer, and momentum in the ocean, we consider the upper ocean surface to be a time dependent permeable membrane through which precipitation, evaporation, ice melt, and river runoff⁹ pass. The expression for fluid transport at the upper surface leads to the *upper ocean kinematic boundary condition*.

2.1.6.1 Orthogonal coordinates

To describe the kinematics of water transport into the ocean, it is useful to introduce an effective transport through a smoothed ocean surface, where smoothing is performed via an ensemble average. We assume that this averaging leads to a surface absent overturns or breaking waves, thus facilitating a mathematical description analogous to the ocean bottom just considered. The value of the geopotential at the ocean surface takes on the value

$$z = \eta(x, y, t) \quad (2.67)$$

at this idealized ocean surface. Correspondingly, the mass flux of material crossing the ocean surface is written

$$\begin{aligned} Q_m &= \frac{(\text{MASS/TIME}) \text{ THROUGH FREE SURFACE}}{\text{HORIZONTAL AREA UNDER FREE SURFACE}} \\ &= - \left(\frac{\rho dA(\hat{\mathbf{n}}) \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{\text{ref}})}{dA} \right) \quad \text{at } z = \eta. \end{aligned} \quad (2.68)$$

In this expression, the outward normal

$$\hat{\mathbf{n}} = \left(\frac{\nabla(z - \eta)}{|\nabla(z - \eta)|} \right) \quad \text{at } z = \eta \quad (2.69)$$

points from the ocean surface at $z = \eta$ into the overlying atmosphere (see Figure 2.3). The velocity \mathbf{v}^{ref} is taken from a point fixed on the free surface, so that

$$\partial_t(z - \eta) + \mathbf{v}^{\text{ref}} \cdot \nabla(z - \eta) = 0, \quad (2.70)$$

⁹River runoff generally enters the ocean at a nonzero depth rather than through the surface. Many global models, however, have traditionally inserted river runoff to the top model cell. Such can become problematic numerically and physically when the top grid cells are refined to levels common in coastal modelling. Hence, more applications are now considering the input of runoff throughout a nonzero depth. Likewise, sea ice can melt at depth, thus necessitating a mass transport to occur within the ocean between the liquid and solid water masses.

or equivalently

$$w^{\text{ref}} = (\partial_t + \nabla \cdot \mathbf{u}^{\text{ref}}) \eta \quad (2.71)$$

or

$$|\nabla(z - \eta)| \hat{\mathbf{n}} \cdot \mathbf{v}^{\text{ref}} = \partial_t \eta \quad \text{at } z = \eta. \quad (2.72)$$

Finally, the area element $dA_{(\hat{\mathbf{n}})}$ measures an infinitesimal area element on the ocean surface $z = \eta$, and it is given by (see Section 20.13.2 of [Griffies \(2004\)](#))

$$dA_{(\hat{\mathbf{n}})} = |\nabla(z - \eta)| dA \quad \text{at } z = \eta, \quad (2.73)$$

where $dA = dx dy$ is the horizontal area element. Use of these relations leads to the surface kinematic boundary condition written in material form

$$\rho \frac{d(z - \eta)}{dt} = -Q_m \quad \text{at } z = \eta. \quad (2.74)$$

Contrary to the solid earth condition (2.49), where $z + H$ is materially constant, permeability of the ocean surface leads to a nontrivial material evolution of $z - \eta$.

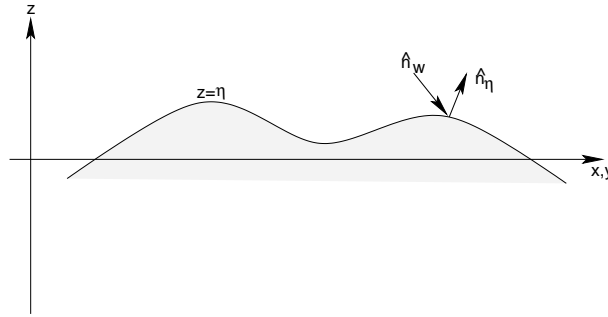


Figure 2.3: Schematic of the ocean's upper surface with a smoothed undulating surface at $z = \eta(x, y, t)$, outward normal direction $\hat{\mathbf{n}}_\eta$, and normal direction $\hat{\mathbf{n}}_w$ orienting the passage of water across the surface. Undulations of the surface height are on the order of a few metres due to tidal fluctuations in the open ocean, and order 10m-20m in certain embayments (e.g., Bay of Fundy in Nova Scotia). When imposing the weight of sea ice onto the ocean surface, the surface height can depress even further, on the order of 5m-10m, with larger values possible in some cases. It is important for simulations to employ numerical schemes facilitating such wide surface height undulations.

As an alternative means to develop the surface kinematic boundary condition, return to the expression (2.21) for mass conservation, rewritten here for completeness

$$\partial_t \left(\int_{-H}^{\eta} dz \rho \right) + \nabla \cdot \left(\int_{-H}^{\eta} dz \rho \mathbf{u} \right) = Q_m + \int_{-H}^{\eta} dz \rho \mathcal{S}^{(M)}. \quad (2.75)$$

Next, perform the derivative operations on the integrals, making use of Leibnitz's Rule when differentiating the integrals. The first step of the derivation leads to

$$[\rho(\partial_t + \mathbf{u} \cdot \nabla) \eta]_{z=\eta} + [\rho \nabla H \cdot \mathbf{u}]_{z=-H} + \int_{-H}^{\eta} dz [\rho_{,t} + \nabla \cdot (\rho \mathbf{u})] = Q_m + \int_{-H}^{\eta} dz \rho \mathcal{S}^{(M)}. \quad (2.76)$$

The Eulerian mass conservation relation (2.10) and bottom kinematic boundary condition (2.48) render the Eulerian form of the surface kinematic boundary condition

$$\rho(\partial_t + \mathbf{u} \cdot \nabla) \eta = Q_m + \rho w \quad \text{at } z = \eta. \quad (2.77)$$

2.1.6.2 Generalized vertical coordinates

To derive the s -coordinate surface kinematic boundary condition, we proceed as for the bottom boundary condition in Section 2.1.5.2. Here, the coordinate transformation is time dependent

$$(\bar{x}, \bar{y}, \bar{z}, \bar{t}) = (x, y, \eta(x, y, t), t). \quad (2.78)$$

The horizontal gradient and time derivative operators are therefore related by

$$\nabla_{\bar{z}} = \nabla_z + (\nabla\eta) \partial_z \quad (2.79)$$

$$\partial_{\bar{t}} = \partial_t + (\partial_t \eta) \partial_z. \quad (2.80)$$

Hence, the relation (2.57) between vertical velocity components takes the following form at the ocean surface

$$w = z_{,s} (d/dt - \partial_{\bar{t}} - \mathbf{u} \cdot \nabla_{\bar{z}}) s + (\partial_t + \mathbf{u} \cdot \nabla) \eta \quad \text{at } z = \eta. \quad (2.81)$$

Substitution of the z -coordinate kinematic boundary condition (2.77) leads to

$$\rho z_{,s} (d/dt - \partial_{\bar{t}} - \mathbf{u} \cdot \nabla_{\bar{z}}) s = -Q_m \quad \text{at } s = s_{\text{top}} \quad (2.82)$$

where $s_{\text{top}} = s(x, y, z = \eta, t)$ is the value of the generalized vertical coordinate at the ocean surface. Reorganizing the result (2.82) leads to the material time derivative form

$$\rho z_{,s} \left(\frac{d(s - s_{\text{top}})}{dt} \right) = -Q_m \quad \text{at } s = s_{\text{top}} \quad (2.83)$$

which is analogous to the z -coordinate result (2.74). Indeed, it can be derived trivially by noting that $dz/dt = z_{,s} ds/dt$. Even so, just as for the bottom kinematic boundary condition considered in Section 2.1.5.2, it is useful to have gone through these manipulations to garner experience and confidence with the formalism.

2.2 Mass conservation and the tracer equation

We revisit here the mathematical description of a mass conserving fluid parcel for the purpose of introducing the evolution equation for trace material within a parcel. For brevity, we ignore the possibilities of mass sources in this discussion, though note as in Section 2.1 that mass sources may be of use for implementing certain subgrid scale schemes in MOM. This discussion here follows that in Section II.2 of DeGroot and Mazur (1984), Section 8.4 of Chaikin and Lubensky (1995), and Section 3.3 of Müller (2006). See also the discussion in Warren (2009).

2.2.1 Eulerian form of mass conservation

Seawater consists of many material constituents, such as freshwater, salts and biogeochemical components, with the possibility also for chemical reactions to take place. For brevity, we ignore chemical reactions, though note that the following discussion can be generalized to such cases (see, for example, Section II-2 in DeGroot and Mazur, 1984).

The mass density of each constituent within a parcel of seawater is given by

$$\rho_n = \frac{\text{mass of component } n}{\text{volume of seawater parcel}}, \quad (2.84)$$

with the total density in a parcel given by the sum over all N constituents

$$\begin{aligned} \rho &= \sum_{n=1}^N \left(\frac{\text{mass of component } n}{\text{volume of seawater parcel}} \right) \\ &= \sum_{n=1}^N \rho_n. \end{aligned} \quad (2.85)$$

Observe that the mass of a seawater parcel is the sum of individual constituent masses (numerator in equation (2.85)), whereas the volume of the parcel is a complicated function of the temperature, pressure, and material constituents.

For an arbitrary finite region within the fluid, conservation of mass for each constituent takes the form

$$\partial_t \left(\int \rho_n dV \right) = - \int \rho_n \mathbf{v}_n \cdot d\mathbf{A}, \quad (2.86)$$

where

$$d\mathbf{A} = \hat{\mathbf{n}} dA_{(\hat{\mathbf{n}})} \quad (2.87)$$

is the area element on the region boundary, with $\hat{\mathbf{n}}$ the outward normal, and \mathbf{v}_n is the velocity of constituent n . Equation (2.86) says that the mass of each constituent within a region is affected by the flow of that constituent through the region boundaries.

Now apply the mass budget (2.86) to a static volume, in which case we can bring the time derivative inside the integral, and use Gauss' Theorem on the boundary integral to render

$$\int dV (\partial_t \rho_n + \nabla \cdot (\rho_n \mathbf{v}_n)) = 0. \quad (2.88)$$

Since the volume is arbitrary, this relation leads to the local expression of mass balance for each constituent

$$\partial_t \rho_n = -\nabla \cdot (\rho_n \mathbf{v}_n). \quad (2.89)$$

Summing over all constituents then leads to the familiar Eulerian expression of mass conservation

$$\partial_t \rho = -\nabla \cdot (\rho \mathbf{v}), \quad (2.90)$$

where

$$\mathbf{v} = \rho^{-1} \sum_{n=1}^N \rho_n \mathbf{v}_n \quad (2.91)$$

is the velocity for the center of mass of the parcel.

The density of seawater is often well approximated by

$$\rho \approx \rho_{\text{salt}} + \rho_{\text{fresh}}, \quad (2.92)$$

where ρ_{salt} is the mass of ocean “salt” per mass of seawater, and ρ_{fresh} is the mass of fresh water per mass of seawater. Other material constituents occur in such small concentrations that their contributions to the seawater density are generally ignored for purposes of ocean modeling.

2.2.2 Mass conservation for parcels

The material time derivative

$$\frac{d}{dt} = \partial_t + \mathbf{v} \cdot \nabla, \quad (2.93)$$

measures time changes of a fluid property for an observer moving with the center of mass velocity \mathbf{v} . Mass conservation (2.90) in the moving, or *Lagrangian*, frame then takes the form

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v}, \quad (2.94)$$

indicating that the density of a fluid parcel increases in regions where currents converge, and density decreases where currents diverge.

2.2.3 Mass conservation for constituents: the tracer equation

Introducing the material time derivative to the constituent mass balances (2.89) leads to the material budget

$$\frac{d\rho_n}{dt} = -\rho_n \nabla \cdot \mathbf{v} - \nabla \cdot [\rho_n (\mathbf{v}_n - \mathbf{v})]. \quad (2.95)$$

Now define the *relative mass flux*

$$\mathbf{J}_n = \rho_n (\mathbf{v}_n - \mathbf{v}) \quad (2.96)$$

to render an expression for the material evolution of the density for each constituent

$$\frac{d\rho_n}{dt} = -\rho_n \nabla \cdot \mathbf{v} - \nabla \cdot \mathbf{J}_n. \quad (2.97)$$

The flux \mathbf{J}_n is nonzero for those motions where the constituent n moves relative to the parcel's center of mass. This motion can be caused by many effects, with molecular diffusion the canonical example, in which case we parameterize \mathbf{J}_n as a downgradient diffusive flux.¹⁰ Notably, the total mass flux vanishes

$$\mathbf{J} = \sum_{n=1}^N \mathbf{J}_n = 0, \quad (2.98)$$

which follows since we choose to measure the parcel motion with respect to its center of mass. Hence, there is no subgrid scale flux for the full density ρ ; i.e., the mass conservation equation (2.90) is exact, even in the presence of subgrid scale processes.

As a final step in our development of mass conservation, introduce the concentration of a material constituent, defined by

$$\begin{aligned} C_n &= \frac{\text{mass of component } n}{\text{mass of seawater parcel}} \\ &= \frac{\rho_n}{\rho}. \end{aligned} \quad (2.99)$$

Substituting this *tracer concentration* into the constituent density equation (2.97) leads to the material form of the *tracer equation*

$$\rho \frac{dC_n}{dt} = -\nabla \cdot \mathbf{J}_n, \quad (2.100)$$

with the Eulerian form given by

$$\partial_t (\rho C_n) = -\nabla \cdot (\rho \mathbf{v} C_n + \mathbf{J}_n). \quad (2.101)$$

This is the Eulerian form of the tracer equation implemented in MOM. It applies to both the material tracers considered here, and the thermodynamical heat tracer described in Section 2.3.

2.3 Thermodynamical tracers

Heating and cooling of the ocean, as well as mass exchange, predominantly occur near the ocean surface. In contrast, transport in the interior is nearly adiabatic and isohaline. Hence, the surface ocean experiences irreversible processes that set characteristics of the water masses moving quasi-isentropically within the ocean interior. Useful labels for these water masses maintain their values when moving within the largely ideal ocean interior. Salinity is a good tracer for such purposes since it is altered predominantly by mixing between waters of varying concentrations, and the resulting salinity after homogenization of two water parcels is the mass weighted mean of the salinities of the individual parcels. These two properties are basic to the material tracers considered in Section 2.2. We discuss here desirable properties of a thermodynamic tracer that tags the heat within a water parcel and evolves analogously to material tracers. Much of this material follows from Chapter 5 of Griffies (2004).

¹⁰For an ocean model, whose grid spacing is far greater than that appropriate for molecular diffusion, the relative motion of a constituent is also affected by far larger subgrid scale processes, such as unresolved eddy advective and diffusive transport.

2.3.1 Potential temperature

Vertical adiabatic and isohaline motion in the ocean changes a fluid parcel's hydrostatic pressure, which thus causes its *in situ* temperature to change in proportion to the adiabatic lapse rate as given by

$$dT = \Gamma dp. \quad (2.102)$$

Consequently, *in situ* temperature is not a useful thermodynamic variable to label water parcels of common origin. Instead, it is more useful to remove the adiabatic pressure effects.

Removing adiabatic pressure effects from *in situ* temperature leads to the concept of *potential temperature*. Potential temperature is the *in situ* temperature that a water parcel of fixed composition would have if it were isentropically transported from its *in situ* pressure to a reference pressure p_r , with the reference pressure typically taken at the ocean surface. Mathematically, the potential temperature θ is the reference temperature obtained via integration of $dT = \Gamma dp$ for an isentropic *in situ* temperature change with respect to pressure (e.g., [Apel, 1987](#)):

$$T = \theta(s, T, p_r) + \int_{p_r}^p \Gamma(s, \theta, p') dp', \quad (2.103)$$

with Γ the lapse rate defined in terms of pressure changes. By definition, the *in situ* temperature T equals the potential temperature θ at the reference pressure $p = p_r$. Elsewhere, they differ by an amount determined by the adiabatic lapse rate. Beneath the diabatic surface mixed layer, a vertical profile of potential temperature is far more constant than *in situ* temperature.

As shown in Section 5.6.1 of [Griffies \(2004\)](#), the potential temperature of a parcel is constant when the parcel's specific entropy ζ and material composition are constant. Mathematically, this result follows by noting that when entropy changes at a fixed pressure and composition, $p = p_r$ so that temperature equals potential temperature. Equation (5.41) of [Griffies \(2004\)](#) then leads to

$$d\zeta = C_p d\ln\theta, \quad (2.104)$$

implying $d\zeta = 0$ if and only if $d\theta = 0$.

2.3.2 Potential enthalpy

Potential temperature has proven useful for many oceanographic purposes. However, we have yet to ask whether it is a convenient variable to mark the heat content in a parcel of seawater. Traditionally, it is the potential temperature multiplied by the heat capacity that is used for this purpose. [Bacon and Fofonoff \(1996\)](#) provide a review with suggestions for this approach. In contrast, [McDougall \(2003\)](#) argues that potential temperature multiplied by heat capacity is less precise, by some two orders of magnitude, than an alternative thermodynamic tracer called *potential enthalpy*.

To understand this issue from a mathematical perspective, consider the evolution equation for potential temperature

$$\rho \frac{d\theta}{dt} = -\nabla \cdot \mathbf{J}_\theta + \Sigma_\theta, \quad (2.105)$$

where \mathbf{J}_θ is a flux due to molecular diffusion, and Σ_θ is a source. That potential temperature evolves in this manner is ensured by its being a scalar field. Consider the mixing of two seawater parcels at the same pressure where the parcels have different potential temperature and salinity. In the absence of the source term, the equilibrated state consists of a single parcel with mass equal to the sum of the two separate masses, and potential temperature and salinity determined by their respective mass weighted means. Does this actually happen in the real ocean? That is, can source terms be ignored? [Fofonoff \(1962\)](#) and [McDougall \(2003\)](#) note that it is indeed the case for salinity (and any other material tracer due to conservation of matter), yet it is not the case for potential temperature. Instead, potential temperature contains source terms that alter the mass weighted average equilibrated state. In contrast, potential enthalpy (discussed below) maintains the desired conservative behavior when mixing at constant reference pressure, and nearly maintains this behavior if mixing parcels at a different pressure. Hence, ocean models which set the source term to zero upon mixing potential temperature are in error. [McDougall \(2003\)](#) quantifies this error.

Potential enthalpy is defined analogously to potential temperature. What motivates the use of potential enthalpy is the observation that the fundamental relation between thermodynamic state variables takes a nearly conservative form when written in terms of potential enthalpy. To see this point, consider the evolution of internal energy (see equation (5.94) in [Griffies, 2004](#)), and introduce the enthalpy per mass (specific enthalpy)

$$\mathcal{H} = \mathcal{I} + p/\rho \quad (2.106)$$

leads to

$$\rho \frac{d\mathcal{H}}{dt} = -\nabla \cdot \mathbf{J}_q + \frac{dp}{dt} + \rho \epsilon. \quad (2.107)$$

Dropping the frictional dissipation term arising from molecular friction leads to the approximate statement

$$\rho \frac{d\mathcal{H}}{dt} - \frac{dp}{dt} \approx -\nabla \cdot \mathbf{J}_q. \quad (2.108)$$

To proceed, the fundamental thermodynamic relation (see equation (5.31) [Griffies, 2004](#), in) becomes

$$d\mathcal{H} = T d\zeta + \rho^{-1} dp + \mu dC \quad (2.109)$$

in terms of enthalpy. Thus, enthalpy can be written as a function of entropy, salt concentration, and pressure,

$$\mathcal{H} = \mathcal{H}(\zeta, C, p). \quad (2.110)$$

Transport of a seawater parcel without changing heat, salt, or momentum occurs without change in entropy, thus rendering

$$\left(\frac{\partial \mathcal{H}}{\partial p} \right)_{\zeta, C} = \rho^{-1}. \quad (2.111)$$

Keeping salinity and entropy fixed (or equivalently fixed salinity and potential temperature) leads to

$$\mathcal{H}(\theta, s, p) = \mathcal{H}^o(\theta, s, p_r) + \int_{p_r}^p \rho^{-1}(\theta, s, p') dp' \quad (2.112)$$

with $\mathcal{H}^o(\theta, s, p_r)$ defining the potential enthalpy of a parcel with potential temperature θ and salinity s . Taking the time derivative and using the approximate relation (2.108) yields

$$\frac{d\mathcal{H}^o}{dt} \approx -\rho^{-1} \nabla \cdot \mathbf{J}_q + \int_p^{p_r} dp' \frac{d\rho^{-1}(\theta, s, p')}{dt}. \quad (2.113)$$

[McDougall \(2003\)](#) shows that for the ocean, the integral

$$\begin{aligned} \int_p^{p_r} dp' \frac{d\rho^{-1}(\theta, s, p')}{dt} &= \int_p^{p_r} dp' \left(\frac{\partial \rho^{-1}}{\partial \theta} \frac{d\theta}{dt} + \frac{\partial \rho^{-1}}{\partial s} \frac{ds}{dt} \right) \\ &= \frac{d\theta}{dt} \int_p^{p_r} dp' \rho^{-1} \alpha - \frac{ds}{dt} \int_p^{p_r} dp' \rho^{-1} \beta \end{aligned} \quad (2.114)$$

has magnitude on the order of the ocean's tiny levels of dissipation arising from molecular viscosity. These expressions introduced the thermal expansion coefficient $\alpha = -\partial \ln \rho / \partial \theta$ and saline contraction coefficient $\beta = \partial \ln \rho / \partial s$. The time derivatives of potential temperature and salinity can be removed from the pressure integrals, since they are each independent of pressure. Given the smallness of $\int_p^{p_r} dp' d\rho^{-1}/dt$, one can write the approximate potential enthalpy equation

$$\rho \frac{d\mathcal{H}^o}{dt} \approx -\nabla \cdot \mathbf{J}_q. \quad (2.115)$$

Hence, potential enthalpy is a state function that approximately specifies the heat in a parcel of seawater, and it evolves analogously to a material tracer such as salinity. See [McDougall \(2003\)](#) for a proof that \mathcal{H}^o more accurately sets the heat for a parcel of seawater than does $C_p \theta$. Given that it does, [McDougall \(2003\)](#) suggests that *conservative temperature*

$$\Theta \equiv \frac{\mathcal{H}^o(\theta, s, p_r)}{C_p^o} \quad (2.116)$$

with $p_r = 0$ is more appropriate than potential temperature as a thermodynamic tracer for use in an ocean model, and generally for measuring heat in the ocean. In this equation

$$\begin{aligned} C_p^o &= \frac{\mathcal{H}(\theta = 25^\circ\text{C}, s = 35\text{psu}, p_r = 0)}{25^\circ\text{C}} \\ &= 3989.245 \text{ J kg}^{-1} \text{ }^\circ\text{K}^{-1} \end{aligned} \quad (2.117)$$

is a heat capacity chosen to minimize the difference between $C_p^o \theta$ and potential enthalpy $\mathcal{H}^o(\theta, s, p_r)$ when averaged over the sea surface.¹¹

Conservative temperature of [McDougall \(2003\)](#) has been recommended by [IOC et al. \(2010\)](#) as the preferred means to measure heat content in a seawater parcel. MOM has the ability to use conservative temperature as its prognostic temperature field. Conservative temperature is the preferred method rather than the older potential temperature, with potential temperature retained for legacy purposes. In the remainder of these notes, we maintain the notation θ to mean conservative temperature, but with all formulas remaining unchanged if interpreted as potential temperature.

2.4 Material time changes over finite regions

In the following sections, we focus on the mass, tracer, and momentum budgets formulated over a finite domain. The domain, or control volume, of interest is that of an ocean model grid cell. The budget for a grid cell is distinct from budgets for infinitesimal mass conserving Lagrangian fluid parcels moving with the fluid. Mass conserving fluid parcels form the fundamental *system* for which the budgets of mass, tracer, momentum, and energy are generally formulated from first principles (see, for example, chapters 3-5 in [Griffies, 2004](#)). Grid cell budgets are then derived from the fundamental parcel budgets.

The grid cells of concern for MOM have vertical sides fixed in space-time, but with the top and bottom generally moving. In particular, the top and bottom either represent the ocean top, ocean bottom, or a surface of constant generalized vertical coordinate. We furthermore assume that at no place in the fluid do the top or bottom surfaces of the grid cell become vertical. This assumption allows for a one-to-one relation to exist between geopotential depth z and the generalized vertical coordinate s introduced in Section 2.1.5.2 (i.e., the relation is invertible).

To establish the grid cell budget, we integrate the budget for mass conserving fluid parcels over the volume of the cell. This section is focused on the mathematics required for integrating the density weighted material time derivative acting on an arbitrary field ψ

$$\rho \frac{d\psi}{dt} = (\rho \psi)_{,t} + \nabla \cdot (\rho \mathbf{v} \psi). \quad (2.118)$$

We start with the partial time derivative on the right hand side, and introduce Cartesian coordinates (x, y, z)

¹¹The value quoted by [McDougall \(2003\)](#) is $C_p^o = 3989.24495292815 \text{ J kg}^{-1} \text{ }^\circ\text{K}^{-1}$.

for the purpose of performing the grid cell integral

$$\begin{aligned}
 \iiint dV (\rho \psi)_{,t} &= \iiint dx dy dz (\rho \psi)_{,t} \\
 &= \iint dx dy \int_{z_1}^{z_2} dz (\rho \psi)_{,t} \\
 &= \iint dx dy \left[-(\rho \psi)_2 \partial_t z_2 + (\rho \psi)_1 \partial_t z_1 + \partial_t \left(\int_{z_1}^{z_2} dz (\rho \psi) \right) \right].
 \end{aligned} \tag{2.119}$$

The second equality follows by noting that the horizontal extent of a grid cell remains static, thus allowing for the horizontal integral to be brought outside of the time derivative. In contrast, the vertical extent has a time dependence, which necessitates the use of Leibniz's Rule. We now use equation (2.58)

$$z_{,t} = -s_{,t} z_{,s} \tag{2.120}$$

which relates time tendencies of the depth of a generalized surface to time tendencies of the surface itself. Equation (2.30) is next used to write

$$\begin{aligned}
 z_{,t} &= -s_{,t} z_{,s} \\
 &= z_{,s} |\nabla s| \hat{\mathbf{n}} \cdot \mathbf{v}^{(\text{ref})},
 \end{aligned} \tag{2.121}$$

in which we introduced the reference velocity $\mathbf{v}^{(\text{ref})}$ for a point sitting on the generalized surface. Finally, recall equation (2.32), which relates the area element on the surface to the horizontal projection $dA = dx dy$ of the surface

$$dA_{(\hat{\mathbf{n}})} = |z_{,s} \nabla s| dA. \tag{2.122}$$

Introducing this area then renders

$$z_{,t} dA = \hat{\mathbf{n}} \cdot \mathbf{v}^{(\text{ref})} dA_{(\hat{\mathbf{n}})}. \tag{2.123}$$

This equation relates the time tendency of the depth of the generalized surface to the normal component of the velocity at a point on the surface. The two are related through the ratio of the area elements. This result is now used for the top and bottom boundary terms in relation (2.119), yielding

$$\iiint dV (\rho \psi)_{,t} = \partial_t \left(\iiint \rho dV \psi \right) - \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{v}^{(\text{ref})} (\rho \psi). \tag{2.124}$$

Hence, the domain integrated Eulerian time tendency of the density weighted field equals the time tendency of the density weighted field integrated over the domain, minus an integral over the domain boundary associated with transport of material across that domain, with proper account taken for time dependence of the domain boundary.

The next step needed for volume integrating the density weighted material time derivative in equation (2.118) involves the divergence of the density weighted field

$$\iiint dV \nabla \cdot (\rho \mathbf{v} \psi) = \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{v} (\rho \psi), \tag{2.125}$$

which follows from Gauss' Law. Combining this result with equation (2.124) leads to the relation

$$\iiint \rho dV \frac{d\psi}{dt} = \partial_t \left(\iiint \rho dV \psi \right) + \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) (\rho \psi). \tag{2.126}$$

Hence, the mass weighted grid cell integral of the material time derivative of a field is given by the time derivative of the mass weighted field integrated over the domain, plus a boundary term that accounts for the transport across the domain boundaries, with allowance made for moving domain boundaries. The

manipulations leading to this result focused on an interior grid cell. The result, however, holds in general for a cell that abuts either the ocean surface or ocean bottom. For the ocean bottom, the boundary term vanishes since the bottom has a zero reference velocity, and there is no normal flow of fluid across the bottom. For the ocean surface, we employ relation (2.68) that defines the dia-surface transport of mass across the ocean surface in a manner analogous to the dia-surface transport (2.37) across an interior surface.

2.5 Basics of the finite volume method

The finite volume method for formulating the discrete equations of an ocean model has been incorporated to the ocean modelling literature only since the late 1990's. The work of [Adcroft et al. \(1997\)](#) is a canonical example of how this method can be used to garner a better representation of the solid earth boundary. In this section, we briefly outline the basis for this method. The interested reader may wish to look at chapter 6 of the book by [Hirsch \(1988\)](#), or the chapter by [Machenhauer et al. \(2009\)](#) for a more thorough introduction, or one of the growing number of monographs devoted exclusively to the method.

The general equations of fluid mechanics can be represented as conservation equations for scalar quantities (e.g., seawater mass and tracer mass) and vector quantities (e.g., linear momentum). As just detailed in Section 2.4, the conservation law for a scalar Ψ over an arbitrary fluid region can be put in the form

$$\partial_t \left(\iiint \Psi \, dV \right) = - \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{F} + \iiint \mathcal{S} \, dV. \quad (2.127)$$

The volume integral is taken over an arbitrary fluid region, and the area integral is taken over the bounding surface to that volume, with outward normal $\hat{\mathbf{n}}$. The flux \mathbf{F} penetrates the surface and acts to alter the scalar, whereas internal sources \mathcal{S} contribute to changes in the scalar throughout the interior of the domain. The budget for the vector linear momentum can be written in this form, with the addition of body forces that act similar to the source term written here (see Section 2.9). Fundamental to the finite volume method is that the fluxes contribute *only* at the boundary to the domain, and not within the interior as well. Hence, the domain can be subdivided into arbitrary shapes, with budgets over the subdivisions summing to recover the global budget.

A discrete finite volume analog to equation (2.127), for a region labeled with the integer J , takes the form

$$\partial_t (V_J \Psi_J) = - \sum_{\text{sides}} (A_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{F}) + V_J \mathcal{S}_J. \quad (2.128)$$

Quantities with the integer J subscript refer to the discrete analogs to the continuum fields and the geometric factors in equation (2.127). In particular, we define the discrete finite volume quantities

$$V_J \equiv \iiint dV \quad (2.129)$$

$$\Psi_J \equiv \frac{\iiint dV \Psi}{\iiint dV} \quad (2.130)$$

$$\mathcal{S}_J \equiv \frac{\iiint dV \mathcal{S}}{\iiint dV}. \quad (2.131)$$

Again, it is due to the conservation form of the fundamental fluid dynamic equation (2.127) that allows for a straightforward finite volume interpretation of the discrete equations. Notably, once formulated as such, the problem shifts from fundamentals to details, with details differing on how one represents the subgrid scale behaviour of the continuum fields. This shift leads to the multitude of discretization methods available for such processes as transport, time stepping, etc. In the following, we endeavour to write the fluid equations of the ocean in the conservation form (2.127). Doing so then renders a finite volume framework for the resulting discrete or semi-discrete equations.

When working with non-Boussinesq budgets, the finite volume interpretation applies directly to the tracer mass per volume, ρC , rather than to the tracer concentration C . The same applies to the linear

momentum per volume, $\rho \mathbf{v}$, rather than to the velocity \mathbf{v} . That is, the finite volume model carries the discrete fields ρ_J , $(\rho C)_J$ and $(\rho \mathbf{v})_J$, defined as

$$\rho_J \equiv \frac{\iiint dV \rho}{\iiint dV} \quad (2.132)$$

$$(\rho C)_J \equiv \frac{\iiint dV \rho C}{\iiint dV} \quad (2.133)$$

$$(\rho \mathbf{v})_J \equiv \frac{\iiint dV \rho \mathbf{v}}{\iiint dV}. \quad (2.134)$$

As we will see in the discussions in Sections 2.6 and 2.9, we actually work with a slightly modified finite volume suite of variables, whereby the finite volume interpretation applies to the seawater mass per horizontal area, the tracer mass per horizontal area and linear momentum per horizontal area

$$(dz\rho)_J \equiv \frac{\iint dA \int dz \rho}{\iint dA} \quad (2.135)$$

$$(dz\rho C)_J \equiv \frac{\iint dA \int dz \rho C}{\iint dA} \quad (2.136)$$

$$(dz\rho \mathbf{v})_J \equiv \frac{\iint dA \int dz \rho \mathbf{v}}{\iint dA}, \quad (2.137)$$

where dz is the thickness of a grid cell, and $dA = dx dy$ is the horizontal projection of its area. The inclusion of thickness facilitates the treatment of grid cells whose thickness is a function of time, such as in MOM. Note that to reduce notational clutter, we employ the same symbol for the continuum field as for the discrete, so we drop the J subscript in the following.

2.6 Mass and tracer budgets over finite regions

The purpose of this section is to extend the kinematics discussed in the previous sections to the case of mass and tracer budgets for finite domains within the ocean fluid. In the formulation of ocean models, these domains are thought of as discrete model grid cells.

2.6.1 General formulation

As described in Section 2.2, the tracer concentration C represents a mass of tracer per mass of seawater for *material tracers* such as salt or biogeochemical tracers. Mathematically, this definition means that for each fluid parcel,

$$\begin{aligned} C &= \frac{\text{mass of tracer}}{\text{mass of seawater}} \\ &= \frac{\rho_C dV}{\rho dV}, \end{aligned} \quad (2.138)$$

where ρ_C is the mass density of tracer within the fluid parcel. In addition to material tracers, we are concerned with a thermodynamical tracer that measures the heat within a fluid parcel. In this case, C is typically taken to be the potential temperature. However, the work of McDougall (2003) prompts us to consider a modified temperature known as *conservative temperature*, which more accurately measures the heat within a fluid parcel *and* is transported, to within a very good approximation, in a manner directly analogous to material tracers. We discussed these temperature variables in Section 2.3.

Given these considerations, the total tracer mass within a finite region of seawater is given by the integral.

$$\begin{aligned} \text{tracer mass in a region} &= \iiint \rho_C dV \\ &= \iiint C \rho dV. \end{aligned} \quad (2.139)$$

Correspondingly, the evolution of tracer mass within a Lagrangian parcel of mass conserving fluid is given by (see Section 5.1 of [Griffies, 2004](#))

$$\rho \frac{dC}{dt} = -\nabla \cdot \mathbf{J} + \rho S^{(C)}, \quad (2.140)$$

where $S^{(C)}$ is a tracer source in the region, with units of tracer concentration per time. The tracer flux \mathbf{J} arises from subgrid scale transport of tracer in the absence of mass transport. Such transport in MOM consists of diffusion and/or unresolved advection. As discussed in Section 2.2.3, this flux is computed with respect to the center of mass of a fluid parcel. It therefore vanishes when the tracer concentration is uniform, in which case the tracer budget reduces to the mass budget (2.7).

We now develop a regional budget for tracer mass over a grid cell. For this purpose, apply the general result (2.126) relating the material time derivative to a regional budget, to render

$$\partial_t \left(\iiint C \rho dV \right) = \iiint S^{(C)} \rho dV - \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v} - \mathbf{v}^{\text{ref}}) \rho C + \mathbf{J}]. \quad (2.141)$$

Again, the left hand side of this equation is the time tendency for tracer mass within the finite sized grid cell region. When the tracer concentration is uniform, the SGS flux vanishes, in which case the tracer budget (2.141) reduces to the finite domain mass budget

$$\partial_t \left(\iiint \rho dV \right) = \iiint S^{(M)} \rho dV - \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v} - \mathbf{v}^{\text{ref}}) \rho]. \quad (2.142)$$

In addition to the tracer flux \mathbf{J} , it is convenient to define the *tracer concentration flux* \mathbf{F} via

$$\mathbf{J} = \rho \mathbf{F}, \quad (2.143)$$

where the dimensions of \mathbf{F} are velocity \times tracer concentration.

In a manner analogous to our definition of a dia-surface velocity component in Section 2.1.4, it is useful to introduce the dia-surface SGS flux component. For this purpose, consider the tracer mass per time crossing a surface of constant generalized vertical coordinate, where this transport arises from SGS processes. Manipulations similar to those used to derive the dia-surface velocity component lead to

$$\begin{aligned} (\text{SGS tracer mass through surface})/(\text{time}) &= dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{J} \\ &= z_{,s} \nabla s \cdot \mathbf{J} dx dy \\ &= (\hat{\mathbf{z}} - \mathbf{S}) \cdot \mathbf{J} dx dy, \end{aligned} \quad (2.144)$$

where \mathbf{S} is the slope vector for the generalized surface defined in equation (2.39). We are therefore led to introduce the dia-surface SGS tracer flux

$$\begin{aligned} J^{(z)} &\equiv \frac{dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{J}}{dA} \\ &= z_{,s} \nabla s \cdot \mathbf{J} \\ &= (\hat{\mathbf{z}} - \mathbf{S}) \cdot \mathbf{J}, \end{aligned} \quad (2.145)$$

where $dA = dx dy$ is the horizontal cross-sectional area. In words, $J^{(z)}$ is the tracer mass per time per horizontal area penetrating surfaces of constant generalized vertical coordinate via processes that are unresolved by the dia-surface velocity component $w^{(z)}$.

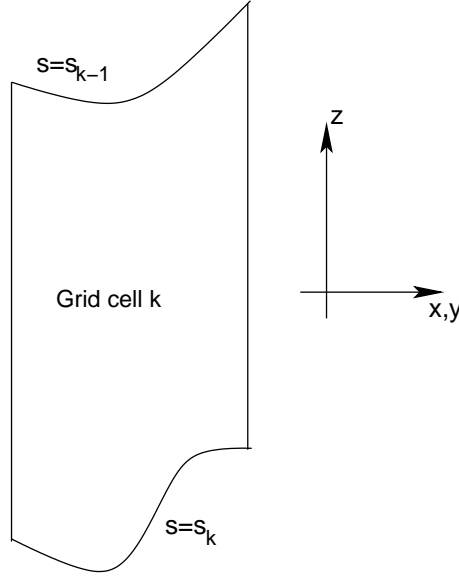


Figure 2.4: Schematic of an ocean grid cell labeled by the vertical integer k . Its sides are vertical and oriented according to \hat{x} and \hat{y} , and its horizontal position is fixed in time. The top and bottom surfaces are determined by constant generalized vertical coordinates s_{k-1} and s_k , respectively. Furthermore, the top and bottom are assumed to always have an outward normal with a nonzero component in the vertical direction \hat{z} . That is, the top and bottom are never vertical. We take the convention that the discrete vertical label k increases as moving downward in the column, and grid cell k is bounded at its upper face by $s = s_{k-1}$ and lower face by $s = s_k$.

2.6.2 Budget for an interior grid cell

Consider the budget for a region bounded away from the ocean surface and bottom, such as that shown in Figure 2.4. We have in mind here a grid cell within a discrete numerical model. There are two assumptions that define a grid cell for our purposes.

- The sides of the cell are vertical, so they are parallel to \hat{z} and aligned with the horizontal coordinate directions (\hat{x}, \hat{y}) . Their horizontal positions are fixed in time.
- The top and bottom of the cell are defined by surfaces of constant generalized vertical coordinate $s = s(x, y, z, t)$. The generalized surfaces do not overturn, which means that $s_{,z}$ is single signed throughout the ocean.

These assumptions lead to the following results for the sides of the grid cell

$$\text{TRACER MASS ENTERING CELL WEST FACE} = \iint_{x=x_1} dy dz (u \rho C + \rho F^x) \quad (2.146)$$

$$\text{TRACER MASS LEAVING CELL EAST FACE} = - \iint_{x=x_2} dy dz (u \rho C + \rho F^x) \quad (2.147)$$

where $x_1 \leq x \leq x_2$ defines the domain boundaries for the east-west coordinates.¹² Similar results hold for the tracer mass crossing the cell in the north-south directions. At the top and bottom of the grid cell

$$\text{TRACER MASS ENTERING CELL BOTTOM FACE} = \iint_{s=s_k} dx dy \rho(w^{(z)} C + F^{(z)}) \quad (2.148)$$

$$\text{TRACER MASS LEAVING CELL TOP FACE} = - \iint_{s=s_{k-1}} dx dy \rho(w^{(z)} C + F^{(z)}). \quad (2.149)$$

To reach this result, we used a result from Section 2.1.4 to write the volume flux passing through the top face of the grid cell

$$dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{\text{ref}}) = w^{(z)} dx dy, \quad (2.150)$$

with $w^{(z)} = z_{,s} ds/dt$ the dia-surface velocity component from Section 2.1.4. A similar relation holds for the bottom face of the cell. The form of the SGS flux passing across the top and bottom is correspondingly given by

$$dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{J} = J^{(z)} dx dy, \quad (2.151)$$

which follows from the general expression (2.145) for the dia-surface tracer flux.

In a model using the generalized coordinate s for the vertical, it is sometimes convenient to do the vertical integrals over s instead of z . For this purpose, recall that with $z_{,s}$ single signed, the vertical thickness of a grid cell is given by equation (2.52), repeated here for completeness

$$dz = z_{,s} ds. \quad (2.152)$$

Bringing these results together, and taking the limit as the volume of the cell in (x, y, s) space goes to zero (i.e., $dx dy ds \rightarrow 0$) leads to

$$\partial_t(z_{,s} \rho C) = z_{,s} \rho \mathcal{S}^{(C)} - \nabla_s \cdot [z_{,s} \rho (\mathbf{u} C + \mathbf{F})] - \partial_s [\rho (w^{(z)} C + F^{(z)})] \quad (2.153)$$

Notably, the horizontal gradient operator ∇_s is computed on surfaces of constant s , and so it is distinct generally from the horizontal gradient ∇_z taken on surfaces of constant z .

As indicated at the end of Section 2.5, we prefer to work with thickness weighted quantities, given the general time dependence of a model grid cell in MOM. Hence, as an alternative to taking the limit as $dx dy ds \rightarrow 0$, consider instead the limit as the time independent horizontal area $dx dy$ goes to zero, thus maintaining the time dependent thickness $dz = z_{,s} ds$ inside the derivative operators. In this case, the thickness weighted tracer mass budget takes the form

$$\partial_t(dz \rho C) = dz \rho \mathcal{S}^{(C)} - \nabla_s \cdot [dz \rho (\mathbf{u} C + \mathbf{F})] - [\rho (w^{(z)} C + F^{(z)})]_{s=s_{k-1}} + [\rho (w^{(z)} C + F^{(z)})]_{s=s_k}. \quad (2.154)$$

Similarly, the thickness weighted mass budget is

$$\partial_t(dz \rho) = dz \rho \mathcal{S}^{(M)} - \nabla_s \cdot (dz \rho \mathbf{u}) - (\rho w^{(z)})_{s=s_{k-1}} + (\rho w^{(z)})_{s=s_k}. \quad (2.155)$$

For clarity, note that the horizontal divergence operator acting on the mass transport takes the form

$$\nabla_s \cdot (dz \rho \mathbf{u}) = \frac{1}{dy} \frac{\partial}{\partial x} (dy dz \rho u) + \frac{1}{dx} \frac{\partial}{\partial y} (dx dz \rho v). \quad (2.156)$$

The mass source $\mathcal{S}^{(M)}$ has units of inverse time that, for self-consistency, must be related to the tracer source via

$$\mathcal{S}^{(M)} = \mathcal{S}^{(C)} (C = 1). \quad (2.157)$$

¹²We use generalized horizontal coordinates, such as those discussed in Griffies (2004). Hence, the directions east, west, north, and south may not correspond to the usual geographic directions. Nonetheless, this terminology is useful for establishing the budgets, whose validity is general.

Additionally, the SGS tracer flux vanishes with a uniform tracer

$$\mathbf{F}(C = 1) = 0. \quad (2.158)$$

Note that by setting the tracer concentration in equation (2.154) to a uniform constant, SGS transport fluxes vanish, thus revealing the mass conservation budget. This procedure for deriving the mass budget from the tracer budget follows trivially from the definition of the tracer concentration given by equation (2.138). It represents a *compatibility condition* between the discrete budgets, and this condition is critical to maintain within a numerical model in order to respect tracer and mass conservation in the simulation. We have more to say about the compatibility condition in Section 2.7.1.

One reason that the thickness weighted budget given by equation (2.154) is more convenient than equation (2.153) is that equation (2.154) expresses the budget in terms of the grid cell thickness dz , rather than the specific thickness z_s . Nonetheless, this point is largely one of style and convenience, as there is no fundamental reason to prefer one form over the other for purposes of developing the discrete equations of an ocean model.

2.6.3 Cells adjacent to the ocean bottom

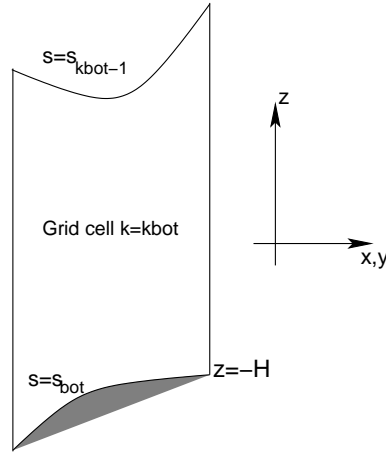


Figure 2.5: Schematic of an ocean grid cell next to the ocean bottom labeled by $k = k_{\text{bot}}$. Its top face is a surface of constant generalized vertical coordinate $s = s_{k_{\text{bot}}-1}$, and the bottom face is determined by the ocean bottom topography at $z = -H$ where $s_{\text{bot}}(x, y, t) = s(x, y, z = -H, t)$.

For a grid cell adjacent to the ocean bottom (Figure 2.5), we assume that just the bottom face of this cell abuts the solid earth boundary. The outward normal $\hat{\mathbf{n}}_H$ to the bottom is given by equation (2.46), and the area element along the bottom is

$$dA_H = |\nabla(z + H)| dx dy. \quad (2.159)$$

Hence, the transport across the solid earth boundary is

$$- \iint dA_H \hat{\mathbf{n}}_H \cdot (\mathbf{v} \rho C + \mathbf{J}) = \iint dx dy (\nabla H + \hat{\mathbf{z}}) \cdot (\mathbf{v} \rho C + \mathbf{J}). \quad (2.160)$$

We assume that there is zero advective mass flux across the bottom, in which case the advective flux drops out since $\mathbf{v} \cdot (\nabla H + \hat{\mathbf{z}}) = 0$ (equation (2.48)). However, the possibility of a nonzero geothermal tracer transport warrants a nonzero SGS tracer flux at the bottom, in which case the bottom tracer flux is written

$$Q_{(\text{bot})}^{(C)} = (\nabla H + \hat{\mathbf{z}}) \cdot \mathbf{J}. \quad (2.161)$$

The corresponding thickness weighted budget is given by

$$\partial_t (dz \rho C) = dz \rho \mathcal{S}^{(C)} - \nabla_s \cdot [dz \rho (\mathbf{u} C + \mathbf{F})] - \left[\rho (w^{(z)} C + z_{,s} \nabla s \cdot \mathbf{F}) \right]_{s=s_{k_{\text{bot}}-1}} + Q_{(\text{bot})}^{(C)}, \quad (2.162)$$

and the corresponding mass budget is

$$\partial_t (dz \rho) = dz \rho S^{(M)} - \nabla_s \cdot (dz \rho \mathbf{u}) - (\rho w^s)_{s=s_{kbot-1}} + Q_{(bot)}^{(M)}, \quad (2.163)$$

where $Q_{(bot)}^{(M)}$ allows for the possibility of mass entering through geothermal boundary sources. For brevity, we drop this term in the following, since it generally is ignored for ocean climate modeling.

2.6.4 Cells adjacent to the ocean surface

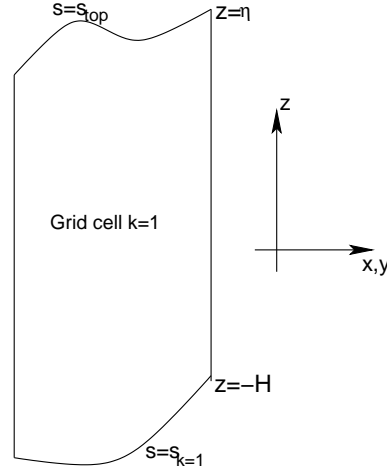


Figure 2.6: Schematic of an ocean grid cell next to the ocean surface labeled by $k = 1$. Its top face is at $z = \eta$, and the bottom is a surface of constant generalized vertical coordinate $s = s_{k=1}$.

For a grid cell adjacent to the ocean surface (Figure 2.6), we assume that just the upper face of this cell abuts the boundary between the ocean and the atmosphere or sea ice. The ocean surface is a time dependent boundary with $z = \eta(x, y, t)$. The outward normal $\hat{\mathbf{n}}_\eta$ is given by equation (2.69), and its area element dA_η is given by equation (2.73).

As the surface can move, we must measure the advective transport with respect to the moving surface. Just as in the dia-surface transport discussed in Section 2.1.4, we consider the velocity of a reference point on the surface

$$\mathbf{v}^{\text{ref}} = \mathbf{u}^{\text{ref}} + \hat{\mathbf{z}} w^{\text{ref}}. \quad (2.164)$$

Since $z = \eta$ represents the vertical position of the reference point, the vertical component of the velocity for this point is given by

$$w^{\text{ref}} = (\partial_t + \mathbf{u}^{\text{ref}} \cdot \nabla) \eta \quad (2.165)$$

which then leads to

$$\mathbf{v}^{\text{ref}} \cdot \nabla (z - \eta) = \eta_{,t}. \quad (2.166)$$

Hence, the advective transport leaving the ocean surface is

$$\begin{aligned} \iint_{z=\eta} dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{\text{ref}}) \rho C &= \iint_{z=\eta} dx dy (-\eta_{,t} + w - \mathbf{u} \cdot \nabla \eta) \rho C \\ &= - \iint_{z=\eta} dx dy Q_m C, \end{aligned} \quad (2.167)$$

where the surface kinematic boundary condition (2.77) was used. The negative sign on the right hand side arises from our convention that $Q_m > 0$ represents an input of mass to the ocean domain. In summary, the

tracer flux leaving the ocean free surface is given by

$$\iint_{z=\eta} dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v} - \mathbf{v}^{\text{ref}}) \rho C + \mathbf{J}] = \iint_{z=\eta} dx dy (-Q_m C + \nabla(z - \eta) \cdot \mathbf{J}). \quad (2.168)$$

In equation (2.168), we formally require the tracer concentration precisely at the ocean surface $z = \eta$. However, as mentioned at the start of Section 2.1.6, it is actually a fiction that the ocean surface is a smooth mathematical function. Furthermore, seawater properties precisely at the ocean surface, known generally as *skin properties*, are generally not what an ocean model carries as its prognostic variable in its top grid cell. Instead, the model carries a bulk property averaged over roughly the upper few tens of centimeters.

To proceed in formulating the boundary condition for an ocean climate model, we consider there to be a boundary layer model that provides us with the total tracer flux passing through the ocean surface. Developing such a model is a nontrivial problem in air-sea and ice-sea interaction theory and phenomenology. For present purposes, we do not focus on these details, and instead just introduce this flux in the form

$$Q^{(C)} = -Q_m C_m + Q_{(\text{turb})}^{(C)} \quad (2.169)$$

where C_m is the tracer concentration within the incoming water. The first term on the right hand side represents the advective transport of tracer through the surface with the fresh water (i.e., ice melt, rivers, precipitation, evaporation). The term $Q_{(\text{turb})}^{(C)}$ arises from parameterized turbulence and/or radiative fluxes, such as sensible, latent, shortwave, and longwave heating appropriate for the temperature equation. A positive value for $Q_{(\text{turb})}^{(C)}$ signals tracer leaving the ocean through its surface. In the special case of zero fresh water flux, then

$$\nabla(z - \eta) \cdot \mathbf{J} = Q_{(\text{turb})}^{(C)} \quad \text{if } Q_m = 0. \quad (2.170)$$

In general, it is not possible to make this identification. Instead, we must settle for the general expression

$$\iint_{z=\eta} dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v} - \mathbf{v}^{\text{ref}}) \rho C + \mathbf{J}] = \iint_{z=\eta} dx dy (-Q_m C_m + Q_{(\text{turb})}^{(C)}). \quad (2.171)$$

The above results lead to the thickness weighted tracer budget for the ocean surface grid cell

$$\begin{aligned} \partial_t (dz \rho C) &= dz \rho \mathcal{S}^{(C)} - \nabla_s \cdot [dz \rho (\mathbf{u} C + \mathbf{F})] \\ &\quad + \left[\rho (w^{(z)} C + z_{,s} \nabla s \cdot \mathbf{F}) \right]_{s=s_{k=1}} + (Q_m C_m - Q_{(C)}^{(\text{turb})}), \end{aligned} \quad (2.172)$$

and the corresponding mass budget

$$\partial_t (dz \rho) = dz \rho \mathcal{S}^{(M)} - \nabla_s \cdot (dz \rho \mathbf{u}) + (\rho w^{(z)})_{s=s_{k=1}} + Q_m. \quad (2.173)$$

2.7 Special considerations for tracers

The purpose of this section is to describe some special considerations for tracers in a numerical ocean model.

2.7.1 Compatibility between vertically integrated mass and tracer budgets

In Section 2.6.2, we considered issues of compatibility between the tracer and mass budgets within a grid cell. Such compatibility follows trivially from the definition of tracer concentration given in Section 2.6.1. We briefly revisit compatibility here, by focusing on the vertically integrated tracer and mass budgets.

Combining the surface tracer budget (2.173), the bottom budget (2.162), and interior budget (2.154), renders the vertically integrated tracer budget

$$\begin{aligned} \partial_t \left(\sum_k dz \rho C \right) = & \sum_k dz \rho \mathcal{S}^{(C)} - \nabla_s \cdot \left(\sum_k dz \rho (\mathbf{u} C + \mathbf{F}) \right) \\ & + \left(Q_m C_m - Q_{(C)}^{(\text{turb})} + Q_{(C)}^{(\text{bott})} \right). \end{aligned} \quad (2.174)$$

As expected, the only contributions from vertical fluxes come from the top and bottom boundaries. Furthermore, by setting the tracer concentration to a uniform constant, in which case the SGS turbulent terms vanish, the tracer budget reduces to the vertically integrated mass budget

$$\partial_t \left(\sum_k dz \rho \right) = \sum_k dz \rho \mathcal{S}^{(M)} - \nabla_s \cdot \mathbf{U}^\rho + Q_m, \quad (2.175)$$

where

$$\mathbf{U}^\rho = \sum_k dz \rho \mathbf{u} \quad (2.176)$$

is the discrete form of the vertically integrated horizontal momentum per volume defined by equation (2.22). As for the individual grid cells, this vertically integrated compatibility between tracer and mass budgets must be carefully maintained by the space and time discretizations used in an ocean model. Otherwise, conservation properties of the model will be compromised (Griffies et al., 2001).

2.7.2 Fresh water budget

Seawater is comprised of freshwater with a relatively fixed ratio of various salts. It is common to consider the budget for the concentration of these salts, which is described by the tracer equation (2.154). As a complement, it may be of interest to formulate a budget for freshwater. In this case, we consider the mass of fresh water within a fluid parcel

$$\begin{aligned} \text{mass of fresh water} &= \text{mass of seawater} - \text{mass of salt} \\ &= \rho dV (1 - S) \\ &= \rho dV W, \end{aligned} \quad (2.177)$$

where S is the salinity (mass of salt per mass of seawater), and

$$W \equiv 1 - S \quad (2.178)$$

is the mass of fresh water per mass of seawater. Results from the tracer budget considered in Section 2.6.2 allow us to derive the following budget for fresh water within an interior ocean model grid cell

$$\begin{aligned} \partial_t (dz \rho W) = & dz \rho (\mathcal{S}^{(M)} - \mathcal{S}^{(S)}) - \nabla_s \cdot [dz \rho (\mathbf{u} W - \mathbf{F})] \\ & - [\rho (w^{(z)} W - F^{(z)})]_{s=s_{k-1}} + [\rho (w^{(z)} W - F^{(z)})]_{s=s_k}. \end{aligned} \quad (2.179)$$

In these relations, the SGS tracer flux components \mathbf{F} and $F^{(z)}$ are those for salt, and $\mathcal{S}^{(S)}$ is the salt source. Equation (2.179) is very similar to the tracer equation (2.154), with modified source term and negative signs on the SGS flux components.

2.7.3 The ideal age tracer

Thiele and Sarmiento (1990) and England (1995) consider an ideal age tracer for Boussinesq fluids. We consider the generalization here to non-Boussinesq fluids, in which

$$\rho \frac{dA}{dt} + \nabla \cdot \mathbf{J} = \rho \mathcal{S}^{(A)}, \quad (2.180)$$

where the age tracer A has dimensions of time and it is initialized globally to zero. It is characterized by the dimensionless *clock* source $\mathcal{S}^{(A)}$, which takes the values

$$\mathcal{S}^{(A)} = \begin{cases} 0 & \text{if } z = \eta \\ 1 & \text{if } z < \eta, \end{cases} \quad (2.181)$$

In a finite difference model, the boundary condition at $z = \eta$ is applied at the top grid cell $k = 1$. In MOM, various age tracers can be defined that differ by the region that their boundary condition is set to zero. Given these prescriptions, A measures the age, in units of time, that a water parcel has spent away from the region where it was set to zero. Therefore, visual maps of A are useful to deduce such physically interesting properties as ventilation times.

From equation (2.154), the budget for tracer mass per area in a grid cell is given by

$$\begin{aligned} \partial_t(dz\rho A) = & dz\rho \mathcal{S}^{(A)} - \nabla_s \cdot [dz\rho(\mathbf{u}A + \mathbf{F})] \\ & - [\rho(w^{(z)}A + F^{(z)})]_{s=s_{k-1}} + [\rho(w^{(z)}A + F^{(z)})]_{s=s_k}. \end{aligned} \quad (2.182)$$

In practice, the clock source is added to the age tracer at the very end of the time step, so that it is implemented as an adjustment process. In this way, we remove the ambiguity regarding the time step to evaluate the ρdz factor that multiplies the age source.

2.7.4 Budgets without dia-surface fluxes

To garner some experience with tracer budgets, it is useful to consider the special case of zero dia-surface transport, either via advection or SGS fluxes, and zero tracer/mass sources. In this case, the thickness weighted mass and tracer mass budgets take the simplified form

$$\partial_t(dz\rho) = -\nabla_s \cdot (dz\rho\mathbf{u}) \quad (2.183)$$

$$\partial_t(dz\rho C) = -\nabla_s \cdot [dz\rho(\mathbf{u}C + \mathbf{F})]. \quad (2.184)$$

The first equation says that the time tendency of the thickness weighted density (mass per area) at a point between two surfaces of constant generalized vertical coordinate is given by the horizontal convergence of mass per area onto that point. The transport is quasi-two-dimensional in the sense that it is only a two-dimensional convergence that determines the evolution. The tracer equation has an analogous interpretation. We illustrate this situation in Figure 2.7. As emphasized in our discussion of the material time derivative (2.43), this simplification of the transport equation does not mean that fluid parcels are strictly horizontal. Indeed, such is distinctly not the case when the surfaces are moving.

A further simplification of the mass and tracer mass budgets ensues when considering adiabatic and Boussinesq flow in isopycnal coordinates. We consider ρ now to represent the constant potential density of the finitely thick fluid layer. In this case, the mass and tracer budgets reduce to

$$\partial_t(dz) = -\nabla_\rho \cdot (dz\mathbf{u}) \quad (2.185)$$

$$\partial_t(dzC) = -\nabla_\rho \cdot [dz(\mathbf{u}C + \mathbf{F})]. \quad (2.186)$$

Equation (2.185) provides a relation for the thickness of the density layers, and equation (2.186) is the analogous relation for the tracer within the layer. These expressions are commonly used in the construction of adiabatic isopycnal models, which are often used in the study of geophysical fluid mechanics of the ocean.

2.8 Forces from pressure

Pressure is a contact force per area that acts in a compressive manner on the boundary of a finite fluid domain (e.g., see Figure 2.8). Mathematically, we have

$$\mathbf{F}_{\text{press}} = - \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} p, \quad (2.187)$$

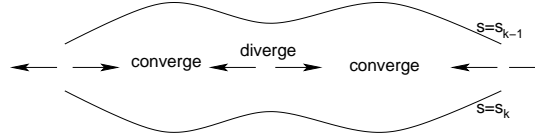


Figure 2.7: Schematic of the horizontal convergence of mass between two surfaces of constant generalized vertical coordinates. As indicated by equation (2.183), when there is zero dia-surface transport, it is just the horizontal convergence that determines the time evolution of mass between the layers. Evolution of thickness weighted tracer concentration in between the layers is likewise evolved just by the horizontal convergence of the thickness weighted advective and diffusive tracer fluxes (equation (2.184)). In this way, the transport is quasi-two-dimensional when the dia-surface transports vanish. A common example of this special system is an adiabatic ocean where the generalized surfaces are defined by isopycnals.

where p is the pressure (with units of a force per area) acting on the boundary of the domain with outward normal \hat{n} and area element $dA_{(\hat{n})}$. The minus sign accounts for the compressive behaviour of pressure. The accumulation of contact pressure forces acting over the bounding area of the domain leads to a net pressure force acting on the domain.

Through use of the Green-Gauss theorem of vector calculus, we can equivalently consider pressure to exert a body force per area at each point within the domain, so that

$$\mathbf{F}_{\text{press}} = - \iiint dV \nabla p, \quad (2.188)$$

where dV is the volume element. That is, the volume integral of the pressure gradient body force over the domain yields the net pressure force.

In the continuum, the two formulations (2.187) and (2.188) yield identical pressure forces. Likewise, in a finite volume discretization, the two forms are identical (e.g., Section 6.2.2 of Hirsch, 1988). But with finite differences, as used in earlier versions of MOM for pressure forces, the two forms can lead to different numerical methods. In the remainder of this section, we further explore the computation of pressure forces according to the two different formulations. Further details of discrete expressions are presented in Chapter 3.

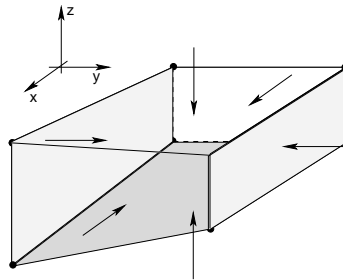


Figure 2.8: Schematic of a grid cell bounded at its top and bottom in general by sloped surfaces and vertical side walls. The top and bottom surfaces can represent linear piecewise approximations to surfaces of constant generalized vertical coordinates, with $s = s_1$ at the top surface and $s = s_2$ at the bottom surface. They could also represent the ocean surface (for the top face) or the ocean bottom (for the bottom face). The arrows represent the pressure contact forces that act in a compressive manner along the boundaries of the grid cell and in a direction normal to the boundaries. These forces arise from contact between the shown fluid volume and adjacent regions. Due to Newton's Third Law, the pressure acting on an arbitrary fluid parcel A due to contact with a parcel B is equal and opposite to the pressure acting on parcel B due to contact with parcel A. If coded according to finite volume budgets, as in Lin (1997) or Adcroft et al. (2008), this law extends to the pressure forces acting between grid cells in an ocean model.

2.8.1 The accumulation of contact pressure forces

Pressure acts as a contact or interfacial stress on the sides of a finite region of fluid. In particular, the total pressure force acting on the grid cell in Figure 2.8 is given by summing the pressure forces acting on the six cell faces

$$\mathbf{F}_{\text{pressure}} = \mathbf{F}_{x=x_1} + \mathbf{F}_{x=x_2} + \mathbf{F}_{y=y_1} + \mathbf{F}_{y=y_2} + \mathbf{F}_{s=s_1} + \mathbf{F}_{s=s_2}. \quad (2.189)$$

The pressure acting on faces with a zonal normal can be written

$$\mathbf{F}_{x=x_1} = \hat{\mathbf{x}} \left(\int dy \int_{z_2}^{z_1} dz p \right)_{x=x_1} \quad (2.190)$$

$$\mathbf{F}_{x=x_2} = -\hat{\mathbf{x}} \left(\int dy \int_{z_2}^{z_1} dz p \right)_{x=x_2} \quad (2.191)$$

where the vertical integral extends from the bottom face at $z_2 = z(x, y, s = s_2, t)$ to the top face at $z_1 = z(x, y, s = s_1, t)$. Likewise, the meridional pressure forces are

$$\mathbf{F}_{y=y_1} = \hat{\mathbf{y}} \left(\int dx \int_{z_2}^{z_1} dz p \right)_{y=y_1} \quad (2.192)$$

$$\mathbf{F}_{y=y_2} = -\hat{\mathbf{y}} \left(\int dx \int_{z_2}^{z_1} dz p \right)_{y=y_2}. \quad (2.193)$$

On the top face, the pressure force is given by

$$\begin{aligned} \mathbf{F}_{s=s_1} &= - \left(\int dy \int dx p z_{,s} \nabla s \right)_{s=s_1} \\ &= - \left(\int dy \int dx p (-\nabla_s z + \hat{\mathbf{z}}) \right)_{s=s_1}. \end{aligned} \quad (2.194)$$

Note the contribution from the generally non-horizontal top face as represented by the two dimensional vector

$$\nabla_s z = \mathbf{S}, \quad (2.195)$$

which is the slope of the surface of constant generalized vertical coordinate relative to the horizontal plane. The pressure force on the bottom face has a similar appearance

$$\mathbf{F}_{s=s_2} = \left(\int dy \int dx p (-\nabla_s z + \hat{\mathbf{z}}) \right)_{s=s_2}. \quad (2.196)$$

If the top and bottom faces are horizontal, as for z -models, the pressure force acting at $s = s_1$ and $s = s_2$ acts solely in the vertical direction. More generally, the pressure force per area on the top and bottom faces is oriented according to the slope of the faces and so has a nontrivial projection into all three directions.

To garner a sense for how pressure acts on the face of a grid cell, consider the case where the top surface of a grid cell rises to the east as shown in Figure 2.9. In this case, the pressure force per area in the $x-z$ plane takes the form

$$\text{PRESSURE FORCE PER AREA ON TOP FACE} = -p [\hat{\mathbf{z}} - (\partial z / \partial x)_s \hat{\mathbf{x}}]. \quad (2.197)$$

Since $(\partial z / \partial x)_s > 0$ for this example, the pressure force per area has a positive component in the $\hat{\mathbf{x}}$ direction, as indicated by the arrow normal to the top surface in Figure 2.9.

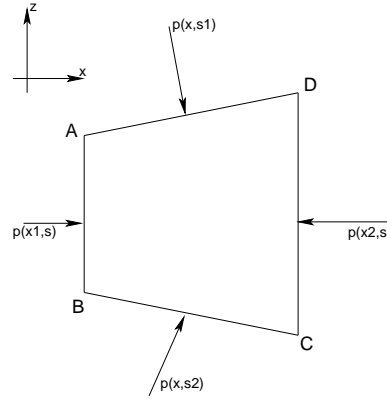


Figure 2.9: The sides of the grid cell, with the slopes top and bottom surfaces more enhanced here than in Figure 2.9. The corners are denoted A, B, C, and D, and oriented in a counterclockwise manner. This is the orientation appropriate for performing a contour integral in order to compute the pressure force acting on the area.

When the top surface represents the surface of the ocean at $z = \eta$, the pressure p is the applied pressure p_a arising from any media above the ocean, such as the atmosphere and sea ice. In this case,

$$\begin{aligned} \text{PRESSURE FORCE PER AREA ON OCEAN SURFACE} &= -p_a \nabla(z - \eta) \\ &= -p_a (\hat{\mathbf{z}} - \nabla\eta), \end{aligned} \quad (2.198)$$

where $\nabla\eta$ is the slope of the ocean surface. Likewise, if the bottom of the grid cell is bounded by the solid earth boundary,

$$\begin{aligned} \text{PRESSURE FORCE PER AREA ON OCEAN BOTTOM} &= p_b \nabla(z + H) \\ &= p_b (\hat{\mathbf{z}} + \nabla H), \end{aligned} \quad (2.199)$$

where ∇H is the bottom slope.

A sum of the pressure forces acting on the six faces of the grid cell determines the acceleration due to pressure acting on a grid cell. Organizing the forces into the three directions leads to

$$F_{\text{pressure}}^x = \left(\int dy \int_{z_2}^{z_1} dz p \right)_{x=x_1} - \left(\int dy \int_{z_2}^{z_1} dz p \right)_{x=x_2} \quad (2.200)$$

$$+ \left(\int dy \int_{x_1}^{x_2} dx z_{,x} p \right)_{s=s_1} - \left(\int dy \int_{x_1}^{x_2} dx z_{,x} p \right)_{s=s_2} \quad (2.201)$$

$$F_{\text{pressure}}^y = \left(\int dx \int_{z_2}^{z_1} dz p \right)_{y=y_1} - \left(\int dx \int_{z_2}^{z_1} dz p \right)_{y=y_2} \quad (2.202)$$

$$+ \left(\int dx \int_{y_1}^{y_2} dy z_{,y} p \right)_{s=s_1} - \left(\int dx \int_{y_1}^{y_2} dy z_{,y} p \right)_{s=s_2} \quad (2.203)$$

$$F_{\text{pressure}}^z = \left(\iint dx dy p \right)_{s=s_2} - \left(\iint dx dy p \right)_{s=s_1}. \quad (2.204)$$

Making the hydrostatic approximation, whereby the vertical momentum equation maintains the inviscid hydrostatic balance, allows us to note that the difference in pressure between the top and bottom surfaces

of the region is determined by the weight of fluid between the surfaces,

$$\iint_{s=s_2} dx dy p - \iint_{s=s_1} dx dy p = g \int \rho dV. \quad (2.205)$$

It is notable that this expression relates the difference in contact forces acting on the domain boundaries to the integral of a body force (the gravitational force) acting throughout the domain interior.

We now work on reformulating the horizontal pressure forces into a manner amenable to finite volume discretization. Referring to Figure 2.9, we can write the horizontal forces in a manner than builds in the orientation of pressure via a counterclockwise contour integral

$$\begin{aligned} F_{\text{pressure}}^x &= - \left(\int dy \int_{z_1}^{z_2} dz p \right)_{x=x_1} - \left(\int dy \int_{x_1}^{x_2} dx z_{,x} p \right)_{s=s_2} \\ &\quad - \left(\int dy \int_{z_2}^{z_1} dz p \right)_{x=x_2} - \left(\int dy \int_{x_2}^{x_1} dx z_{,x} p \right)_{s=s_1} \\ &= - \left(\int dy \int_{z_1}^{z_2} dz p \right)_{x=x_1} - \left(\int dy \int_{x_1}^{x_2} dz p \right)_{s=s_2} \\ &\quad - \left(\int dy \int_{z_2}^{z_1} dz p \right)_{x=x_2} - \left(\int dy \int_{x_2}^{x_1} dz p \right)_{s=s_1} \\ &= - \int dy \oint_{ABCD} dz p. \end{aligned} \quad (2.206)$$

In the penultimate step, we set $z_{,x} dx = dz$, which is an relation valid along the particular contour $ABCD$. That is, in all the integrals, the differential increment dz is taken along the contour surrounding the cell. The counter-clockwise orientation of the integral follows from the compressive nature of pressure. Since the contour of integration is closed, we have the identity

$$\begin{aligned} F_{\text{pressure}}^x &= - \int dy \oint_{ABCD} p dz \\ &= \int dy \oint_{ABCD} z dp. \end{aligned} \quad (2.207)$$

The contour integral form of the pressure force is key to providing a finite volume discretization that is consistent with Newton's Third Law (Lin, 1997; Adcroft et al., 2008). What is needed next is an assumption about the subgrid profiles for pressure and geopotential $\Phi = gz$ in order to evaluate the contour integral.

2.8.2 Pressure gradient body force in hydrostatic fluids

In the early finite difference formulations of the pressure force, modelers discretized the gradient of pressure and performed certain grid averages so that the gradient occurs at the appropriate grid point. Guidance to the discretization details was provided by concerns of energetic consistency (Chapter 14), whereby work done by pressure in the discrete algorithm is balanced by buoyancy work (Bryan, 1969). This general philosophy *still* guides the formulation of the pressure force in MOM.

As with the contact forces formulation, in a hydrostatic fluid we are only concerned with horizontal pressure gradients, since the vertical momentum equation is reduced to the inviscid hydrostatic balance.

Hence, we are concerned with the horizontal acceleration arising from pressure differences in a hydrostatic and non-Boussinesq fluid, and this acceleration can be written¹³

$$\begin{aligned}\rho^{-1} \nabla_z p &= \rho^{-1} (\nabla_s - \nabla_s z \partial_z) p \\ &= \rho^{-1} \nabla_s p + g \nabla_s z \\ &= \rho^{-1} \nabla_s p + \nabla_s \Phi,\end{aligned}\tag{2.208}$$

where the hydrostatic relation $p_{,z} = -\rho g$ was used to reach the second equality, and

$$\Phi = g z \tag{2.209}$$

is the geopotential. To reach this result, we used the expression

$$\nabla_z = \nabla_s - \nabla_s z \partial_z, \tag{2.210}$$

which relates the lateral gradient operator acting on constant depth surfaces, ∇_z , to the lateral operator acting on surfaces of constant generalized vertical coordinate, ∇_s .

Depending on the choice for the vertical coordinate s , discretizations of the pressure gradient body force can result in both terms in equation (2.208) being large and of opposite sign in many regions. This issue is especially pernicious for terrain following coordinates in regions of nontrivial topographic slope (e.g., Griffies et al., 2000a). Hence, this calculation exposes the discrete pressure gradient force to nontrivial numerical truncation errors which can lead to spurious numerical pressure gradients and thus to incorrect simulated currents. Significant effort has gone into reducing such *pressure gradient errors*, especially in terrain following models where undulations of the coordinate surfaces can be large with realistic topography (e.g., see Figure 5.3). Some of these issues are summarized in Section 2 of Griffies et al. (2000a).

The pressure gradient force acting at a point represents the infinitesimal limit of a body force. We see this fact by multiplying the pressure gradient acceleration by the mass of a fluid parcel, which leads to the pressure force acting at a point in the continuum

$$\begin{aligned}\text{PRESSURE GRADIENT FORCE} &= -(\rho dV) \rho^{-1} \nabla_z p \\ &= -dV \nabla_z p \\ &= -dV (\nabla_s p + \rho \nabla_s \Phi).\end{aligned}\tag{2.211}$$

Hence, the pressure force acting on a finite region is given by the integral over the extent of the region

$$\begin{aligned}\text{PRESSURE GRADIENT FORCE OVER REGION} &= - \iiint (\rho dV) \rho^{-1} \nabla_z p \\ &= - \iiint dV \nabla_z p.\end{aligned}\tag{2.212}$$

As stated earlier, a finite volume discretization of this force will take the same form as the finite volume discretization of the pressure contact force discussed in Section 2.8.1, as it should due to the Green-Gauss Theorem invoked to go from equation (2.187) to (2.188). However, these formulations generally *do not* provide for a clear energetic interpretation as promoted by the finite difference formulation of Bryan (1969).

2.9 Linear momentum budget

The purpose of this section is to formulate the budget for linear momentum over a finite region of the ocean, with specific application to ocean model grid cells. The material here requires many of the same elements as in Section 2.6, but with added richness arising from the vector nature of momentum, and the additional considerations of forces from pressure, friction, gravity, and planetary rotation. Note that we initially formulate the equations using the pressure contact force, as this provides a general formulation. Afterwards, we specialize to hydrostatic fluids, and thus write the pressure force as a gradient (Section 2.8.2), as commonly done in primitive equation ocean models

¹³For a Boussinesq fluid, equation (2.208) is modified by a factor of ρ/ρ_0 .

2.9.1 General formulation

The budget of linear momentum for a finite region of fluid is given by the following relation based on Newton's second and third laws

$$\begin{aligned} \partial_t \left(\iiint dV \rho \mathbf{v} \right) = & \iiint dV \mathcal{S}^{(\mathbf{v})} - \iint dA_{(\hat{\mathbf{n}})} [\hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{\text{ref}})] \rho \mathbf{v} \\ & + \iint dA_{(\hat{\mathbf{n}})} (\hat{\mathbf{n}} \cdot \boldsymbol{\tau} - \hat{\mathbf{n}} p) - \iiint dV \rho [g \hat{\mathbf{z}} + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge \mathbf{v}]. \end{aligned} \quad (2.213)$$

The left hand side is the time tendency of the region's linear momentum. The first term on the right hand side, $\mathcal{S}^{(\mathbf{v})}$, is a momentum source, with units momentum per volume per time. This term is nonzero if, for example, the addition of mass to the ocean via a source occurs with a nonzero momentum. Often, it is assumed that mass is added with zero velocity, and so does not appear as a momentum source. The second term is the advective transport of linear momentum across the boundary of the region, with recognition that the region's boundaries are generally moving with velocity \mathbf{v}^{ref} . The third term is the integral of the contact stresses due to friction and pressure. These stresses act on the boundary of the fluid domain. We already discussed the forces from pressure in Section 2.8. The stress tensor $\boldsymbol{\tau}$ is a symmetric second order tensor that parameterizes subgrid scale transport of momentum. The final term on the right hand side is the volume integral of body forces due to gravity and the Coriolis force.¹⁴ In addition, there is a body force arising from the nonzero curvature of the spherical space. This curvature leads to the advection metric frequency (see equation (4.49) of Griffies (2004))

$$\mathcal{M} = v \partial_x \ln dy - u \partial_y \ln dx. \quad (2.214)$$

In spherical coordinates where

$$dx = (r \cos \phi) d\lambda \quad (2.215)$$

$$dy = r d\phi, \quad (2.216)$$

with r the distance from the earth's center, λ the longitude, and ϕ the latitude, the advective metric frequency takes the form

$$\mathcal{M} = (u/r) \tan \phi. \quad (2.217)$$

The advection metric frequency arises since linear momentum is not conserved on the sphere.¹⁵ Hence, the linear momentum budget picks up this extra term that is a function of the chosen lateral coordinates.

2.9.2 An interior grid cell

At the west side of a grid cell, $\hat{\mathbf{n}} = -\hat{\mathbf{x}}$ whereas $\hat{\mathbf{n}} = \hat{\mathbf{x}}$ on the east side. Hence, the advective transport of linear momentum entering through the west side of the grid cell and that which is leaving through the east side are given by

$$\text{TRANSPORT ENTERING FROM WEST} = \iint_{x=x_1} dy ds z_s u (\rho \mathbf{v}) \quad (2.218)$$

$$\text{TRANSPORT LEAVING THROUGH EAST} = - \iint_{x=x_2} dy ds z_s u (\rho \mathbf{v}). \quad (2.219)$$

¹⁴The wedge symbol \wedge represents a vector cross product, also commonly written as \times . The wedge is typically used in the physics literature, and is preferred here to avoid confusion with the horizontal coordinate x .

¹⁵Linear momentum is not conserved for ideal flow on a sphere. Instead, angular momentum is conserved for ideal fluid flow on the sphere in the absence of horizontal boundaries (see Section 4.11.2 of Griffies (2004)).

Similar results hold for momentum crossing the cell boundaries in the north and south directions. Momentum crossing the top and bottom surfaces of an interior cell is given by

$$\text{TRANSPORT ENTERING FROM THE BOTTOM} = \iint_{s=s_2} dx dy w^{(z)}(\rho \mathbf{v}) \quad (2.220)$$

$$\text{TRANSPORT LEAVING FROM THE TOP} = - \iint_{s=s_1} dx dy w^{(z)}(\rho \mathbf{v}). \quad (2.221)$$

Forces due to the contact stresses at the west and east sides are given by

$$\text{CONTACT FORCE ON WEST SIDE} = - \iint_{x=x_1} dy ds z_{,s} (\hat{\mathbf{x}} \cdot \boldsymbol{\tau} - \hat{\mathbf{x}} p) \quad (2.222)$$

$$\text{CONTACT FORCE ON EAST SIDE} = \iint_{x=x_2} dy ds z_{,s} (\hat{\mathbf{x}} \cdot \boldsymbol{\tau} - \hat{\mathbf{x}} p) \quad (2.223)$$

with similar results at the north and south sides. At the top of the cell, $dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} = \nabla s dx dy$ whereas $dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} = -\nabla s dx dy$ at the bottom. Hence,

$$\text{CONTACT FORCE ON CELL TOP} = \iint_{s=s_{k-1}} dx dy z_{,s} (\nabla s \cdot \boldsymbol{\tau} - p \nabla s) \quad (2.224)$$

$$\text{CONTACT FORCE ON CELL BOTTOM} = - \iint_{s=s_k} dy ds z_{,s} (\nabla s \cdot \boldsymbol{\tau} - p \nabla s). \quad (2.225)$$

Bringing these results together, and taking limit as the time independent horizontal area $dx dy \rightarrow 0$, leads to the thickness weighted budget for the momentum per horizontal area of an interior grid cell

$$\begin{aligned} \partial_t (dz \rho \mathbf{v}) &= dz S^{(\mathbf{v})} - \nabla_s \cdot [dz \mathbf{u}(\rho \mathbf{v})] + (w^{(z)} \rho \mathbf{v})_{s=s_k} - (w^{(z)} \rho \mathbf{v})_{s=s_{k-1}} \\ &\quad + \partial_x [dz (\hat{\mathbf{x}} \cdot \boldsymbol{\tau} - \hat{\mathbf{x}} p)] + \partial_y [dz (\hat{\mathbf{y}} \cdot \boldsymbol{\tau} - \hat{\mathbf{y}} p)] \\ &\quad + [z_{,s} (\nabla s \cdot \boldsymbol{\tau} - p \nabla s)]_{s=s_{k-1}} - [z_{,s} (\nabla s \cdot \boldsymbol{\tau} - p \nabla s)]_{s=s_k} \\ &\quad - \rho dz [g \hat{\mathbf{z}} + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge \mathbf{v}]. \end{aligned} \quad (2.226)$$

Note that both the time and horizontal partial derivatives are for positions fixed on a constant generalized vertical coordinate surface. Also, the pressure force as written here is a shorthand for the more complete contour integral formulation provided in Section 2.8 (e.g., equation (2.207)). Additionally, we have yet to take the hydrostatic approximation, so these equations are written for the three components of the velocity.

The first term on the right hand side of the thickness weighted momentum budget (2.226) is the momentum source, and the second is the convergence of advective momentum fluxes occurring within the layer. We discussed the analogous flux convergence for the tracer and mass budgets in Section 2.7.4. The third and fourth terms arise from the transport of momentum across the upper and lower constant s interfaces. The fifth and sixth terms arise from the horizontal convergence of pressure and viscous stresses. The seventh and eighth terms arise from the frictional and pressure stresses acting on the constant generalized surfaces. These forces provide an interfacial stress between layers of constant s . Note that even in the absence of frictional stresses, interfacial stresses from pressure acting on the generally curved s surface can transmit momentum between vertically stacked layers. The final term arises from the gravitational force, the Coriolis force, and the advective frequency.

2.9.3 Cell adjacent to the ocean bottom

As for the tracer and mass budgets, we assume zero mass flux through the ocean bottom at $z = -H(x, y)$. However, there is generally a nonzero stress at the bottom due to both the pressure between the fluid

and the bottom, and unresolved features in the flow which can correlate or anti-correlate with bottom topographic features (Holloway (1999)). The area integral of the stresses lead to a force on the fluid at the bottom

$$\mathbf{F}_{\text{bottom}} = - \iint_{z=-H} dx dy [\nabla(z+H) \cdot \boldsymbol{\tau} - p \nabla(z+H)]. \quad (2.227)$$

Details of the stress term requires fine scale information that is generally unavailable. For present purposes we assume that some boundary layer model provides information that is schematically written

$$\boldsymbol{\tau}^{\text{bot}} = \nabla(z+H) \cdot \boldsymbol{\tau} \quad (2.228)$$

where $\boldsymbol{\tau}^{\text{bot}}$ is a vector bottom stress. Taking the limit as the horizontal area vanishes leads to the thickness weighted budget for momentum per horizontal area of a grid cell next to the ocean bottom

$$\begin{aligned} \partial_t (dz \rho \mathbf{v}) = & dz \mathcal{S}^{(\mathbf{v})} - \nabla_s \cdot [dz \mathbf{u}(\rho \mathbf{v})] - (w^{(z)} \rho \mathbf{v})_{s=s_{k\text{bot}-1}} \\ & + \partial_x [dz (\hat{\mathbf{x}} \cdot \boldsymbol{\tau} - \hat{\mathbf{x}} p)] + \partial_y [dz (\hat{\mathbf{y}} \cdot \boldsymbol{\tau} - \hat{\mathbf{y}} p)] \\ & + [z_{,s} (\nabla s \cdot \boldsymbol{\tau} - p \nabla s)]_{s=s_{k\text{bot}-1}} \\ & - \boldsymbol{\tau}^{\text{bot}} + p_b \nabla(z+H) \\ & - \rho dz [g \hat{\mathbf{z}} + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge \mathbf{v}]. \end{aligned} \quad (2.229)$$

2.9.4 Cell adjacent to the ocean surface

There is a nonzero mass and momentum flux through the upper ocean surface at $z = \eta(x, y, t)$, and contact stresses are applied from resolved and unresolved processes involving interactions with the atmosphere and sea ice. Following the discussion of the tracer budget at the ocean surface in Section 2.6.4 leads to the expression for the transport of momentum into the ocean due to mass transport at the surface

$$- \iint dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v} - \mathbf{v}^{\text{ref}}) \rho \mathbf{v}] = \iint_{z=\eta} dx dy Q_m \mathbf{v}. \quad (2.230)$$

The force arising from the contact stresses at the surface is written

$$\mathbf{F}_{\text{contact}} = \iint_{z=\eta} dx dy [\nabla(z-\eta) \cdot \boldsymbol{\tau} - p \nabla(z-\eta)]. \quad (2.231)$$

Bringing these results together leads to the force acting at the ocean surface

$$\mathbf{F}_{\text{surface}} = \iint_{z=\eta} dx dy [\nabla(z-\eta) \cdot \boldsymbol{\tau} - p \nabla(z-\eta) + Q_m \mathbf{v}]. \quad (2.232)$$

Details of the various terms in this force are generally unknown. Therefore, just as for the tracer at $z = \eta$ in Section 2.6.4, we assume that a boundary layer model provides information about the total force, and that this force is written

$$\mathbf{F}_{\text{surface}} = \iint_{z=\eta} dx dy [\boldsymbol{\tau}^{\text{top}} - p_a \nabla(z-\eta) + Q_m \mathbf{v}_m], \quad (2.233)$$

where \mathbf{v}_m is the velocity of the water crossing the ocean surface. This velocity is typically taken to be equal to the velocity of the ocean currents in the top cells of the ocean model, but such is not necessarily the case when considering the different velocities of, say, river water and precipitation. The stress $\boldsymbol{\tau}^{\text{top}}$ is that arising from the wind, as well as interactions between the ocean and sea ice. Letting the horizontal area vanish

leads to the thickness weighted budget for a grid cell next to the ocean surface

$$\begin{aligned}
\partial_t (dz \rho \mathbf{v}) &= dz S^{(\mathbf{v})} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{v})] + (w^{(z)} \rho \mathbf{v})_{s=s_k=1} \\
&\quad + \partial_x [dz (\hat{\mathbf{x}} \cdot \boldsymbol{\tau} - \hat{\mathbf{x}} p)] + \partial_y [dz (\hat{\mathbf{y}} \cdot \boldsymbol{\tau} - \hat{\mathbf{y}} p)] \\
&\quad - [z_{,s} (\nabla s \cdot \boldsymbol{\tau} - p \nabla s)]_{s=s_k=1} \\
&\quad + [\boldsymbol{\tau}^{\text{top}} - p_a \nabla (z - \eta) + Q_m \mathbf{v}_m] \\
&\quad - \rho dz [g \hat{\mathbf{z}} + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge \mathbf{v}].
\end{aligned} \tag{2.234}$$

2.9.5 Horizontal momentum equations for hydrostatic fluids

We now assume the fluid to maintain a hydrostatic balance, which is the case for primitive equation ocean general circulation models. In this case, we exploit the pressure gradient body force as discussed in Section 2.8.2. Specializing the momentum budgets from Sections 2.9.2, 2.9.3, and 2.9.4 to use the hydrostatic pressure gradient force (again, interpreted according to the finite volume form given in Section 2.8) leads to the horizontal linear momentum budget for interior, bottom, and surface grid cells

$$\begin{aligned}
[\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge] (\rho dz \mathbf{u}) &= dz S^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] \\
&\quad - dz (\nabla_s p + \rho \nabla_s \Phi) \\
&\quad + \partial_x (dz \hat{\mathbf{x}} \cdot \boldsymbol{\tau}) + \partial_y (dz \hat{\mathbf{y}} \cdot \boldsymbol{\tau}) \\
&\quad - [w^{(z)} \rho \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_k-1} \\
&\quad + [w^{(z)} \rho \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_k}.
\end{aligned} \tag{2.235}$$

$$\begin{aligned}
[\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge] (\rho dz \mathbf{u}) &= dz S^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] \\
&\quad - dz (\nabla_s p + \rho \nabla_s \Phi) \\
&\quad + \partial_x (dz \hat{\mathbf{x}} \cdot \boldsymbol{\tau}) + \partial_y (dz \hat{\mathbf{y}} \cdot \boldsymbol{\tau}) \\
&\quad - [w^{(z)} \rho \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_{kbot}-1} \\
&\quad - \boldsymbol{\tau}^{\text{bottom}}
\end{aligned} \tag{2.236}$$

$$\begin{aligned}
[\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge] (\rho dz \mathbf{u}) &= dz S^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] \\
&\quad - dz (\nabla_s p + \rho \nabla_s \Phi) \\
&\quad + \partial_x (dz \hat{\mathbf{x}} \cdot \boldsymbol{\tau}) + \partial_y (dz \hat{\mathbf{y}} \cdot \boldsymbol{\tau}) \\
&\quad + [\boldsymbol{\tau}^{\text{wind}} + Q_m \mathbf{u}_w] \\
&\quad + [w^{(z)} \rho \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_1}.
\end{aligned} \tag{2.237}$$

2.10 The Boussinesq budgets

We consider various depth-based vertical coordinates in Section 5.1. These coordinates are used to discretize the Boussinesq model equations where the volume of a parcel is conserved rather than the mass. A detailed discussion of the interpretation of the Boussinesq equations in terms of density weighted fields is given by McDougall et al. (2002) and Griffies (2004). For now, we gloss over those details by quoting the Boussinesq equations for volume, tracer, and momentum as arising from setting all density factors to the constant ρ_o , except when multiplied by the gravitational acceleration in the hydrostatic balance (i.e., for calculation of pressure and geopotential, the full density is used). The density ρ_o is a representative density of the ocean fluid. In MOM4 we set

$$\rho_o = 1035 \text{ kg/m}^3, \tag{2.238}$$

although this value can be changed via altering a parameter statement and thus recompiling the code). For much of the ocean, the *in situ* density deviates less than 3% from 1035 kg m^{-3} (see page 47 of Gill (1982)).

The replacement of density in the mass, tracer, and linear momentum budgets over a grid cell in the ocean interior leads to the following budgets for the hydrostatic model

$$\begin{aligned}
\partial_t(dz) &= dz \mathcal{S}^{(V)} - \nabla_s \cdot (dz \mathbf{u}) - (w^{(z)})_{s=s_{k-1}} + (w^{(z)})_{s=s_k} \\
\partial_t(dz C) &= dz \mathcal{S}^{(C)} - \nabla_s \cdot [dz(\mathbf{u} C + \mathbf{F})] \\
&\quad - (w^{(z)} C + F^{(z)})_{s=s_{k-1}} \\
&\quad + (w^{(z)} C + F^{(z)})_{s=s_k} \\
[\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge](\rho_o dz \mathbf{u}) &= dz \mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u}(\rho_o \mathbf{u})] \\
&\quad - dz(\nabla_s p + \rho \nabla_s \Phi) \\
&\quad + \partial_x(dz \hat{\mathbf{x}} \cdot \boldsymbol{\tau}) + \partial_y(dz \hat{\mathbf{y}} \cdot \boldsymbol{\tau}) \\
&\quad - [w^{(z)} \rho_o \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_{k-1}} \\
&\quad + [w^{(z)} \rho_o \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_k}.
\end{aligned} \tag{2.239}$$

The first equation reduces to a volume budget rather than a mass budget found for the non-Boussinesq system. In this equation, $\mathcal{S}^{(V)}$ is a volume source with units of inverse time. Likewise, $\mathcal{S}^{(\mathbf{u})}$ is a velocity source (with units of acceleration). The Boussinesq equations for a grid cell adjacent to the ocean bottom are given by

$$\begin{aligned}
\partial_t(dz) &= dz \mathcal{S}^{(V)} - \nabla_s \cdot (dz \mathbf{u}) - (w^{(z)})_{s=s_{kbot-1}} \\
\partial_t(dz C) &= dz \mathcal{S}^{(C)} - \nabla_s \cdot [dz(\mathbf{u} C + \mathbf{F})] \\
&\quad - (w^{(z)} C + F^{(z)})_{s=s_{kbot-1}} \\
&\quad + Q_{(bot)}^{(C)} \\
[\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge](\rho_o dz \mathbf{u}) &= dz \mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u}(\rho_o \mathbf{u})] \\
&\quad - dz(\nabla_s p + \rho \nabla_s \Phi) \\
&\quad + \partial_x(dz \hat{\mathbf{x}} \cdot \boldsymbol{\tau}) + \partial_y(dz \hat{\mathbf{y}} \cdot \boldsymbol{\tau}) \\
&\quad - [w^{(z)} \rho_o \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_{kbot-1}} \\
&\quad - \boldsymbol{\tau}^{bottom}
\end{aligned} \tag{2.240}$$

and the equations for a cell next to the ocean surface are

$$\begin{aligned}
\partial_t(dz) &= dz \mathcal{S}^{(V)} - \nabla_s \cdot (dz \mathbf{u}) + (w^{(z)})_{s=s_{k=1}} + Q_m/\rho_o \\
\partial_t(dz C) &= dz \mathcal{S}^{(C)} - \nabla_s \cdot [dz(\mathbf{u} C + \mathbf{F})] \\
&\quad + (w^{(z)} C + F^{(z)})_{s=s_{k=1}} \\
&\quad + ((Q_m/\rho_o) C_m - Q_{(C)}^{(turb)}) \\
[\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge](\rho_o dz \mathbf{u}) &= dz \mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u}(\rho_o \mathbf{u})] \\
&\quad - dz(\nabla_s p + \rho \nabla_s \Phi) \\
&\quad + \partial_x(dz \hat{\mathbf{x}} \cdot \boldsymbol{\tau}) + \partial_y(dz \hat{\mathbf{y}} \cdot \boldsymbol{\tau}) \\
&\quad + [\boldsymbol{\tau}^{wind} + Q_m \mathbf{u}_w] \\
&\quad + [w^{(z)} \rho_o \mathbf{u} - z_{,s} \nabla s \cdot \boldsymbol{\tau}]_{s=s_1}.
\end{aligned} \tag{2.241}$$

THE HYDROSTATIC PRESSURE FORCE

Contents

3.1	Hydrostatic pressure forces at a point	65
3.2	Pressure gradient body force	66
3.2.1	Depth based vertical coordinates	67
3.2.1.1	Geopotential vertical coordinates	68
3.2.1.2	z^* and $\sigma^{(z)}$ vertical coordinate	69
3.2.2	A test case for zero cross-coordinate flow	69
3.2.3	Pressure based vertical coordinates	70
3.3	Pressure gradient body force in B-grid MOM	71
3.3.1	Depth based vertical coordinates	72
3.3.2	Pressure based vertical coordinates	73
3.4	Pressure gradient body force in C-grid MOM	74
3.4.1	Depth based vertical coordinates	74
3.4.2	Pressure based vertical coordinates	74

The purpose of this chapter is to detail issues related to computing the pressure force in hydrostatic ocean models. Care is taken to split the pressure force into its slow and fast components, thus facilitating a split of the momentum equation for use in an explicit time stepping scheme. Additional consideration is given to the distinct needs of the B-grid and C-grid implementations available in MOM.

In Section 2.8, we encountered two formulations of the pressure force. The first computes the pressure gradient body force (Section 2.8.2), and considers the pressure force to be acting at a point. This interpretation follows from a finite difference interpretation of the velocity equation, following the energetic approach of Bryan (1969) and all versions of MOM. The second formulation applies a finite volume interpretation advocated in Chapter 2, with particular attention given to the contour integral form of pressure as derived in Section 2.8.1. The finite volume approach does *not* lend itself to straightforward energetic conversion arguments (Chapter 14). It is for this reason that we maintain the finite difference approach of Bryan (1969) in MOM.

3.1 Hydrostatic pressure forces at a point

A hydrostatic fluid maintains the balance

$$\frac{\partial p}{\partial z} = -\rho g. \quad (3.1)$$

This balance means that the pressure at a point in a hydrostatic fluid is determined by the weight of fluid above this point. This relation is maintained quite well in the ocean on horizontal spatial scales larger than

roughly 1km. Precisely, when the squared ratio of the vertical to horizontal scales of motion is small, then the hydrostatic approximation is well maintained. In this case, the vertical momentum budget reduces to the hydrostatic balance, in which case vertical acceleration and friction are neglected. If we are interested in explicitly representing such motions as Kelvin-Helmholtz billows and flow within a convective chimney, vertical accelerations are nontrivial and so the non-hydrostatic momentum budget must be used.

The hydrostatic balance greatly affects the algorithms used to numerically solve the equations of motion. [Marshall et al. \(1997\)](#) highlight these points in the context of developing an algorithm suited for both hydrostatic and non-hydrostatic simulations. However, so far no long-term global climate simulations have been run at resolutions sufficiently refined to require the non-hydrostatic equations. Additionally, many regional and coastal models, even some with grid resolutions finer than 1km, still maintain the hydrostatic approximation, and thus they must parameterize the unrepresented non-hydrostatic motions.

As discussed in Section 2.8.2, at a point in the continuum, the horizontal acceleration arising from pressure differences in a hydrostatic and non-Boussinesq fluid can be written¹

$$\begin{aligned}\rho^{-1} \nabla_z p &= \rho^{-1} (\nabla_s - \nabla_s z \partial_z) p \\ &= \rho^{-1} \nabla_s p + g \nabla_s z \\ &= \rho^{-1} (\nabla_s p + \rho \nabla_s \Phi)\end{aligned}\tag{3.2}$$

where the hydrostatic relation $\partial_z p = -\rho g$ was used to reach the second equality, and

$$\Phi = g z \tag{3.3}$$

is the geopotential. The general expression for the horizontal pressure gradient to evaluate in an ocean model is thus given by

$$\nabla_z p = \nabla_s p + \rho \nabla_s \Phi \tag{3.4}$$

For cases where the density is constant on s surfaces, we can combine the two terms on the right hand side into the gradient of a scalar, thus rendering a horizontal pressure gradient force with a zero curl. This special case holds for geopotential and pressure coordinates. It also holds for isopycnal coordinates in the special case of an idealized linear equation of state. However, it does not hold in the more general case, in which the difficulty of numerically computing the acceleration from pressure arises when there are contributions from *both* terms. Generally, both terms can be large and of opposite sign in many regions. In this case, the simulation is exposed to nontrivial numerical truncation errors that can, for example, lead to spurious pressure gradients that spin up an unforced fluid with initially flat isopycnals. However, in certain cases one term dominates, in which case an accurate pressure gradient is simpler to compute numerically.

Significant effort has gone into reducing such *pressure gradient errors*, especially in terrain following models where undulations of the coordinate surfaces can be large with realistic bottom topography (e.g., see Figure 5.3). Some of these issues are summarized in Section 2 of [Griffies et al. \(2000a\)](#). Perhaps the most promising approach is that proposed by [Shchepetkin and McWilliams \(2002\)](#). It is notable that difficulties with pressure gradient errors have largely been responsible for the near absence of sigma models being used for long term global ocean climate simulations.²

3.2 Pressure gradient body force

As stated above, the presence of both terms on the right hand side of equation (3.4) complicates the numerical implementation of the horizontal pressure gradient force. The problem is that numerical errors in one term are often not compensated by the other term, and such can lead to spurious flows. For the quasi-horizontal depth based and pressure based coordinates supported by MOM (i.e., $s = z$, $s = z^*$, $s = p$, or $s = p^*$; see Chapter 5), these errors are quite small. The reason is that these choices ensure that one of the two terms appearing in equation (3.4) is significantly smaller than the other. Nonetheless, it is useful to provide a formulation that even further reduces the potential for errors for both the quasi-horizontal coordinates, as well as the terrain following coordinates $\sigma^{(z)}$ and $\sigma^{(p)}$ (Chapter 5).

¹To obtain this result for a Boussinesq fluid, multiply both sides of equation (3.2) by ρ/ρ_0 .

²The work of [Diansky et al. \(2002\)](#) is one example of a published global sigma model used for climate purposes.

In addition to reducing errors associated with a numerical computation of the pressure gradient, we aim to split the pressure gradient into two terms associated with the slowly evolving internal modes and the faster external mode. Details of this split are a function of the vertical coordinate. This split in the pressure gradient then facilitates our treatment of the vertically integrated momentum equations, as discussed in Section 10.9.

In the following, we are motivated by the formulation of the pressure gradient commonly applied to z -models. [Adcroft and Campin \(2004\)](#) extended this treatment to the z^* vertical coordinate. We take it one more step in order to handle all vertical coordinates supported by MOM. [Hallberg \(1997\)](#) goes further by treating the pressure gradient in isopycnal layered models using a realistic equation of state, and [Adcroft et al. \(2008\)](#) present a more accurate approach for generalized vertical coordinate models.

3.2.1 Depth based vertical coordinates

As mentioned on page 47 of [Gill \(1982\)](#), *in situ* density in the bulk of the ocean deviates less than 3% from the constant density

$$\rho_o = 1035 \text{ kg/m}^3. \quad (3.5)$$

The hydrostatic pressure associated with this constant density has no horizontal gradients, and so it does not contribute to horizontal pressure gradient forces. For increased accuracy computing the horizontal pressure gradient, it is useful to remove this term from the calculation of hydrostatic pressure. For this purpose, we write the hydrostatic balance as

$$\begin{aligned} \frac{\partial p}{\partial z} &= -g\rho \\ &= -g(\rho_o + \rho'), \end{aligned} \quad (3.6)$$

which has an associated split in the hydrostatic pressure field

$$p = p_a + p_o(z) + p'(x, y, z, t). \quad (3.7)$$

We can solve for the pressures by assuming

$$p_o(z = \eta) = 0 \quad (3.8)$$

$$p'(z = \eta) = 0, \quad (3.9)$$

which leads to

$$\begin{aligned} p_o &= -g\rho_o(z - \eta) \\ &= -\rho_o\Phi + g\rho_o\eta, \end{aligned} \quad (3.10)$$

and

$$p' = g \int_z^\eta \rho' dz, \quad (3.11)$$

and thus

$$p = p_a + g\rho_o\eta - \rho_o\Phi + p'. \quad (3.12)$$

Splitting off the free surface height is advantageous as it allows for a split of the pressure gradient into its fast two dimensional barotropic contributions and slow three dimensional baroclinic contributions. This split in pressure gradient facilitates the development of a split-explicit time stepping method for the momentum equations considered in Section 10.9. Details of the split in pressure are dependent on the vertical coordinate choice. We now discuss the three depth based vertical coordinates used in MOM.

3.2.1.1 Geopotential vertical coordinates

We first consider the horizontal pressure gradient realized with geopotential vertical coordinates. We are here motivated by the desire to split the dynamics into fast and slow portions, as approximated by depth integrating the momentum equation (Section 10.9).

The anomalous pressure p' maintains a dependence on surface height through the upper limit on the vertical integral in equation (3.11). When working with geopotential vertical coordinates, it is very convenient to isolate this dependence by exploiting a very accurate approximation described below. This split then allows us to exclusively place the surface height dependent pressure gradient into the vertically integrated momentum equation. The slow component to the pressure gradient then has no dependence on the surface height; it is instead just a function of the anomalous density. The slow pressure gradient component thus vanishes when the density is horizontally unstratified; i.e., when there is no baroclinicity.

To facilitate the split described above, we proceed in the following manner

$$\begin{aligned}
 p' &= g \int_z^\eta \rho' dz \\
 &= g \int_z^0 \rho' dz + g \int_0^\eta \rho' dz \\
 &\approx g \int_z^0 \rho' dz + g \eta \rho'_{\text{surf}} \\
 &\equiv p'_{\text{clinic}} + p'_{\text{surf}}.
 \end{aligned} \tag{3.13}$$

The approximation made in the third step remains good where density is well mixed between $z = 0$ and $z = \eta$, and this is generally the case for large scale modelling. Here, density in the surface region of the ocean is assumed to take on the value

$$\rho_{\text{surf}} = \rho_o + \rho'_{\text{surf}}, \tag{3.14}$$

which is a function of horizontal position and time. The anomalous pressure p' has therefore been separated into two pressures, where the anomalous *surface pressure*

$$p'_{\text{surf}} = \rho'_{\text{surf}} g \eta \tag{3.15}$$

is a function of the surface height and surface density, and the pressure

$$p'_{\text{clinic}} = g \int_z^0 \rho' dz \tag{3.16}$$

is the anomalous hydrostatic *baroclinic pressure* within the region from a depth $z < 0$ to $z = 0$. Again, the baroclinic pressure is independent of the surface height, and so its horizontal gradients are only a function of density.

This split of pressure thus renders the horizontal pressure gradient (equation (3.4))

$$\begin{aligned}
 (\nabla_z p)_{\text{approx}} &= \nabla_s p + \rho \nabla_s \Phi \\
 &= \nabla_s (p_a + g \rho_o \eta - \rho_o \Phi + p') + \rho \nabla_s \Phi \\
 &\approx \nabla (p_a + g \rho_o \eta + p'_{\text{surf}}) + \nabla_s p'_{\text{clinic}} + (\rho - \rho_o) \nabla_s \Phi \\
 &= \underbrace{\nabla (p_a + g \rho_{\text{surf}} \eta)}_{\text{fast}} + \underbrace{\nabla_s p'_{\text{clinic}} + \rho' \nabla_s \Phi}_{\text{slow}}.
 \end{aligned} \tag{3.17}$$

In a geopotential vertical coordinate model, interior grid cells are discretized at levels of constant geopotential. Hence, the gradient ∇_s reduces to the constant geopotential gradient ∇_z . In this case the horizontal gradient of the geopotential vanishes, $\nabla_z \Phi = 0$. At the bottom, however, MOM employs bottom partial step topography (Pacanowski and Gnanadesikan, 1998). The bottom cells are thus *not* discretized along a constant geopotential. Hence, just at the bottom, there is a nontrivial gradient of the geopotential Φ (see Figure 5.1). In general, note how the geopotential is multiplied by the anomalous density $\rho' = \rho - \rho_o$, thus minimizing the impact of this term.

3.2.1.2 z^* and $\sigma^{(z)}$ vertical coordinate

The new issue that arises when moving away from geopotential coordinates is that the geopotential $\Phi = gz$ has a nonzero along coordinate gradient in the interior, whereas with geopotential coordinates it remains nonzero only along the partial bottom stepped topography. The presence of Φ gradients in the interior is fundamental.

Following the discussion in Section 3.2.1, we are led to the following expressions for the horizontal pressure gradient. The exact expression relevant for the z^* and $\sigma^{(z)}$ coordinates is given by

$$\begin{aligned}
 (\nabla_z p)_{\text{exact}} &= \nabla_s p + \rho \nabla_s \Phi \\
 &= \nabla_s (p_a + p_o + p') + \rho \nabla_s \Phi \\
 &= \nabla_s (p_a - \rho_o \Phi + g \rho_o \eta + p') + \rho \nabla_s \Phi \\
 &= \underbrace{\nabla (p_a + g \rho_o \eta)}_{\text{fast}} + \underbrace{\nabla_s p' + \rho' \nabla_s \Phi}_{\text{slow}}.
 \end{aligned} \tag{3.18}$$

Note that we have *assumed* that the geopotential falls inside the slow portion of the pressure gradient. This assumption is made *even though* the depth of a grid point is a function of the undulating surface height. The validity of this assumption can be assessed by the integrity and stability of the simulation.

To facilitate a unified treatment in subsequent manipulations, we define

$$\begin{aligned}
 p_{\text{surf}} &= \rho_{\text{surf}} g \eta & s = z \\
 p_{\text{surf}} &= \rho_0 g \eta & s = z^*, \sigma^{(z)}
 \end{aligned} \tag{3.19}$$

and

$$\begin{aligned}
 p' &= g \int_z^0 \rho' dz & s = z \\
 p' &= g \int_z^{\eta} \rho' dz & s = z^*, \sigma^{(z)}.
 \end{aligned} \tag{3.20}$$

In both the exact and approximated pressure gradient expressions, the geopotential gradient $\nabla_s \Phi$ in the ocean interior is weighted by the small density deviation $\rho' = \rho - \rho_o$. For quasi-horizontal depth-based vertical coordinates supported in MOM (Section 5.1), the horizontal gradient of the geopotential is small, and the ρ' weighting further reduces its contribution. For terrain following coordinates, the horizontal gradient term is not small, and the ρ' weighting is essential to reduce its magnitude.

3.2.2 A test case for zero cross-coordinate flow

In the development of generalized vertical coordinates, a useful test case was suggested by Alistair Adcroft. We focus here on the special case of $s = z^*$. In this test, initialize the density field as a function only of the vertical coordinate z^* . The domain is flat bottomed and doubly periodic in the horizontal, thus precluding pressure gradients due to side boundaries or topography. In a state of rest, there is no horizontal pressure gradients, and so no motion. As a body force is applied to the barotropic equations, such as through an ideal tidal forcing, there will now be a nontrivial surface height field η as well as a nontrivial barotropic velocity. Both pieces of the slow contribution to the horizontal pressure gradient (3.18) develop a nontrivial vertical structure, and this will initiate baroclinic structure and thus a nonzero cross coordinate vertical velocity $w^{(s)}$. This cross coordinate velocity will be much smaller in the $s = z^*$ case than with $s = z$, given that z^* follows the motion of the free surface.

In order to further test the integrity of the z^* implementation, we wish to truncate the pressure calculation in this test so that there will be no slow pressure gradients developed when the tidal forcing is applied, and hence there will be no cross coordinate motion. For this purpose, truncate the slow piece of the horizontal pressure gradient (3.18) as

$$\nabla_s p' + \rho' \nabla_s \Phi \rightarrow \nabla_s p'_{\text{truncate}}. \tag{3.21}$$

In this truncation, we drop the geopotential term $\rho' \nabla_s \Phi$, as this term will produce nontrivial horizontal gradients as the surface height undulates. We also introduce a truncated perturbation pressure determined

by

$$\begin{aligned}
p' &= g \int_z^\eta \rho' dz \\
&= g \int_{s(z)}^{s(\eta)} \rho' z_{,s} ds \\
&= g(1 + \eta/H) \int_{z^*}^0 \rho' dz^* \\
&= p'_{\text{truncate}} + (g\eta/H) \int_{z^*}^0 \rho' dz^*.
\end{aligned} \tag{3.22}$$

To reach the penultimate step, we used $z_{,s} = (1 + \eta/H)$ for $s = z^*$. The coordinate increments used to define the pressure field p'_{truncate} are static in a model discretizing the vertical according to $s = z^*$. Hence, $\nabla_s p'_{\text{truncate}} = 0$ if the density is a function only of z^* . So when the model's slow pressure field is comprised of just p'_{truncate} , the ideal tidal test in the torus should maintain zero cross coordinate flow, $w^{z^*} = 0$, even as the surface height fluctuates. Testing to see that this property is maintained is a useful means for evaluating the integrity of the algorithm.

3.2.3 Pressure based vertical coordinates

A complementary discussion to the above is now given for pressure based vertical coordinates. Since for pressure based vertical coordinates we solve for the bottom pressure, it is useful to formulate the geopotential in terms of the bottom pressure rather than the atmospheric pressure. For this purpose, consider the following identities

$$\begin{aligned}
\Phi + gH &= g \int_{-H}^z dz \\
&= g \int_{p_b}^p \frac{\partial z}{\partial p} dp \\
&= - \int_{p_b}^p \rho^{-1} dp \\
&= - \int_{p_b}^p (\rho_o^{-1} + \rho^{-1} - \rho_o^{-1}) dp \\
&= (p_b - p)/\rho_o + \rho_o^{-1} \int_{p_b}^p (\rho'/\rho) dp \\
&= (p_b - p)/\rho_o - (g/\rho_o) \int_{-H}^z \rho' dz.
\end{aligned} \tag{3.23}$$

We are thus led to the expression

$$\rho_o \Phi = p_b - p + \rho_o (\Phi_b + \Phi'), \tag{3.24}$$

where

$$\rho_o \Phi' = -g \int_{-H}^z \rho' dz \tag{3.25}$$

is an anomalous geopotential similar to the anomalous hydrostatic pressure introduced in Section 3.2.1, and

$$\Phi_b = -gH \quad (3.26)$$

is the geopotential at the ocean bottom. The horizontal pressure force is therefore written

$$\begin{aligned} \nabla_s p + \rho \nabla_s \Phi &= \nabla_s p + (\rho/\rho_o) \nabla(p_b + \rho_o \Phi_b) - (\rho/\rho_o) \nabla_s p + \rho \nabla_s \Phi' \\ &= \underbrace{(\rho/\rho_o) \nabla(p_b + \rho_o \Phi_b)}_{\text{fast}} + \underbrace{\rho \nabla_s \Phi' - (\rho'/\rho_o) \nabla_s p}_{\text{slow}}. \end{aligned} \quad (3.27)$$

Note that the three-dimensional pressure term $(\rho'/\rho_o) \nabla_s p$ is weighted by the generally very small density deviation $\rho' = \rho - \rho_o$. For the non-terrain following quasi-horizontal pressure-based vertical coordinates supported in MOM (Section 5.2), the horizontal gradient of the pressure is small, and the weighting by (ρ'/ρ_o) further reduces its contribution. Also note that the *fast* contribution is here weighted by the density, and so this term may appear to require further splitting into $\rho = \rho_o + \rho'$ before identifying the fast two dimensional contribution. However, as the non-Boussinesq formulation here considers momentum per area, the baroclinic velocity includes density weighting (see equation (11.1)). This is how we are to split the horizontal momentum equations into fast two dimensional motions and slow three dimensional motions for purposes of time stepping. We consider these issues further in Sections 10.9 and 11.1.

During the testing of this formulation for the pressure gradient, we found it useful to write the anomalous geopotential in the following form

$$\begin{aligned} -(\rho_o/g) \Phi' &= \int_{-H}^z \rho' dz \\ &= \int_{-H}^{\eta} \rho' dz - \int_z^{\eta} \rho' dz \\ &= \frac{p_b - p_a}{g} - \rho_o(H + \eta) - \int_z^{\eta} \rho' dz \\ &= \frac{p_b - p_a - p'}{g} - \rho_o(H + \eta). \end{aligned} \quad (3.28)$$

To reach this result, we used the hydrostatic balance for the full ocean column in the form

$$\int_{-H}^{\eta} \rho' dz = \frac{p_b - p_a}{g} - \rho_o(H + \eta), \quad (3.29)$$

as well as the definition (3.11) of the anomalous hydrostatic pressure

$$p' = g \int_z^{\eta} \rho' dz \quad (3.30)$$

used in Section 3.2.1 for the depth based vertical coordinates. These results then lead to the identity

$$p_b + \rho_o(\Phi_b + \Phi') = p' + p_a + \rho_o g \eta. \quad (3.31)$$

3.3 Pressure gradient body force in B-grid MOM

We now detail how the pressure gradient body force is represented in the B-grid generalized level coordinate version of MOM. As the pressure force acts to accelerate a fluid parcel, our aim is to determine the

pressure force acting at the velocity cell point. Much in the derivation of the pressure force depends on assumptions regarding where pressure is computed in the discrete model. For the B-grid and C-grid, hydrostatic pressure is coincident with the tracer fields as shown in Figure 2.9, which illustrates a typical case where a grid cell is bounded by vertical sidewalls with generally nonhorizontal tops and bottoms.

As mentioned in Section 2.8.2, we prefer to discretize the pressure gradient body force as it facilitates the splitting of the pressure force into fast and slow components. The result here is a derivation of the Pacanowski and Gnanadesikan (1998) discrete pressure gradient body force as originally implemented for the treatment of partial step bottom topography. Their discussion is relevant here, since the pressure gradient force in the presence of partial step bottoms must account for the pressure between cells that live at different depths. This is also the essential issue for the treatment of pressure with the generalized level coordinates of MOM.

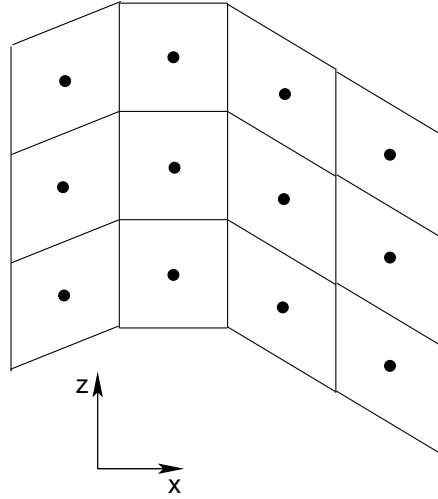


Figure 3.1: Illustration of a vertical slice through a set of grid cells in the x - z plane for a generalized level coordinate version of MOM. The center point in each cell is a tracer point. As the temperature and salinity tracers, along with pressure, determine density, and as density determines hydrostatic pressure, the hydrostatic pressure is coincident with tracer points.

3.3.1 Depth based vertical coordinates

The aim here is to discretize the pressure gradient body force written in the forms (3.18) and (3.17)

$$\nabla_s p + \rho \nabla_s \Phi = \nabla(p_a + p_{\text{surf}}) + \nabla_s p' + \rho' \nabla_s \Phi, \quad (3.32)$$

where p_{surf} and p' are defined according to equations (3.19) and (3.20), respectively. Our focus here is the slowly evolving three dimensional terms $\nabla_s p' + \rho' \nabla_s \Phi$. The first term is straightforward to discretize according to the assumptions regarding the placement of pressure given in Figure 3.1. Pressure sits at tracer points, which are at the corners of velocity cells. Hence, to approximate pressure at the west and east faces of the cell, one can average the pressure found at the corners. A grid weighted average may be appropriate, but the simplest method, which is energetically consistent (see Sections 14.6 and 14.8) is an unweighted average in which

$$\nabla_s p' \approx \hat{\mathbf{x}} \text{FDX.NT}(\text{FAY}(p')) + \hat{\mathbf{y}} \text{FDY.ET}(\text{FAX}(p')) \quad (3.33)$$

The forward algebraic averaging operators are defined according to

$$\text{FAX}(A) = \overline{A_i}^x = \frac{A_{i+1} + A_i}{2} \quad (3.34)$$

$$\text{FAX}(Y) = \overline{A_j}^y = \frac{A_{j+1} + A_j}{2}, \quad (3.35)$$

with the second expression in each equation exposing the notation used in the ocean model code. Additionally, finite difference operators have been introduced

$$\begin{aligned} \text{FDX_NT}(A) &= \frac{A_{i+1} - A_i}{\text{dx}u_{i,j}}. \\ \text{FDY_ET}(A) &= \frac{A_{j+1} - A_j}{\text{dy}u_{i,j}}. \end{aligned} \quad (3.36)$$

These operators are used for fields that live at the north face and east face, respectively, of a tracer cell.

The geopotential contribution in (3.45) is computed using the geopotential values at the tracer points, and so its gradient is located at the tracer cell faces. To have the density multiplier at the same point requires that it be averaged prior to multiplying. Finally, an orthogonal spatial average is required to place the product onto the velocity point. The result is given by

$$\rho' \nabla_s \Phi \approx \hat{\mathbf{x}} \left[\text{FAY}[\delta_i \Phi \text{FAX}(\rho')] / \text{dx}u_{i,j} \right] + \hat{\mathbf{y}} \left[\text{FAX}[\delta_j \Phi \text{FAY}(\rho')] / \text{dy}u_{i,j} \right]. \quad (3.37)$$

Exposing just the pressure gradient force, the corresponding zonal and meridional momentum equations for the B-grid Boussinesq fluid take the form

$$\partial_t (u \rho_o \text{dzu})^{\text{pressure}} = -\text{dzu} \left(\text{FDX_NT}(\text{FAY}(p_a + p_{\text{surf}} + p')) + \text{FAY}[\delta_i \Phi \text{FAX}(\rho')] / \text{dx}u_{i,j} \right) \quad (3.38)$$

$$\partial_t (v \rho_o \text{dzu})^{\text{pressure}} = -\text{dzu} \left(\text{FDY_ET}(\text{FAX}(p_a + p_{\text{surf}} + p')) + \text{FAX}[\delta_j \Phi \text{FAY}(\rho')] / \text{dy}u_{i,j} \right). \quad (3.39)$$

3.3.2 Pressure based vertical coordinates

The aim is to discretize the pressure gradient body force written in the form (3.27)

$$\nabla_s p + \rho \nabla_s \Phi = (\rho / \rho_o) \nabla (p_b + \rho_o \Phi_b) + \rho \nabla_s \Phi' - (\rho' / \rho_o) \nabla_s p \quad (3.40)$$

and to do so in a manner analogous to the Boussinesq case. In particular, we consider here the slow three dimensional contribution $\rho \nabla_s \Phi' - (\rho' / \rho_o) \nabla_s p$ and write for the pressure term

$$\rho' \nabla_s p \approx \hat{\mathbf{x}} \left[\text{FAY}[\delta_i p \text{FAX}(\rho')] / \text{dx}u_{i,j} \right] + \hat{\mathbf{y}} \left[\text{FAX}[\delta_j p \text{FAY}(\rho')] / \text{dy}u_{i,j} \right], \quad (3.41)$$

which is analogous to the discrete $\rho' \nabla_s \Phi$ contribution in equation (3.37). The geopotential term is discretized as

$$\rho \nabla_s \Phi' \approx \hat{\mathbf{x}} \rho \text{FDX_NT}(\text{FAY}(\Phi')) + \hat{\mathbf{y}} \rho \text{FDY_ET}(\text{FAX}(\Phi')), \quad (3.42)$$

which is analogous to the discrete version of $\nabla_s p'$ in equation (3.33). Note that the density ρ in equation (3.42) is centered on the velocity cell.

Exposing just the pressure gradient force, the zonal and meridional momentum equations for the B-grid non-Boussinesq fluid take the form

$$\partial_t (u \rho_o \text{dzu})^{\text{pressure}} = -\rho_o \text{dzu} \text{FDX_NT}(\text{FAY}(p_b / \rho_o + \Phi_b + \Phi')) + \text{dzu} \text{FAY}[\delta_i p \text{FAX}(\rho' / \rho_o)] / \text{dx}u_{i,j} \quad (3.43)$$

$$\partial_t (v \rho_o \text{dzu})^{\text{pressure}} = -\rho_o \text{dzu} \text{FDY_ET}(\text{FAX}(p_b / \rho_o + \Phi_b + \Phi')) + \text{dzu} \text{FAX}[\delta_j p \text{FAY}(\rho' / \rho_o)] / \text{dy}u_{i,j}. \quad (3.44)$$

3.4 Pressure gradient body force in C-grid MOM

We now detail how the pressure gradient body force is represented in the C-grid generalized level coordinate version of MOM.

3.4.1 Depth based vertical coordinates

The aim here is to discretize the pressure gradient body force written in the forms (3.18) and (3.17)

$$\nabla_s p + \rho \nabla_s \Phi = \nabla(p_a + p_{\text{surf}}) + \nabla_s p' + \rho' \nabla_s \Phi, \quad (3.45)$$

where p_{surf} and p' are defined according to equations (3.19) and (3.20), respectively. Our focus here is the slowly evolving three dimensional terms $\nabla_s p' + \rho' \nabla_s \Phi$. The first term is straightforward to discretize according to the assumptions regarding the placement of pressure on a C-grid (Figure 9.2)

$$\nabla_s p' \approx \hat{x} \text{FDX_T}(p') + \hat{y} \text{FDY_T}(p'), \quad (3.46)$$

where the averaging operators required for the B-grid are absent. Additionally, finite difference operators have been introduced

$$\begin{aligned} \text{FDX_T}(A) &= \frac{A_{i+1} - A_i}{dx_{te_{i,j}}} \\ \text{FDY_T}(A) &= \frac{A_{j+1} - A_j}{dy_{tn_{i,j}}}, \end{aligned} \quad (3.47)$$

where the grid distances are defined in Figure 9.7. These operators are used for fields that live at the tracer point.

The geopotential contribution in (3.45) is computed using the geopotential values at the tracer points, and so its gradient is located at the tracer cell faces. To have the density multiplier at the same point requires that it be averaged prior to multiplying, so that

$$\rho' \nabla_s \Phi \approx \hat{x} \text{FAX}(\rho') \text{FDX_T}(\Phi) + \hat{y} \text{FAY}(\rho') \text{FDY_T}(\Phi). \quad (3.48)$$

Exposing just the pressure gradient force, the corresponding zonal and meridional momentum equations for the C-grid Boussinesq fluid take the form

$$\partial_t(u \rho_o dzte)^{\text{pressure}} = -dzte [\text{FDX_T}(p_a + p_{\text{surf}} + p') + \text{FAX}(\rho') \text{FDX_T}(\Phi)] \quad (3.49)$$

$$\partial_t(v \rho_o dztn)^{\text{pressure}} = -dztn [\text{FDY_T}(p_a + p_{\text{surf}} + p') + \text{FAY}(\rho') \text{FDY_T}(\Phi)]. \quad (3.50)$$

3.4.2 Pressure based vertical coordinates

The aim is to discretize the pressure gradient body force written in the form (3.27)

$$\nabla_s p + \rho \nabla_s \Phi = (\rho/\rho_o) \nabla(p_b + \rho_o \Phi_b) + \rho \nabla_s \Phi' - (\rho'/\rho_o) \nabla_s p \quad (3.51)$$

and to do so in a manner analogous to the Boussinesq case. In particular, we consider here the slow three dimensional contribution $\rho \nabla_s \Phi' - (\rho'/\rho_o) \nabla_s p$ and write for the pressure term

$$\rho' \nabla_s p \approx \hat{x} \text{FAX}(\rho') \text{FDX_T}(p) + \hat{y} \text{FAY}(\rho') \text{FDY_T}(p), \quad (3.52)$$

which is analogous to the discrete $\rho' \nabla_s \Phi$ contribution in equation (3.48). The geopotential term is discretized as

$$\rho \nabla_s \Phi' \approx \hat{x} \rho \text{FDX_T}(\Phi') + \hat{y} \rho \text{FDY_T}(\Phi'). \quad (3.53)$$

Importantly, the density factor in each term is on the respective tracer cell faces.

Exposing just the pressure gradient force, the corresponding zonal and meridional momentum equations for the C-grid non-Boussinesq fluid take the form

$$\partial_t(u \rho_o dzte)^{\text{pressure}} = -\rho_o dzte \text{FDX_T}(p_b/\rho_o + \Phi_b + \Phi') + dzte \text{FAX}(\rho'/\rho_o) \text{FDX_T}(p) \quad (3.54)$$

$$\partial_t(v \rho_o dztn)^{\text{pressure}} = -\rho_o dztn \text{FDY_T}(p_b/\rho_o + \Phi_b + \Phi') + dztn \text{FAY}(\rho'/\rho_o) \text{FDY_T}(p). \quad (3.55)$$

PARAMETERIZATIONS WITH GENERALIZED LEVEL COORDINATES

Contents

4.1	Friction	75
4.1.1	Vertical friction	76
4.1.2	A comment on nonlinear vertical friction	76
4.1.3	Lateral friction	77
4.1.4	Bottom stress	78
4.1.5	Summary of the linear momentum budget	78
4.2	Diffusion and skew diffusion	79
4.2.1	Vertical diffusion	79
4.2.2	Horizontal diffusion	79
4.2.3	Neutral physics	79
4.2.3.1	The velocity field from Gent and McWilliams (1990)	80
4.2.3.2	Neutral slopes	81
4.2.3.3	Fluxes for neutral diffusion	82
4.2.3.4	Fluxes for skew diffusion	83
4.2.3.5	Summary of the neutral fluxes	84

The parameterization of subgrid scale (SGS) processes is of fundamental importance to ocean models. Details of how these processes are parameterized depend on the choice of vertical coordinates. The purpose of this chapter is to describe how various SGS parameterizations are formulated with generalized level coordinates of MOM. As we will see, by diagnosing the vertical grid cell thicknesses according to the methods described in Section 10.3, parameterizations implemented in the geopotential MOM4.0 code remain algorithmically unaltered when converting to the generalized level coordinate MOM.

4.1 Friction

The convergence of frictional stress leads to a friction force acting on fluid parcels. The purpose of this section is to detail the form of friction appearing in the generalized level coordinates of MOM. For this purpose, we follow much of the discussion in Chapter 17 of Griffies (2004). In particular, Section 17.3.4 leads us to take the physical components to the frictional stress tensor in the form

$$\boldsymbol{\tau} = \begin{pmatrix} \tau^{xx} & \tau^{xy} & \rho \kappa u_{,z} \\ \tau^{xy} & -\tau^{xx} & \rho \kappa v_{,z} \\ \rho \kappa u_{,z} & \rho \kappa v_{,z} & 0 \end{pmatrix}, \quad (4.1)$$

where κ is a non-negative viscosity with units $\text{m}^2 \text{s}^{-1}$. Taking $\tau^{33} = 0$ is consistent with use of the hydrostatic approximation, which reduces the vertical momentum equation to the inviscid hydrostatic balance. We comment in Section 4.1.3 on the form of the two-dimensional transverse elements τ^{xx} and τ^{xy} . Further details of lateral friction are given in Chapter 25.

4.1.1 Vertical friction

As the gravitational force is so critical to stratified fluids close to a hydrostatic balance, it is typical in ocean modelling to single out the vertical direction. In particular, closures for the unresolved vertical exchange of momentum are usually taken to be proportional to the vertical derivative, or shear, of the horizontal velocity field. This argument leads to the form of the stress tensor given by equation (4.1). For a generalized level coordinate model, the vertical shear elements take the form

$$\rho \kappa \mathbf{u}_{,z} = \rho \kappa s_{,z} \mathbf{u}_{,s}. \quad (4.2)$$

In MOM, the left hand side of this expression is numerically evaluated for purposes of computing the vertical shear. That is, vertical derivatives are computed for arbitrary vertical coordinates just as in geopotential coordinates. This result follows by diagnosing the vertical grid cell thicknesses using the methods described in Section 10.3, where we make use of the relation between vertical coordinates

$$dz = z_{,s} ds. \quad (4.3)$$

Now return to the thickness weighted momentum budget for a grid cell discussed in Section 2.9. The above considerations lead us to write the frictional stress acting on a generalized surface as

$$\begin{aligned} z_{,s} \nabla s \cdot \boldsymbol{\tau} &= (\hat{\mathbf{z}} - \mathbf{S}) \cdot \boldsymbol{\tau} \\ &\approx \hat{\mathbf{z}} \cdot \boldsymbol{\tau} \\ &= \rho \kappa \mathbf{u}_{,z}. \end{aligned} \quad (4.4)$$

The second step used the small angle approximation to drop the extra slope term. Alternatively, we can interpret the dia-surface frictional stress $z_{,s} \nabla s \cdot \boldsymbol{\tau}$ as parameterized by $\rho \kappa \mathbf{u}_{,z}$. Either way, the result (4.4) is the form that vertical frictional stress is implemented in MOM.

4.1.2 A comment on nonlinear vertical friction

As noted above, we choose in MOM to implement vertical friction, and vertical tracer diffusion (Section 4.2.1) just as in a geopotential coordinate model. This method is facilitated by diagnosing the vertical thickness of a grid cell according to equation (4.3) (see Section 10.3), prior to computing vertical derivatives.

We now mention an alternative method, *not implemented in MOM*, since this method is often seen in the literature. The alternative is to compute the vertical shear according to the right hand side of equation (4.2). The density weighted inverse specific thickness $\rho/z_{,s}$ adds a nonlinear term to the vertical friction, and this complicates the numerical treatment (Hallberg, 2000). It is reasonable to approximate this factor by a constant for the dimensionful quasi-horizontal coordinates considered in Sections 5.1 and 5.2.¹ For the Boussinesq case with depth-based vertical coordinates, this approximation results in

$$\rho \kappa / z_{,s} \approx \rho_o \kappa, \quad (4.5)$$

where $z_{,s} \approx 1$ follows from the results for all but the sigma coordinate in Table 5.1. The vertical friction therefore becomes

$$\begin{aligned} (\rho \kappa \mathbf{u}_{,z})_{,z} &\approx \rho_o s_{,z} (\kappa s_{,z} \mathbf{u}_{,s})_{,s} \\ &\approx \rho_o (\kappa \mathbf{u}_{,s})_{,s}. \end{aligned} \quad (4.6)$$

Likewise, dimensionful pressure-based coordinates used for non-Boussinesq fluids have

$$\rho \kappa / z_{,s} \approx -g \rho_o^2 \kappa, \quad (4.7)$$

¹Terrain following sigma coordinates, which are dimensionless, are notable exceptions to this result.

as follows for all but the sigma coordinate in Table 5.2. The vertical friction therefore becomes

$$(\rho \kappa \mathbf{u}_{,z})_{,z} \approx \rho_o (g \rho_o)^2 (\kappa \mathbf{u}_{,s})_{,s}. \quad (4.8)$$

The above approximations are well motivated physically since the value of the vertical viscosity is not known to better than 10%, and the above approximations are well within this range for vertical coordinates whose iso-surfaces are quasi-horizontal. Similar arguments were presented by Losch et al. (2004). Additionally, the approximations are very convenient numerically since they allow us to continue implementing vertical physical processes in a linear manner as traditionally handled in z-models. Such facilitates straightforward time implicit methods to stably handle large vertical viscosities. Without these approximations, or without use of the geopotential-based approach described above in Section 4.1.1, vertical physical processes are nonlinear. Arbitrarily stable numerical methods for such processes require an iterative scheme such as that discussed by Hallberg (2000) employed in isopycnal models.

4.1.3 Lateral friction

There is no fundamental theory to prescribe the form of lateral friction at the resolutions available for large scale ocean modelling. Indeed, many argue that the form commonly used in models is wrong (Holloway, 1992). We take the perspective that lateral friction in ocean models provides a numerical closure. This perspective motivates us to prescribe friction in a manner that maintains basic symmetry properties of the physical system, and which is convenient to implement.

The deformation rates are a basic element of the lateral frictional stress. Using generalized orthogonal horizontal coordinates and z for the vertical, the deformation rates given in Section 17.7.1 of Griffies (2004) take the form

$$e_T = (dy)(u/dy)_{,x} - (dx)(v/dx)_{,y} \quad (4.9)$$

$$e_S = (dx)(u/dx)_{,y} + (dy)(v/dy)_{,x} \quad (4.10)$$

where dx and dy are the infinitesimal horizontal grid increments. Consistent with lateral friction being considered a numerical closure, we place no fundamental importance on the horizontal derivatives being taken on constant z surfaces. Hence, we propose to use the *same* mathematical form for the deformation rates regardless the vertical coordinate. That is, for the generalized level coordinates used in MOM, the deformation rates are computed according to the lateral strains within surfaces of constant vertical coordinate.

As shown in the Appendix to Griffies and Hallberg (2000), and further detailed in Section 17.10 of Griffies (2004), the divergence of the thickness weighted lateral stress within a layer, $\nabla \cdot \tau$, leads to the thickness weighted forces per volume acting in the generalized horizontal directions

$$\begin{aligned} dz \rho F^x &= (dy)^{-2} [(dy)^2 dz \tau^{xx}]_{,x} + (dx)^{-2} [(dx)^2 dz \tau^{xy}]_{,y} \\ dz \rho F^y &= (dx)^{-2} [(dx)^2 dz \tau^{yy}]_{,y} + (dy)^{-2} [(dy)^2 dz \tau^{xy}]_{,x}. \end{aligned} \quad (4.11)$$

We extend the forms for the stress tensor given in Chapter 17 of Griffies (2004) by assuming that all horizontal derivatives appearing in the stress tensor are taken along surfaces of constant generalized level coordinate. Notably, the forms all have an overall density factor, such as the general form given by equation (17.119) in Griffies (2004)

$$\begin{pmatrix} \tau^{xx} & \tau^{xy} \\ \tau^{xy} & -\tau^{xx} \end{pmatrix} = \rho \begin{pmatrix} (A e_T + D \Delta R_x^x) & (A e_S + D \Delta R_y^x) \\ (A e_S + D \Delta R_x^y) & (-A e_T + D \Delta R_y^y) \end{pmatrix}, \quad (4.12)$$

with R a rotation matrix

$$R_{(n)}^{(m)} = \begin{pmatrix} \sin 2\theta & -\cos 2\theta \\ -\cos 2\theta & -\sin 2\theta \end{pmatrix}, \quad (4.13)$$

$A \geq 0$ is a non-negative viscosity weighting the isotropic stress tensor, and $D \geq 0$ is a non-negative viscosity weighting the anisotropic stress tensor. For the Boussinesq fluid, the density factor in the stress tensor is

set to the constant ρ_o . Furthermore, the specific thickness z_s is a depth independent function when using the depth-based Boussinesq vertical coordinates (Table 5.1 in Section 5.1). For the mass conserving non-Boussinesq pressure-based vertical coordinates (Table 5.2 in Section 5.2), the density weighted specific thickness ρz_s is a depth independent function, which then simplifies the density weighted thickness of a grid cell $\rho dz = \rho z_s ds$. These results are familiar from the analogous simplifications arising for other terms in the scalar and momentum budgets discussed in Chapter 2. We consider the needs of spatial discretization for the B-grid and C-grid in Chapter 25.

4.1.4 Bottom stress

We exposed the form of bottom stress in Section 2.9.3, and it generally leads to a bottom force given by

$$\begin{aligned} \mathbf{F}_{\text{bottom}} &= - \iint_{z=-H} dx dy \nabla(z+H) \cdot \boldsymbol{\tau} \\ &= - \iint_{z=-H} dx dy \boldsymbol{\tau}^{\text{bottom}}. \end{aligned} \quad (4.14)$$

A common method to parameterize this force is to consider unresolved small scale processes to give rise to a dissipative drag written in the form

$$\mathbf{F}_{\text{bottom}} = - \iint_{z=-H} dx dy [\rho C_D \mathbf{u}_b (\mathbf{u}_b^2 + \mathbf{u}_{\text{tide}}^2)^{1/2}], \quad (4.15)$$

where it is only the horizontal bottom force that appears in hydrostatic models. In this equation, C_D is a dimensionless drag coefficient with common values taken as

$$C_D \approx 10^{-3}. \quad (4.16)$$

Because the precise value of C_D is not well known, the product ρC_D is approximated in MOM as

$$\rho C_D \approx \rho_o C_D. \quad (4.17)$$

The velocity \mathbf{u}_{tide} represents a residual horizontal velocity that is not resolved in models running without tidal forcing. Hence, even with the bottom flow weak, the residual velocity keeps the drag nontrivial. A common value for the residual velocity is

$$|\mathbf{u}_{\text{tide}}| \approx 0.05 \text{ m s}^{-1}. \quad (4.18)$$

The velocity \mathbf{u}_b is formally the velocity within the bottom boundary layer, but it is commonly taken in models as the velocity at the grid cell adjacent to the bottom. Note that our assumed form of the unresolved bottom stresses take the form of a *bottom drag*. See [Holloway \(1999\)](#) for more general forms where the unresolved bottom stresses may act to accelerate the resolved flow field.

4.1.5 Summary of the linear momentum budget

The horizontal linear momentum budgets for interior, bottom, and surface grid cells are given by equations (2.226), (2.229), and (2.234). We rewrite them here for future reference, incorporating the more detailed form for friction appropriate for hydrostatic models

$$\begin{aligned} [\partial_t + (f + \mathcal{M})\hat{\mathbf{z}} \wedge](dz \rho \mathbf{u}) &= \rho dz \mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u}(\rho \mathbf{u})] \\ &\quad - dz(\nabla_s p + \rho \nabla_s \Phi) + dz \rho \mathbf{F} \\ &\quad - [\rho(w^{(s)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_{k-1}} \\ &\quad + [\rho(w^{(s)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_k} \end{aligned} \quad (4.19)$$

$$\begin{aligned}
[\partial_t + (f + \mathcal{M})\hat{\mathbf{z}} \wedge](dz\rho\mathbf{u}) &= \rho dz\mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] \\
&\quad - dz(\nabla_s p + \rho \nabla_s \Phi) + dz\rho\mathbf{F} \\
&\quad - [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{kbot}-1} \\
&\quad - \boldsymbol{\tau}^{\text{bottom}}
\end{aligned} \tag{4.20}$$

$$\begin{aligned}
[\partial_t + (f + \mathcal{M})\hat{\mathbf{z}} \wedge](dz\rho\mathbf{u}) &= \rho dz\mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] \\
&\quad - dz(\nabla_s p + \rho \nabla_s \Phi) + dz\rho\mathbf{F} \\
&\quad + [\boldsymbol{\tau}^{\text{wind}} + Q_m\mathbf{u}_m] \\
&\quad + [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{k=1}}.
\end{aligned} \tag{4.21}$$

As discussed in Section 2.8.2, we prefer to work with the pressure gradient body force acting within the grid cell of a primitive equation ocean model, rather than the accumulation of contact pressures acting at the faces. This formulation in terms of body forces is convenient in a hydrostatic fluid as it facilitates a numerical treatment of pressure in the discrete ocean climate model (Section 3.3).

4.2 Diffusion and skew diffusion

Some of the results for friction are also applicable for diffusion. However, neutral diffusion and skew diffusion require some added considerations.

4.2.1 Vertical diffusion

Dianeutral tracer transport is often parameterized with a diffusive closure, and these closures require the dianeutral derivative of tracer. For most parameterizations, dianeutral derivatives are computed with a vertical derivative (see Section 7.4 of Griffies (2004)), and these derivatives are computed in MOM just as done for the velocity shears for vertical friction described in Section 4.1.1. Hence, vertical diffusion of tracer concentration is implemented by a direct computation of the finite differenced vertical derivative

$$(\rho\kappa C_{,z})_{,z} \approx \delta_z(\rho\kappa C_{,z}) \tag{4.22}$$

where C is the tracer concentration and κ is the vertical diffusivity.

4.2.2 Horizontal diffusion

Horizontal diffusion is used infrequently in the interior regions of the ocean in climate simulations with MOM4, since neutral physics is preferred for physical reasons. However, near the surface boundary, arguments presented in Treguier et al. (1997), Ferrari et al. (2008), and Ferrari et al. (2010) motivate orienting lateral diffusive processes along surfaces of constant generalized level coordinate when in the surface turbulent boundary, and along topography following coordinates for the bottom turbulent boundary layer. Hence, it is useful to consider the form that horizontal diffusion takes in generalized level coordinates.

When computing the horizontal fluxes as downgradient along surfaces of constant vertical coordinate s , we consider

$$\rho\mathbf{F} = -\rho A \nabla_s C, \tag{4.23}$$

with A a horizontal diffusivity. The thickness weighted horizontal diffusion operator is therefore given by

$$R^{\text{horz}} = -\nabla_s \cdot (dz\rho\mathbf{F}). \tag{4.24}$$

4.2.3 Neutral physics

As for horizontal and vertical diffusion, we compute the tracer flux from neutral physics as $\rho\mathbf{F}$, where \mathbf{F} is the tracer concentration flux formulated as in a Boussinesq model, and ρ is the *in situ* density for a non-Boussinesq model and ρ_o for a Boussinesq model. This approach requires a bit of justification for the neutral skewness from Gent and McWilliams (1990), and we provide such in this section. The bottomline

is there are no nontrivial issues involved with implementing this scheme in a non-Boussinesq model. The only issue arising with generalized level coordinates thus relates to the computation of neutral direction slopes.

Neutral diffusion fluxes are oriented relative to neutral directions. Hence, the slope of the neutral direction relative to the surface of constant vertical coordinate is required to construct the neutral diffusion flux.

The scheme of [Gent and McWilliams \(1990\)](#) requires the slope of the neutral direction relative to the geopotential surface, since this slope provides a measure of the available potential energy. For simplicity, we use the same slope for both neutral diffusion and skew diffusion in MOM. Doing so facilitates a straightforward extension of the neutral physics technology employed in the z-model MOM4.0 to the generalized coordinates supported for later versions of MOM. It however produces a modified [Gent and McWilliams \(1990\)](#) scheme in which skew diffusion relaxes neutral directions toward surfaces of constant vertical coordinate rather than constant geopotential surfaces. For surfaces of constant vertical coordinate that are quasi-horizontal, the modified skew diffusion scheme should act in a manner quite similar to that in a z-model. However, for the terrain following coordinates $\sigma^{(z)}$ and $\sigma^{(p)}$, novel issues arise which have *not* been considered in the MOM formulation of [Gent and McWilliams \(1990\)](#) skewness. Hence, the use of neutral physics parameterizations with terrain following vertical coordinates is *not* recommended in MOM.

4.2.3.1 The velocity field from [Gent and McWilliams \(1990\)](#)

As formulated by [Gent et al. \(1995\)](#), the parameterization of [Gent and McWilliams \(1990\)](#) is typically considered from the perspective of a Boussinesq ocean model. For the purposes of advective transport of tracer, we add a non-divergent velocity $\mathbf{v}^* = \nabla \wedge \Psi$ to the non-divergent resolved scale velocity \mathbf{v} . The parameterized vector streamfunction is given by

$$\Psi = -\kappa \mathbf{S} \wedge \hat{\mathbf{z}}, \quad (4.25)$$

where \mathbf{S} is the neutral slope and $\kappa > 0$ is a kinematic diffusivity. In this way, volume conservation remains unchanged, thus removing the need to modify the model's kinematic relations used to diagnose the vertical velocity component w .

The above results can be seen from a finite volume perspective by considering the volume conservation equation for an interior model grid cell (Section 2.10), in which

$$\partial_t(dz) = dz \mathcal{S}^{(V)} - \nabla_s \cdot (dz \mathbf{u}) - (w^{(z)})_{s=s_{k-1}} + (w^{(z)})_{s=s_k}, \quad (4.26)$$

where $\mathcal{S}^{(V)}$ is a volume source, and $w^{(z)}$ is the dia-surface velocity component defined in Section 2.1.4. Use of the [Gent et al. \(1995\)](#) advective velocity

$$\mathbf{u}^* = -\partial_z(\kappa \mathbf{S}) \quad (4.27)$$

leads to the finite volume result

$$\int_{z_k}^{z_{k-1}} dz \mathbf{u}^* = -(\kappa \mathbf{S})_{k-1} + (\kappa \mathbf{S})_k, \quad (4.28)$$

which renders

$$\begin{aligned} -\nabla_s \cdot \left(\int_{z_k}^{z_{k-1}} dz \mathbf{u}^* \right) - w_{k-1}^* + w_k^* &= \nabla_s \cdot (\kappa \mathbf{S})_{k-1} - \nabla_s \cdot (\kappa \mathbf{S})_k - w_{k-1}^* + w_k^* \\ &= 0. \end{aligned} \quad (4.29)$$

Hence, there is no modification of the volume in a grid cell from the [Gent et al. \(1995\)](#) velocity field.

We now extend the formulation to a non-Boussinesq fluid, in which case the mass conservation takes the form (see equation (2.155) in Section 2.6.2)

$$\partial_t(dz\rho) = dz\rho \mathcal{S}^{(M)} - \nabla_s \cdot (dz\rho \mathbf{u}) - (\rho w^{(z)})_{s=s_{k-1}} + (\rho w^{(z)})_{s=s_k}, \quad (4.30)$$

with $\mathcal{S}^{(M)}$ a mass source. Define a density weighted horizontal advection velocity according to

$$\rho \mathbf{u}^* = -\partial_z (\rho \kappa \mathbf{S}), \quad (4.31)$$

in which case the vector streamfunction from the Boussinesq case is extended to the non-Boussinesq merely by introducing a density weighting

$$\rho \Psi = -\rho \kappa \mathbf{S} \wedge \hat{\mathbf{z}}. \quad (4.32)$$

This result then leads to

$$\begin{aligned} -\nabla_s \cdot \left(\int_{z_k}^{z_{k-1}} \rho \, dz \, \mathbf{u}^* \right) - (\rho w^*)_{s=s_{k-1}} + (\rho w^*)_{s=s_k} = \\ \nabla_s \cdot (\rho \kappa \mathbf{S})_{k-1} - \nabla_s \cdot (\rho \kappa \mathbf{S})_k - (\rho w^*)_{s=s_{k-1}} + (\rho w^*)_{s=s_k} \\ = 0, \end{aligned} \quad (4.33)$$

which means there is no modification to the mass of a grid cell through the introduction of the non-Boussinesq [Gent et al. \(1995\)](#) parameterization.

Note that in the continuum, the above finite volume results mean that the non-Boussinesq [Gent et al. \(1995\)](#) velocity satisfies

$$\nabla_s \cdot (\rho z_{,s} \mathbf{u}^*) + \partial_s (\rho z_{,s} w^*) = 0, \quad (4.34)$$

where s is the vertical coordinate. For the geopotential case with $s = z$, we have

$$\nabla_z \cdot (\rho \mathbf{u}^*) + \partial_z (\rho w^*) = 0, \quad (4.35)$$

which reduces to the familiar non-divergence condition

$$\nabla_z \cdot \mathbf{u}^* + \partial_z w^* = 0 \quad (4.36)$$

in the Boussinesq case.

4.2.3.2 Neutral slopes

A key to the implementation of neutral physics is the slope of a neutral direction relative to either the geopotential or a surface of constant generalized level coordinate. Implicit in the following is the assumption that the neutral slope is finite relative to each surface.

The neutral slope relative to the geopotential is

$$\begin{aligned} \mathbf{S}_{(\gamma/z)} &= \nabla_\gamma z \\ &= -z_{,\gamma} \nabla_z \gamma \end{aligned} \quad (4.37)$$

with γ the locally referenced potential density. The (γ/z) subscript notation highlights that the neutral slope is computed relative to a geopotential. The relation between this slope and the others can be seen by noting that in generalized vertical coordinates, the horizontal gradient ∇_z is computed using the transformation (6.33) in [Griffies \(2004\)](#) so that

$$\begin{aligned} \mathbf{S}_{(\gamma/z)} &= -z_{,\gamma} (\nabla_s - \mathbf{S}_{(s/z)} \partial_z) \gamma \\ &= \mathbf{S}_{(\gamma/s)} + \mathbf{S}_{(s/z)}. \end{aligned} \quad (4.38)$$

This equation identifies the slope of the vertical coordinate surface relative to the geopotential

$$\begin{aligned} \mathbf{S}_{(s/z)} &= \nabla_s z \\ &= -z_{,s} \nabla_z s \end{aligned} \quad (4.39)$$

and the slope of the neutral direction relative to the vertical coordinate surface

$$\begin{aligned}
 S_{(\gamma/s)} &= \nabla_{\gamma} s \\
 &= -z_{,\gamma} \nabla_s \gamma \\
 &= -z_{,s} s_{,\gamma} \nabla_s \gamma.
 \end{aligned} \tag{4.40}$$

In words, equation (4.38) says that the slope of the neutral direction relative to the geopotential equals to the slope of the neutral direction relative to the vertical coordinate surface plus the slope of the vertical coordinate surface relative to the geopotential. In isopycnal models, the slope $S_{(\gamma/s)}$ is very small for much of the ocean. Except for the sigma coordinates, each of the depth-based and pressure-based vertical coordinates discussed in Sections 5.1 and 5.2 have $S_{(s/z)}$ typically less than 10^{-4} and $S_{(\gamma/s)}$ less than 10^{-2} . For sigma coordinates, both $S_{(\gamma/s)}$ and $S_{(s/z)}$ can be nontrivial in much of the model domain affected by topography.

Figure 4.1 illustrates the relation (4.38) between slopes. This figure shows a particular zonal-vertical slice, with slope given by the tangent of the indicated angle. That is, the x -component of the slope vectors are given by

$$\begin{aligned}
 S_{(s/z)} &= \tan \alpha_{(s/z)} \\
 S_{(\gamma/z)} &= \tan \alpha_{(\gamma/z)} \\
 S_{(\gamma/s)} &= \tan \alpha_{(\gamma/s)}.
 \end{aligned} \tag{4.41}$$

In this example, $S_{(s/z)} < 0$ whereas $S_{(\gamma/z)} > 0$. Note that the angle between the generalized surface and the isopycnal surface, $S_{(\gamma/s)}$, is larger in absolute value for this example than $S_{(\gamma/z)}$. This case may be applicable to certain regions of σ -models, whereas for isopycnal models $S_{(\gamma/s)}$ will generally be smaller than $S_{(\gamma/z)}$. The generally nontrivial angle $S_{(\gamma/s)}$ found in sigma models is yet another reason we do not recommend the use of neutral physics as implemented in MOM along with terrain following vertical coordinates. Significant work is required to ensure a proper treatment of neutral physics with terrain following coordinates, and we are not prepared to support such in MOM.

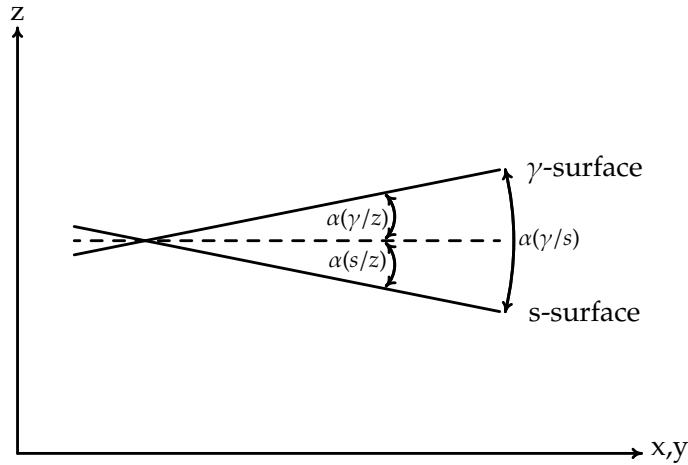


Figure 4.1: Relationship between the slopes of surfaces of constant depth, constant generalized vertical coordinate s , and locally referenced potential density γ . Shown here is a case where the slope is projected onto a single horizontal direction, so that the slope is given by the tangent of the indicated angle.

4.2.3.3 Fluxes for neutral diffusion

The relative slope between the neutral direction and vertical coordinate is required to compute the neutral diffusion flux. We assume here that this slope is small, thus allowing us to approximate the full diffusion

tensor of [Redi \(1982\)](#) with the small slope approximated tensor of [Gent and McWilliams \(1990\)](#). To lend mathematical support for these comments, we start with the neutral diffusion flux as written for the small slope approximation in z -models. As discussed in Section 14.1.4 of [Griffies \(2004\)](#), this flux has horizontal and vertical components given by

$$\mathbf{F}^{(h)} = -A_I \nabla_\gamma C \quad (4.42)$$

$$F^{(z)} = -A_I \mathbf{S}_{(\gamma/z)} \cdot \nabla_\gamma C. \quad (4.43)$$

Converting this flux to a form appropriate for general vertical coordinates requires a transformation of the gradient operator

$$\begin{aligned} \nabla_\gamma &= \nabla_z + \mathbf{S}_{(\gamma/z)} \partial_z \\ &= \nabla_s + [\mathbf{S}_{(\gamma/z)} - \mathbf{S}_{(s/z)}] \partial_z \\ &= \nabla_s + \mathbf{S}_{(\gamma/s)} \partial_z. \end{aligned} \quad (4.44)$$

The third equality used the slope relation (4.38).

As seen in Section 2.6, the thickness weighted tracer budget contains a contribution from the convergence of a SGS flux in the form

$$R = -\nabla_s \cdot (dz \gamma \mathbf{F}) - [\gamma z_{,s} \nabla_s \cdot \mathbf{F}]_{s=s_{k-1}} + [\gamma z_{,s} \nabla_s \cdot \mathbf{F}]_{s=s_k}. \quad (4.45)$$

We are therefore led to consider the dia-surface flux component

$$\begin{aligned} F^{(s)} &= z_{,s} \nabla_s \cdot \mathbf{F} \\ &= (\hat{\mathbf{z}} - \mathbf{S}_{(s/z)}) \cdot \mathbf{F} \\ &= -A_I (\mathbf{S}_{(\gamma/z)} - \mathbf{S}_{(s/z)}) \cdot \nabla_\gamma C \\ &= -A_I \mathbf{S}_{(\gamma/s)} \cdot \nabla_\gamma C \\ &= \mathbf{S}_{(\gamma/s)} \cdot \mathbf{F}^{(h)}. \end{aligned} \quad (4.46)$$

This flux component, as well as the horizontal flux component, take forms isomorphic to those for the specific case of $s = z$ given by equations (4.42) and (4.43). This isomorphism follows from the need to only have information about the relative slope between the generalized surfaces of constant s and the neutral directions.

4.2.3.4 Fluxes for skew diffusion

An arbitrary tracer has a [Gent and McWilliams \(1990\)](#) skew flux in the form

$$\mathbf{F} = A_{\text{gm}} (\mathbf{S}_{(\gamma/z)} C_{,z} - \hat{\mathbf{z}} \mathbf{S}_{(\gamma/z)} \cdot \nabla_z C), \quad (4.47)$$

where A_{gm} is a non-negative skew diffusivity. The horizontal component of this flux is converted to general coordinates via

$$\begin{aligned} \mathbf{F}^{(h)} &= A_{\text{gm}} (\mathbf{S}_{(\gamma/s)} + \mathbf{S}_{(s/z)}) C_{,z} \\ &\approx A_{\text{gm}} \mathbf{S}_{(\gamma/s)} C_{,z}. \end{aligned} \quad (4.48)$$

Consistent with this same approximation, we are led to the dia-surface component of the skew flux

$$\begin{aligned} z_{,s} \nabla_s \cdot \mathbf{F} &= (\hat{\mathbf{z}} - \mathbf{S}_{(s/z)}) \cdot \mathbf{F} \\ &= -A_{\text{gm}} (\mathbf{S}_{(\gamma/z)} \cdot \nabla_z + \mathbf{S}_{(\gamma/z)} \cdot \mathbf{S}_{(s/z)} \partial_z) C \\ &= -A_{\text{gm}} \mathbf{S}_{(\gamma/z)} \cdot (\nabla_s - \mathbf{S}_{(s/z)} \partial_z) C - A_{\text{gm}} \mathbf{S}_{(\gamma/z)} \cdot \mathbf{S}_{(s/z)} \partial_z C \\ &= -A_{\text{gm}} \mathbf{S}_{(\gamma/z)} \cdot \nabla_s C \\ &\approx -A_{\text{gm}} \mathbf{S}_{(\gamma/s)} \cdot \nabla_s C. \end{aligned} \quad (4.49)$$

These approximations are reasonable where $|\mathbf{S}_{(s/z)}|$ is much smaller than $|\mathbf{S}_{(\gamma/z)}|$ if $\mathbf{S}_{(\gamma/z)}$ is nontrivial. When the neutral slope $\mathbf{S}_{(\gamma/z)}$ vanishes, as for regions of zero baroclinicity, this approximation may not be valid when $s \neq z$. However, in regions of vanishing baroclinicity, we expect the error to be of minimal consequence to the simulation since either the z or s based skew fluxes are close to zero. In general, approximating the slope as proposed here leads the modified [Gent and McWilliams \(1990\)](#) scheme to dissipate neutral slopes as they deviate from surfaces of constant generalized vertical coordinate. So long as these surfaces are quasi-horizontal, the modified scheme should perform in a physically relevant manner.

4.2.3.5 Summary of the neutral fluxes

The horizontal and dia-surface components to the small angle neutral diffusion flux take the form

$$\begin{aligned} \mathbf{F}^{(h)} &= -A_I (\nabla_s + \mathbf{S}_{(\gamma/s)} \partial_z) C \\ F^{(s)} &= \mathbf{S}_{(\gamma/s)} \cdot \mathbf{F}^{(h)} \end{aligned} \quad (4.50)$$

where the slope is given by

$$\begin{aligned} \mathbf{S}_{(\gamma/s)} &= \nabla_\gamma s \\ &= -z_{,\gamma} \nabla_s \gamma. \end{aligned} \quad (4.51)$$

The horizontal and dia-surface skew flux components are approximated by

$$\begin{aligned} \mathbf{F}^{(h)} &\approx A_{\text{gm}} \mathbf{S}_{(\gamma/s)} C_{,z} \\ F^{(s)} &\approx -A_{\text{gm}} \mathbf{S}_{(\gamma/s)} \cdot \nabla_s C. \end{aligned} \quad (4.52)$$

Each of these neutral fluxes are isomorphic to the fluxes used in the z -model MOM4.0. This isomorphism enables us to transfer the neutral physics technology from MOM4.0 directly to the generalized level versions of MOM.

DEPTH AND PRESSURE BASED VERTICAL COORDINATES

Contents

5.1	Depth based vertical coordinates	85
5.1.1	Depth coordinate	86
5.1.2	An example of depth coordinates	87
5.1.3	Depth deviation coordinate	87
5.1.4	Zstar coordinate	89
5.1.5	Depth sigma coordinate	90
5.1.6	Summary of the depth based vertical coordinates	91
5.2	Pressure based coordinates	92
5.2.1	Pressure coordinate	92
5.2.2	Pressure deviation coordinate	93
5.2.3	Pstar coordinate	93
5.2.4	Pressure sigma coordinate	94
5.2.5	Summary of the pressure based vertical coordinates	94

The purpose of this chapter is to document issues related to the choice of vertical coordinates. In MOM, only depth-based and pressure-based coordinates are supported. Isopycnal coordinates are not supported. Furthermore, terrain following sigma coordinates are coded in MOM. However, more work is required in MOM to reduce pressure gradient errors (Section 3.2) and consistently employ neutral physics (see [Lemarié et al. \(2012b\)](#) for some intriguing results on these topics of terrain coordinate ocean models). Much in this chapter is derived from lectures of [Griffies \(2005\)](#) at the 2004 GODAE School.

5.1 Depth based vertical coordinates

We use depth based vertical coordinates in this section to discretize the Boussinesq equations.¹ Depth based coordinates are also known as *volume based* coordinates, since for a Boussinesq model that uses depth as the vertical coordinate, the volume of interior grid cells is constant in the absence of sources. Correspondingly, depth based coordinates are naturally suited for Boussinesq fluids.

¹[Greatbatch and McDougall \(2003\)](#) discuss an algorithm for non-Boussinesq dynamics in a z-model. Their methods are implemented in mom4p0a and mom4p0b of [Griffies et al. \(2004\)](#). This approach may be of special use for non-Boussinesq non-hydrostatic z-models. However, when focusing on hydrostatic models as we do here, pressure based vertical coordinates discussed in Section 5.2 are more convenient to realize non-Boussinesq dynamics.

5.1.1 Depth coordinate

With a free surface, the vertical domain over which the z -coordinate

$$s = z \tag{5.1}$$

ranges is given by the time dependent interval

$$-H \leq z \leq \eta. \tag{5.2}$$

Consequently, the sum of the vertical grid cell increments equals to the total depth of the column

$$\sum_k dz = H + \eta. \tag{5.3}$$

The trivial form of the specific thickness $z_{,s} = 1$ greatly simplifies the Boussinesq budgets.

The depth coordinate is very useful for many purposes in global climate modelling, and models based on depth are the most popular ocean climate models. Their advantages include the following.

- Simple numerical methods have been successfully used in this framework.
- For a Boussinesq fluid, the horizontal pressure gradient can be easily represented in an accurate manner.
- The equation of state for ocean water can be accurately represented in a straightforward manner (e.g., [Jackett et al. \(2006\)](#)).
- The upper ocean mixed layer is well parameterized using a z -coordinate.

Unfortunately, these models have some well known disadvantages, which include the following.

- Representation of tracer transport within the quasi-adiabatic interior is cumbersome, with problems becoming more egregious as mesoscale eddies are admitted ([Griffies et al. \(2000b\)](#)).
- Representation and parameterization of bottom boundary layer processes and flow are unnatural.

Grid cells have static vertical increments $ds = dz$ when $s = z$, except for the top where $\partial_t(dz) = \partial_t \eta$. The time dependent vertical range of the coordinate slightly complicates a numerical treatment of the surface cell in z -models (see [Griffies et al. \(2001\)](#) for details of one such treatment).

Placing all changes in ocean thickness in the top grid cell exposes the free surface geopotential coordinate models to two pesky problems. First, when adding fresh water to the ocean, such as for with simulations of land ice melting, the top cell thickens, which means the representation of vertical processes is coarsened. Conversely, the surface cell can be lost (i.e., can become dry) if the free surface depresses below the depth of the top grid cell's bottom face. This is a very inconvenient feature that limits the use of z -coordinates.² In particular, the following studies may require very refined vertical resolution and/or large undulations of the surface height, and so would not be accessible with a conventional free surface z -model.

- Process studies of surface mixing and biological cycling may warrant very refined upper ocean grid cell thickness, some as refined as 1m.
- Realistic tidal fluctuations in some parts of the World Ocean can reach 10m-20m.
- Coastal models tend to require refined vertical resolution to represent shallow coastal processes along the continental shelves and near-shore.
- When coupled to a sea ice model, the weight of the ice will depress the ocean free surface.

²Linearized free surfaces, in which the budgets for tracer and momentum are formulated assuming a constant top cell thickness, avoid problems with vanishing top cells. However, such models do not conserve total tracer or volume, and so are of limited use for long term climate studies (see [Griffies et al. \(2001\)](#) and [Campin et al. \(2004\)](#) for discussion).

5.1.2 An example of depth coordinates

In some of the following discussion, we illustrate aspects of vertical coordinates by diagnosing the values for the coordinates from a realistic z -model run with partial step thicknesses. *Partial steps* have arbitrary thickness set to accurately represent the bottom topography. The partial step technology was introduced by [Adcroft et al. \(1997\)](#) in the C-grid MITgcm, and further discussed by [Pacanowski and Gnanadesikan \(1998\)](#) for the B-grid Modular Ocean Model (MOM). Figure 5.1 compares the representation of topography in a z -model using partial steps as realized in the MOM code of [Griffies et al. \(2004\)](#). Many z -models have incorporated the partial step technology as it provides an important facility to accurately represent flow and waves near topography.

Because of partial steps, the level next to the ocean bottom has grid cell centers that are generally at different depths. Hence, the bottom cell in a partial step z -model computes its pressure gradient with two terms: one due to gradients across cells with the same grid cell index k , and another due to slopes in the bottom topography. Details of the pressure gradient calculation are provided in Chapter 3. All other cells, including the surface, have grid cell centers that are at fixed depths. Figure 5.2 illustrates the lines of constant partial step depth for this model.

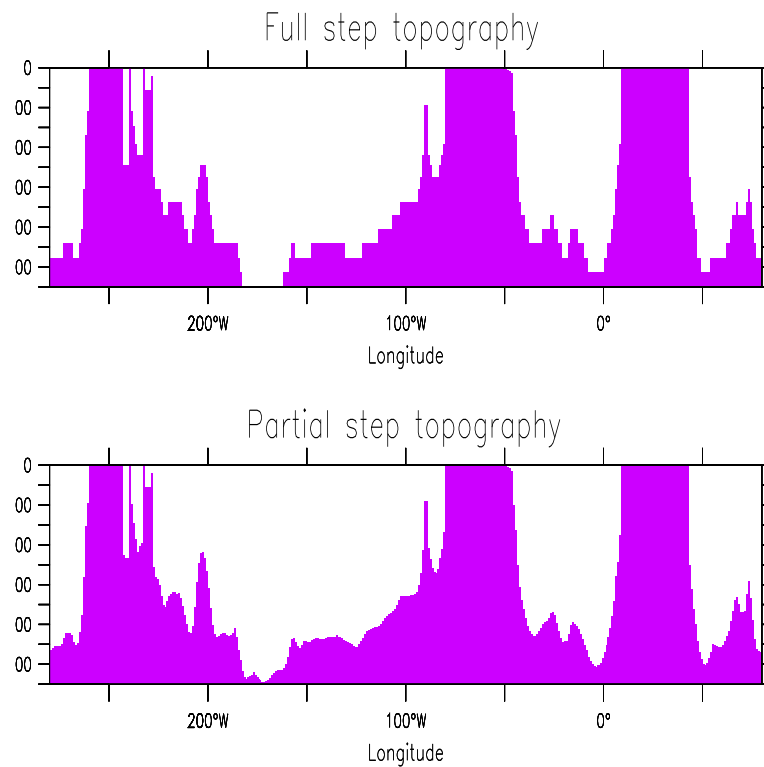


Figure 5.1: Comparison of the partial step versus full step representation of topography as realized in the z -model discussed by [Griffies et al. \(2005\)](#). This vertical section is taken along the equator. The model horizontal grid has one degree latitudinal resolution. The main differences are in the deep ocean in regions where the topographic slope is gradual. Steep sloped regions, and those in the upper ocean with refined vertical resolution, show less distinctions.

5.1.3 Depth deviation coordinate

The depth deviation coordinate

$$s = z - \eta \quad (5.4)$$

removes the restriction on upper ocean grid cell resolution present with $s = z$. That is, $s = 0$ is the time independent coordinate value of the ocean surface, no matter how much the free surface depresses or

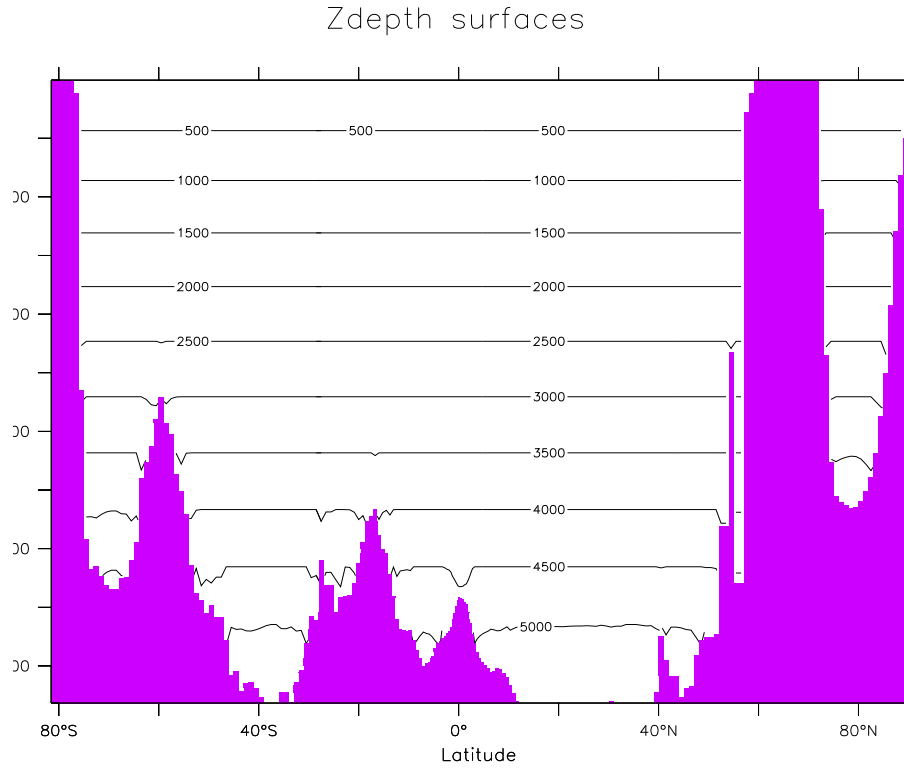


Figure 5.2: This figure contours the depth of grid cell centers used in a modern geopotential ocean model. Deviations from the horizontal occur next to the bottom due to use of a partial bottom step representation of topography, as illustrated in Figure 5.1. In this case, the bottom cell has an arbitrary thickness according to the methods of [Adcroft et al. \(1997\)](#) and [Pacanowski and Gnanadesikan \(1998\)](#). This technology is common in modern geopotential ocean models, as it provides a more faithful and robust representation of the ocean bottom. Shown here is a north-south section along 150°W .

grows. Hence, no surface cells vanish so long as $\eta > -H$. If $\eta < -H$, the bottom topography is exposed, in which case the model's land-sea boundaries are altered. Such necessitates a model that can allow for wetting and drying of grid cells. Alternatively, it requires a model where ocean is extended globally, with infinitesimally thin ocean layers present over land. We do not have such features in MOM.

The depth deviation coordinate ranges between $-(H + \eta) \leq s \leq 0$. The only time dependent interface in s -space is at the bottom of the column. Consequently, by solving the problem at the ocean surface, the deviation coordinate introduces a problem to the ocean bottom where bottom cells can now vanish. To see this problem, discretize the deviation coordinate s according to time independent values s_k . For example, the s_k values can be set as the depths of cells in a model with $s = z$. When η evolves, depth z and $s = z - \eta$ become different, and so the depth of a grid cell must be diagnosed based on the time independent value of s_k and the time dependent surface height

$$z_k = s_k + \eta. \quad (5.5)$$

If the time dependent depth of the upper interface of a bottom grid cell is diagnosed to be deeper than the actual bottom depth $z = -H$, then we know that the bottom grid cell has vanished and so there are problems. To maintain nonvanishing cells requires a limit on how negative η can become. For example, if the upper interface of a bottom cell is -5000m and the bottom interface (at the ocean bottom) is $H = 5005\text{m}$, then the bottom cell is lost if $\eta < -5\text{m}$. This restriction is of some consequence when aiming to use partial bottom steps (see Figure 5.1) along with tides and sea ice. In practice, if one is interested in allowing thick sea ice and nontrivial tidal fluctuations, then it will be necessary to keep the bottom partial steps thicker than roughly 10m - 20m . This is arguably a less onerous constraint on the model's vertical grid spacing than the complementary problem at the ocean surface encountered with the traditional z -coordinate $s = z$.

In summary, grid cells have static grid increments $ds = dz$ for all cells except the bottom. At the bottom, $\partial_t(dz) = \partial_t(ds) = \eta_{,t}$. Hence, the thickness of the bottom cell grows when the surface height grows, and it thins when the surface height becomes negative. The bottom cell can be lost if η becomes too negative. The sum of the vertical increments yields the total depth of the column $\sum_k ds = (H + \eta)$. Because the surface height fluctuations are so much smaller than changes in bottom topography, the depth deviation coordinate appears nearly the same as the depth coordinate when viewed over the full depth range of a typical model such as in Figure 5.2.

The author knows of no model routinely using the depth deviation coordinate. It does appear to have advantages for certain applications over the depth coordinate. However, the $zstar$ coordinate discussed next resolves problems at *both* the top and bottom, and so is clearly preferable. The depth deviation coordinate is *not* implemented in MOM for these reasons.

5.1.4 Zstar coordinate

To overcome problems with vanishing surface and/or bottom cells, we consider the $zstar$ coordinate

$$z^* = H \left(\frac{z - \eta}{H + \eta} \right). \quad (5.6)$$

This coordinate is closely related to the “eta” coordinate used in many atmospheric models (see Black (1994) for a review of eta coordinate atmospheric models). It was originally used in ocean models by Stacey et al. (1995) for studies of tides next to shelves, and it has been recently promoted by Adcroft and Campin (2004) for global climate modelling.

The surfaces of constant z^* are quasi-horizontal. Indeed, the z^* coordinate reduces to z when η is zero. In general, when noting the large differences between undulations of the bottom topography versus undulations in the surface height, it is clear that surfaces constant z^* are very similar to the depth surfaces shown in Figure 5.2. These properties greatly reduce difficulties of computing the horizontal pressure gradient relative to terrain following sigma models discussed next. Additionally, since $z^* = z$ when $\eta = 0$, no flow is spontaneously generated in an unforced ocean starting from rest, regardless the bottom topography.³ This behavior is in contrast to the case with sigma models, where pressure gradient errors in the presence of nontrivial topographic variations can generate nontrivial spontaneous flow from a resting state, depending on the sophistication of the pressure gradient solver.⁴ The quasi-horizontal nature of the coordinate surfaces also facilitates the implementation of neutral physics parameterizations in z^* models using the same techniques as in z -models (see Chapters 13-16 of Griffies (2004) for a discussion of neutral physics in z -models, as well as Section 4.2.3 and Chapter 23 in this document for treatment in MOM).

The range over which z^* varies is time independent

$$-H \leq z^* \leq 0. \quad (5.7)$$

Hence, all cells remain nonvanishing, so long as the surface height maintains $\eta > -H$. This is a minor constraint relative to that encountered on the surface height when using $s = z$ or $s = z - \eta$.

Because z^* has a time independent range, all grid cells have static increments ds , and the sum of the vertical increments yields the time independent ocean depth

$$\sum_k ds = H. \quad (5.8)$$

The z^* coordinate is therefore invisible to undulations of the free surface, since it moves along with the free surface. This property means that no spurious vertical transport is induced across surfaces of constant z^* by the motion of external gravity waves. Such spurious transport can be a problem in z -models, especially those with tidal forcing. Quite generally, the time independent range for the z^* coordinate is a very convenient property that allows for a nearly arbitrary vertical resolution even in the presence of large amplitude fluctuations of the surface height, again so long as $\eta > -H$.

³Because of the use of partial bottom steps, there are two terms contributing to horizontal pressure gradients within the bottom level when $s = z$. As discussed by Pacanowski and Gnanadesikan (1998), these two terms lead to modest pressure gradient errors. These errors, however, are far smaller than those encountered with σ coordinates.

⁴Shchepetkin and McWilliams (2002) provide a thorough discussion of pressure gradient solvers along with methods for reducing the pressure gradient error.

5.1.5 Depth sigma coordinate

The depth-sigma coordinate

$$\begin{aligned}\sigma &= z^*/H \\ &= \left(\frac{z - \eta}{H + \eta} \right)\end{aligned}\tag{5.9}$$

is the canonical *terrain following* coordinate. Figure 5.3 illustrates this coordinate in a realistic model. The sigma coordinate has a long history of use in coastal modelling. For reviews, see [Greatbatch and Mellor \(1999\)](#) and [Ezer et al. \(2002\)](#). Models based on the sigma coordinate have also been successfully extended to basinwide studies (e.g., [Lemarié et al., 2012b](#)), as well as global work by [Diansky et al. \(2002\)](#).

Just as for z^* , the range over which the sigma coordinate varies is time independent. Here, it is given by the dimensionless range

$$-1 \leq \sigma \leq 0.\tag{5.10}$$

Hence, all cells have static grid increments ds , and the sum of the vertical increments yields unity

$$\sum_k ds = 1.\tag{5.11}$$

So long as the surface height is not depressed deeper than the ocean bottom (i.e., so long as $\eta > -H$), then all cells remain nonvanishing.⁵

Some further key advantages of the sigma coordinate are the following.

- It provides a natural framework to represent bottom influenced flow and to parameterize bottom boundary layer processes.
- Thermodynamic effects associated with the equation of state are well represented with this coordinate.

However, some of the disadvantages are the following:

- As with the z -models, representation of the quasi-adiabatic interior is cumbersome due to numerical truncation errors inducing unphysically large levels of spurious mixing, especially in the presence of vigorous mesoscale eddies. Parameterization of these processes using neutral physics schemes may be more difficult numerically than in the z -models. The reason is that neutral directions generally have slopes less than 1/100 relative to the horizontal, but can have order unity slopes relative to sigma surfaces. The larger relative slopes precludes the *small slope approximation* commonly made with z -model implementations of neutral physics. The small slope approximation provides for simplification of the schemes, and improves computational efficiency.
- Sigma models have difficulty accurately representing the horizontal pressure gradient in the presence of realistic topography, where slopes are commonly larger than 1/100 (see Section 2.8 for a discussion of the pressure gradient calculation).

[Griffies et al. \(2000a\)](#) notes that there are few examples of global climate models running with terrain following vertical coordinates. [Diansky et al. \(2002\)](#) is the only exception known to the author. This situation is largely due to problems representing realistic topography without incurring unacceptable pressure gradient errors, as well as difficulties implementing parameterizations of neutral physical processes. There are notable efforts to resolve these problems, such as the pressure gradient work of [Shchepetkin and McWilliams \(2002\)](#). Continued efforts along these lines may soon facilitate the more common use of terrain following coordinates for global ocean climate modelling. At present, the terrain following sigma coordinate is coded in MOM in hopes that it will motivate researchers to further investigate its utility for ocean modelling.

⁵If $\eta < -H$, besides drying up a region of ocean, the specific thickness $z_s = H + \eta$ changes sign, which signals a singularity in the vertical grid definition. The same problem occurs for the z^* coordinate.

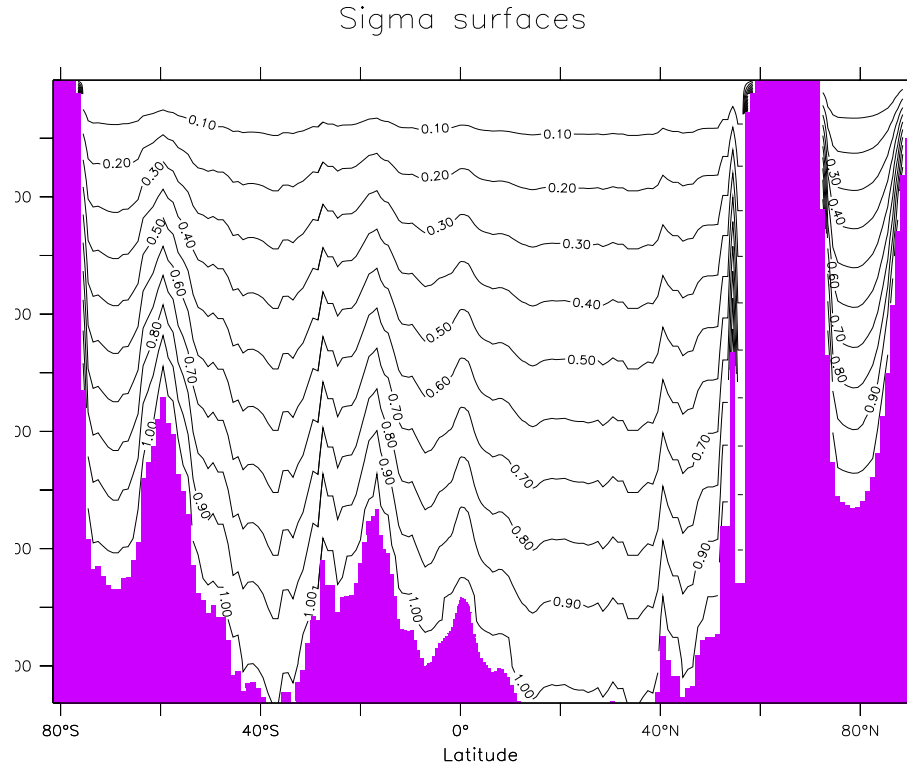


Figure 5.3: Constant sigma surfaces as diagnosed in a z-model. Shown here is a section along $150^\circ W$, as in Figure 5.2. Note the strong variations in the contours, as determined by changes in the bottom topography.

5.1.6 Summary of the depth based vertical coordinates

Depth based vertical coordinates are natural for Boussinesq equations. These coordinates and their specific thicknesses z_s are summarized in Table 5.1. Notably, both the sigma and zstar coordinates have time independent ranges, but time dependent specific thicknesses. In contrast, the depth and depth deviation coordinates have time dependent depth ranges and time independent specific thicknesses. If plotted with the same range as those given in Figure 5.2, surfaces of constant depth deviation and constant zstar are indistinguishable from surfaces of constant depth. This result follows since the surface height undulations are so much smaller than undulations in the bottom topography, thus making the depth deviation and zstar coordinates very close to horizontal in most parts of the ocean.

COORD	DEFINITION	RANGE	z_s
geopotential	z	$-H \leq z \leq \eta$	1
z-deviation	$z' = z - \eta$	$-(H + \eta) \leq z' \leq 0$	1
z-star	$z^* = H(z - \eta)/(H + \eta)$	$-H \leq z^* \leq 0$	$1 + \eta/H$
z-sigma	$\sigma^{(z)} = (z - \eta)/(H + \eta)$	$-1 \leq \sigma \leq 0$	$H + \eta$

Table 5.1: Table of vertical coordinates based on depth. These coordinates are naturally used for discretizing the Boussinesq equations. Note that the specific thickness z_s is depth independent. This property proves to be important for developing numerical algorithms in Section 10.8. The coordinates $s = z$, $s = z^*$, and $s = \sigma^{(z)}$ are coded in MOM, whereas the depth deviation coordinate *is not*.

5.2 Pressure based coordinates

The second class of vertical coordinates that we discuss is based on pressure. Pressure-based coordinates are used to discretize the non-Boussinesq equations, and these coordinates are also known as *mass based* coordinates. This name is based on noting that for a non-Boussinesq fluid using pressure, the mass of interior grid cells is constant without sources (e.g., see equation (2.205)).

Pressure coordinates provide a straightforward way to generalize Boussinesq z -models to non-Boussinesq pressure models (Huang et al., 2001; DeSzoek and Samelson, 2002; Marshall et al., 2004; Losch et al., 2004). The reason is that there is an isomorphism between the Boussinesq equations written in depth based coordinates and non-Boussinesq equations written in pressure based coordinates. The root of this isomorphism is the simplification of the density weighted specific thickness ρz_s for pressure based coordinates. We detail this point in the following discussions.

Pressure based vertical coordinates that we consider include the following:

$$s = p \quad \text{pressure} \quad (5.12)$$

$$s = p - p_a \quad \text{pressure-deviation} \quad (5.13)$$

$$s = \left(\frac{p - p_a}{p_b - p_a} \right) \quad \text{pressure-sigma} \quad (5.14)$$

$$s = p_b^o \left(\frac{p - p_a}{p_b - p_a} \right) \quad \text{pressure-star.} \quad (5.15)$$

In these equations, p is the hydrostatic pressure at some depth within the ocean fluid, p_a is the pressure applied at the ocean surface $z = \eta$ from any media above the ocean, such as the atmosphere and sea ice, p_b is the hydrostatic pressure at the solid-earth lower boundary arising from all fluid above the bottom (ocean water and p_a above the ocean), and p_b^o is a time independent reference pressure, usually taken to be the bottom pressure in a resting ocean.⁶ Since $p_{,z} = -\rho g < 0$ is single signed for the hydrostatic fluid, pressure provides a well defined vertical coordinate. Strengths and weaknesses of the corresponding depth based coordinates also hold for the pressure based coordinates, with the main difference being that pressure based models are non-Boussinesq.

5.2.1 Pressure coordinate

With a free surface, the vertical domain over which the p -coordinate

$$s = p \quad (5.16)$$

ranges is given by

$$p_a \leq p \leq p_b. \quad (5.17)$$

Hence, the surface and bottom boundaries are time dependent, whereas the density weighted specific thickness is constant

$$\rho z_s = -g^{-1} \quad (5.18)$$

where the hydrostatic equation $p_{,z} = -\rho g$ was used. Relation (5.18) is the root of the isomorphism between Boussinesq depth based models and non-Boussinesq pressure based models.

The time dependent range for the pressure coordinate complicates the treatment of both the top and bottom cells. In particular, if the bottom pressure is less than the time independent discrete pressure level at the top interface of the lowest cell, then there is no mass within the bottom cell. Likewise, if the applied pressure is greater than the discrete pressure level at the bottom interface of the top cell, then there is no mass in the top cell. These results mean that grid cells have static vertical coordinate increments $ds = dp$ for all cells except the top and bottom. At the top, $\partial_t(ds) = \partial_t p_a$ and at the bottom $\partial_t(ds) = -\partial_t p_b$. The

⁶Note that equation (11.64) of Griffies (2004) used the time dependent p_b rather than the time independent reference pressure p_b^o . The former vertical coordinate has not been used in practice, and so we focus here on that coordinate defined with the reference pressure p_b^o .

associated mass per unit area in the cells evolves according to $\partial_t(\rho dz) = -g^{-1} \partial_t p_a$ at the surface, and $\partial_t(\rho dz) = g^{-1} \partial_t p_b$ at the bottom. Hence, the mass within the top cell decreases when the applied pressure increases, and the mass in the bottom cell increases when the bottom pressure increases. Both the surface and the bottom cells can therefore vanish depending on the applied and bottom pressures.

The sum of the vertical coordinate increments can be found by noting the total mass per area is given by

$$\begin{aligned} g^{-1}(p_b - p_a) &= \sum \rho dz \\ &= \sum \rho z_{,s} ds \\ &= -g^{-1} \sum ds, \end{aligned} \quad (5.19)$$

thus yielding the time dependent result

$$\sum ds = -(p_b - p_a). \quad (5.20)$$

5.2.2 Pressure deviation coordinate

The pressure deviation coordinate

$$s = p - p_a \quad (5.21)$$

removes the restriction on upper ocean grid cell resolution since $s = 0$ is the time independent value of the ocean surface. That is, this coordinate ranges between

$$0 \leq s \leq p_b - p_a. \quad (5.22)$$

This coordinate is isomorphic to the depth deviation coordinate $s = z - \eta$ discussed in Section 5.1.3, and shares the same limitations which prompt us *not* to have this coordinate coded in MOM.

In summary, grid cells have static vertical coordinate increments ds for all cells except the bottom. At the bottom $\partial_t(ds) = -\partial_t(p_b - p_a)$. The associated mass per unit area in the bottom cell evolves according to $\partial_t(\rho dz) = g^{-1} \partial_t(p_b - p_a)$. As for the pressure coordinate, the sum of the vertical coordinate increments yields

$$\sum ds = -(p_b - p_a). \quad (5.23)$$

5.2.3 Pstar coordinate

The pstar coordinate is given by

$$p^* = p_b^o \left(\frac{p - p_a}{p_b - p_a} \right), \quad (5.24)$$

where p_b^o is a time independent reference pressure. Two possible choices for p_b^o include

$$p_b^o = g \int_{-H}^0 dz \rho^{\text{init}}, \quad (5.25)$$

or the simpler case of

$$p_b^o = g \rho_o H. \quad (5.26)$$

The p^* coordinate is isomorphic to the z^* coordinate, with p^* extending over the time independent range

$$0 \leq p^* \leq p_b^o. \quad (5.27)$$

The sum of the vertical coordinate increments can be found by noting the total mass per area is given by

$$\begin{aligned} g^{-1}(p_b - p_a) &= \sum \rho dz \\ &= \sum \rho z_{,s} ds \\ &= -\left(\frac{p_b - p_a}{g p_b^0}\right) \sum ds, \end{aligned} \quad (5.28)$$

thus yielding the time independent result

$$\sum ds = -p_b^0. \quad (5.29)$$

5.2.4 Pressure sigma coordinate

The pressure-sigma terrain following coordinate

$$\sigma(p) = \left(\frac{p - p_a}{p_b - p_a} \right) \quad (5.30)$$

is the pressure analog to the depth based sigma coordinate $\sigma^{(z)} = (z - \eta)/(H + \eta)$. This coordinate has been used by [Huang et al. \(2001\)](#), and it shares the same advantages and disadvantages as the depth-based sigma coordinate. Grid cells have static vertical coordinate increments ds for all cells. The associated mass per unit area never vanishes in any cell, so long as the bottom pressure is greater than the applied pressure.

The sum of the vertical coordinate increments can be found by noting the total mass per area is given by

$$\begin{aligned} g^{-1}(p_b - p_a) &= \sum \rho dz \\ &= \sum \rho z_{,s} ds \\ &= -g^{-1}(p_b - p_a) \sum ds, \end{aligned} \quad (5.31)$$

thus yielding the time independent result

$$\sum ds = -1. \quad (5.32)$$

5.2.5 Summary of the pressure based vertical coordinates

A technical reason that the pressure based coordinates considered here are so useful for non-Boussinesq hydrostatic modelling is that $\rho z_{,s}$ is either a constant or a two-dimensional field. In contrast, for depth based models $\rho z_{,s}$ is proportional to the three-dimensional *in situ* density ρ , thus necessitating special algorithmic treatment for non-Boussinesq z -models (see [Greatbatch and McDougall \(2003\)](#) and [Griffies \(2004\)](#)). Table 5.2 summarizes the pressure-based coordinates discussed in this section. The pressure and pressure deviation coordinates have time dependent ranges but time independent specific thicknesses $\rho z_{,s}$. The sigma and pstar coordinates have time independent range but time dependent specific thickness.

As Table 5.2 reveals, the specific thickness $z_{,s}$ is negative for the pressure-based coordinates, whereas it is positive for the depth-based coordinate (Table 5.1). The sign change arises since upward motion in a fluid column increases the geopotential coordinate z yet decreases the hydrostatic pressure p . To establish a convention, we assume that the thickness of a grid cell in z space is always positive

$$dz = z_{,s} ds > 0 \quad (5.33)$$

as is the case in the conventional z -models. With $z_{,s} < 0$ for the pressure-based vertical coordinates, the thickness of grid cells in s space is negative

$$ds < 0 \quad \text{for pressure-based coordinates with } z_{,s} < 0. \quad (5.34)$$

COORD	DEFINITION	RANGE	$g \rho z_s$
pressure	p	$p_a \leq p \leq p_b$	-1
p-deviation	$p' = p - p_a$	$0 \leq p' \leq p_b - p_a$	-1
pstar	$p^* = p_b^o (p - p_a) / (p_b - p_a)$	$0 \leq p^* \leq p_b^o$	$-(p_b - p_a) / p_b^o$
p-sigma	$\sigma^{(p)} = (p - p_a) / (p_b - p_a)$	$0 \leq \sigma \leq 1$	$-(p_b - p_a)$

Table 5.2: Table of vertical coordinates based on pressure. These coordinates are naturally used for non-Boussinesq dynamics. Note that the density weighted specific thickness ρz_s is depth independent. This property proves to be important for developing numerical algorithms in Section 10.8. The coordinates $s = p$, $s = p^*$, and $s = \sigma^{(p)}$ are coded in MOM, whereas the pressure deviation coordinate *is not*.

EQUATION OF STATE AND RELATED QUANTITIES

Contents

6.1	Introduction	97
6.2	Linear equation of state	99
6.3	The two realistic equations of state	99
6.3.1	The Jackett et al. (2006) equation of state	99
6.3.2	TEOS-10 equation of state	102

The purpose of this chapter is to present features of the equation of state used in MOM, with the discussion here an extension of that given in [Griffies et al. \(2004\)](#). The following summarizes the realistic equations of state available in recent versions of MOM.

- MOM4.0 uses the [McDougall et al. \(2003\)](#) equation of state.
- MOM4p1 uses the [Jackett et al. \(2006\)](#) equation of state.
- MOM as of the 2012 release uses either of the [Jackett et al. \(2006\)](#) equation (retained for legacy purposes) or the [IOC et al. \(2010\)](#) version recommended for new simulations.

The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_density.F90

6.1 Introduction

It is important that the equation of state be accurate over the range of temperature, salinity, and pressure values occurring in ocean simulations. Reasons for needing such accuracy include the following.

- Density is needed to compute the hydrostatic pressure, whose horizontal gradients drive ocean currents in the primitive equations.
- The locally referenced vertical derivative of density determines the static stability of a vertical fluid column. This stability determines conditions for convective instability and is used to compute Richardson numbers necessary for mixing for such schemes as [Pacanowski and Philander \(1981\)](#), [Chen et al. \(1994\)](#), [Large et al. \(1994\)](#), and [Simmons et al. \(2004\)](#).

- The locally referenced derivatives with respect to potential temperature and salinity

$$\rho_{,\theta} = \left(\frac{\partial \rho}{\partial \theta} \right)_{p,s} \quad (6.1)$$

$$\rho_{,s} = \left(\frac{\partial \rho}{\partial s} \right)_{p,\theta} \quad (6.2)$$

are important for computing both the vertical stratification, and to compute the neutral slopes used for sub-grid-scale tracer transport as in [Griffies et al. \(1998\)](#); [Griffies \(1998\)](#).

- The following combination of second derivatives is used to diagnose the potential for cabbeling to occur in the ocean [McDougall \(1987b\)](#)

$$\begin{aligned} C &= \frac{\partial \alpha}{\partial \theta} + 2 \frac{\alpha}{\beta} \frac{\partial \alpha}{\partial s} - \left(\frac{\alpha}{\beta} \right)^2 \frac{\partial \beta}{\partial s} \\ &= -\rho^{-1} \left[\rho_{,\theta\theta} - 2 \rho_{,\theta s} \left(\frac{\rho_{,\theta}}{\rho_{,s}} \right) + \rho_{,ss} \left(\frac{\rho_{,\theta}}{\rho_{,s}} \right)^2 \right]. \end{aligned} \quad (6.3)$$

- The following combination of second derivatives is used to diagnose the potential for thermobaricity and halobaricity to occur in the ocean [McDougall \(1987b\)](#)

$$\begin{aligned} T &= \beta \partial_p \left(\frac{\alpha}{\beta} \right) \\ &= \frac{\partial \alpha}{\partial p} - \frac{\alpha}{\beta} \frac{\partial \beta}{\partial p} \\ &= -\rho^{-1} \left[\rho_{,\theta p} - \rho_{,ps} \left(\frac{\rho_{,\theta}}{\rho_{,s}} \right) \right]. \end{aligned} \quad (6.4)$$

Note that the name *thermobaricity* is generally used for this parameter, and we evaluate it as given here. However, there are actually contributions from both halobaricity (dependency of haline contraction coefficient on the pressure) and thermobaricity (dependency of thermal expansion coefficient on the pressure). But the thermal piece is generally far larger [McDougall \(1987b\)](#).

In early versions of MOM, density was computed according to the [Bryan and Cox \(1972\)](#) cubic polynomial approximation to the UNESCO equation of state ([Gill, 1982](#)). That approach was quite useful for certain applications. Unfortunately, it has limitations that are no longer acceptable for global climate modeling. First, the polynomials are fit at discrete depth levels. The use of partial cells makes this approach cumbersome since with partial cells it is necessary to generally compute density at arbitrary depths. Second, the cubic approximation is inaccurate for many regimes of ocean climate modeling, such as wide ranges in salinity associated with rivers and sea ice. For these two reasons, a more accurate method is desired.

[Feistel \(1993\)](#), [Feistel and Hagen \(1995\)](#), and [Feistel \(2003\)](#) studied the equilibrium thermodynamics of seawater and produced a more accurate EOS than UNESCO by using more recent empirical data. [McDougall et al. \(2003\)](#) produced a fit to [Feistel and Hagen \(1995\)](#) to render an expression convenient for use in ocean models, and [Jackett et al. \(2006\)](#) updated this equation of state based on [Feistel \(2003\)](#). Finally, [IOC et al. \(2010\)](#) presents a recent update to the equation of state that is the result of a SCOR working group on the thermodynamics of seawater.

The following equations of state (EOS) are currently available in MOM for computing density.

- A linear equation of state whereby density is a linear function of potential temperature and salinity. This EOS is relevant only for idealized simulations.
- The second EOS is that proposed by [Jackett et al. \(2006\)](#).

- The third is that recommended by [IOC et al. \(2010\)](#).

Either the [Jackett et al. \(2006\)](#) or [IOC et al. \(2010\)](#) equation of state are more accurate than the UNESCO EOS due to the use of more accurate empirical data as reported in [Feistel \(1993\)](#), [Feistel and Hagen \(1995\)](#), and [Feistel \(2003\)](#). Such equations of state are now standard in ocean climate modeling.

6.2 Linear equation of state

The default linear equation of state in MOM assumes that density is a linear function of potential temperature and salinity. There is no pressure dependence. Due to the absence of pressure effects, the linear equation of state leads to a density that is more precisely thought of as a potential density. The form used for this equation of state is

$$\rho(\mathbf{x}, t) = \rho_0 - \tilde{\alpha} \theta(\mathbf{x}, t) + \tilde{\beta} s(\mathbf{x}, t). \quad (6.5)$$

The default settings are

$$\tilde{\alpha} = 0.255 (\text{kg/m}^3) ^\circ\text{K}^{-1} \quad (6.6)$$

$$\tilde{\beta} = 0 \quad (6.7)$$

$$\rho_0 = 1035 \text{ kg/m}^3. \quad (6.8)$$

Hence, the density partial derivatives are given by

$$\rho_{,\theta} = -\tilde{\alpha} \quad (6.9)$$

$$\rho_{,s} = \tilde{\beta}. \quad (6.10)$$

The cabbeling and thermobaric parameters vanish for this linear equation of state.

6.3 The two realistic equations of state

The equation of state for [Jackett et al. \(2006\)](#) has 25 terms, and the [IOC et al. \(2010\)](#) equation of state from [McDougall et al. \(2012\)](#) has 48 terms. The form for both equations are motivated by that of [Wright \(1997\)](#) and it takes the following form for the *in situ* density ρ written in terms of pressure, salinity, and potential or conservative temperature

$$\rho(s, \theta, p) = \frac{P_n(s, \theta, p)}{P_d(s, \theta, p)}, \quad (6.11)$$

where p is the gauge pressure in units of decibars, θ is the potential temperature referenced to zero pressure in units of *Celsius*, and s is salinity in *psu*. Note the conversion between mks pressure and decibars is given by

$$10^{-4} \text{ db} = 1 \text{ Pa}. \quad (6.12)$$

The *gauge pressure* is given by

$$p = P_{\text{absolute}} - 10.1325 \text{ dbars} \quad (6.13)$$

where the absolute pressure is the *in situ* pressure measured in the ocean.

For the [Jackett et al. \(2006\)](#) equation of state, salinity is measured in practical salinity units (psu). For the [IOC et al. \(2010\)](#) equation of state available in MOM, salinity is measured in parts per thousand appropriate for the absolute salinity or preformed salinity.

6.3.1 The [Jackett et al. \(2006\)](#) equation of state

The [Jackett et al. \(2006\)](#) equation of state has been fit over the range

$$0 \text{ psu} \leq s \leq 40 \text{ psu} \quad (6.14)$$

$$-3^\circ\text{C} \leq \theta \leq 40^\circ\text{C} \quad (6.15)$$

$$0 \text{ db} \leq p \leq 8000 \text{ db}. \quad (6.16)$$

A check value for the density is $\rho = 1033.213387 \text{ kg.m}^{-3}$ with $s = 35 \text{ psu}$, $\theta = 20^\circ \text{C}$, and $p = 2000 \text{ db} = 2 \times 10^7 \text{ Pa}$. The coefficients a_n and b_n are tabulated in Table A2 [Jackett et al. \(2006\)](#).

The polynomial functions P_n and P_d appearing in the 25-term equation of state (6.11) of [Jackett et al. \(2006\)](#) are given by

$$P_n = a_0 + a_1 \theta + a_2 \theta^2 + a_3 \theta^3 + a_4 s + a_5 s \theta + a_6 s^2 + a_7 p + a_8 p \theta^2 + a_9 p s + a_{10} p^2 + a_{11} p^2 \theta^2 \quad (6.17)$$

$$P_d = b_0 + b_1 \theta + b_2 \theta^2 + b_3 \theta^3 + b_4 \theta^4 + b_5 s + b_6 s \theta + b_7 s \theta^3 + b_8 s^{3/2} + b_9 s^{3/2} \theta^2 + b_{10} p + b_{11} p^2 \theta^3 + b_{12} p^3 \theta. \quad (6.18)$$

Rearrangement in order to reduce multiplications leads to

$$P_n = a_0 + \theta (a_1 + \theta (a_2 + a_3 \theta)) + s (a_4 + a_5 \theta + a_6 s) + p (a_7 + a_8 \theta^2 + a_9 s + p (a_{10} + a_{11} \theta^2)) \quad (6.19)$$

$$P_d = b_0 + \theta (b_1 + \theta (b_2 + \theta (b_3 + \theta b_4))) + s (b_5 + \theta (b_6 + b_7 \theta^2) + s^{1/2} (b_8 + b_9 \theta^2)) + p (b_{10} + p \theta (b_{11} \theta^2 + b_{12} p)). \quad (6.20)$$

The first order partial derivatives of density for the 25 term equation of state are

$$\left(\frac{\partial \rho}{\partial \theta} \right)_{s,p} = \rho \left(\frac{1}{P_n} \left(\frac{\partial P_n}{\partial \theta} \right)_{s,p} - \frac{1}{P_d} \left(\frac{\partial P_d}{\partial \theta} \right)_{s,p} \right) \quad (6.21)$$

$$\left(\frac{\partial \rho}{\partial s} \right)_{\theta,p} = \rho \left(\frac{1}{P_n} \left(\frac{\partial P_n}{\partial s} \right)_{\theta,p} - \frac{1}{P_d} \left(\frac{\partial P_d}{\partial s} \right)_{\theta,p} \right) \quad (6.22)$$

$$\left(\frac{\partial \rho}{\partial p} \right)_{\theta,s} = \rho \left(\frac{1}{P_n} \left(\frac{\partial P_n}{\partial p} \right)_{\theta,s} - \frac{1}{P_d} \left(\frac{\partial P_d}{\partial p} \right)_{\theta,s} \right). \quad (6.23)$$

Since divisions are computationally more expensive than multiplications, we find it useful to rearrange these results to

$$\left(\frac{\partial \rho}{\partial \theta} \right)_{s,p} = (P_d)^{-1} \left[\left(\frac{\partial P_n}{\partial \theta} \right)_{s,p} - \rho \left(\frac{\partial P_d}{\partial \theta} \right)_{s,p} \right] \quad (6.24)$$

$$\left(\frac{\partial \rho}{\partial s} \right)_{\theta,p} = (P_d)^{-1} \left[\left(\frac{\partial P_n}{\partial s} \right)_{\theta,p} - \rho \left(\frac{\partial P_d}{\partial s} \right)_{\theta,p} \right] \quad (6.25)$$

$$\left(\frac{\partial \rho}{\partial p} \right)_{\theta,s} = (P_d)^{-1} \left[\left(\frac{\partial P_n}{\partial p} \right)_{\theta,s} - \rho \left(\frac{\partial P_d}{\partial p} \right)_{\theta,s} \right] \quad (6.26)$$

where $(P_d)^{-1}$ can be saved in a temporary array, thus reducing the number of divisions.¹

The second order density partial derivatives are

$$\left(\frac{\partial^2 \rho}{\partial \theta^2} \right)_{s,p} = (P_d)^{-1} \left[\frac{\partial^2 P_n}{\partial \theta^2} - 2 \frac{\partial \rho}{\partial \theta} \frac{\partial P_d}{\partial \theta} - \rho \frac{\partial^2 P_d}{\partial \theta^2} \right] \quad (6.27)$$

$$\left(\frac{\partial^2 \rho}{\partial s^2} \right)_{\theta,p} = (P_d)^{-1} \left[\frac{\partial^2 P_n}{\partial s^2} - 2 \frac{\partial \rho}{\partial s} \frac{\partial P_d}{\partial s} - \rho \frac{\partial^2 P_d}{\partial s^2} \right] \quad (6.28)$$

$$\left(\frac{\partial^2 \rho}{\partial s \partial \theta} \right) = (P_d)^{-1} \left[\frac{\partial^2 P_n}{\partial s \partial \theta} - \frac{\partial \rho}{\partial s} \frac{\partial P_d}{\partial \theta} - \frac{\partial \rho}{\partial \theta} \frac{\partial P_d}{\partial s} - \rho \frac{\partial^2 P_d}{\partial s \partial \theta} \right] \quad (6.29)$$

$$\left(\frac{\partial^2 \rho}{\partial s \partial p} \right) = (P_d)^{-1} \left[\frac{\partial^2 P_n}{\partial s \partial p} - \frac{\partial \rho}{\partial p} \frac{\partial P_d}{\partial s} - \frac{\partial \rho}{\partial s} \frac{\partial P_d}{\partial p} - \rho \frac{\partial^2 P_d}{\partial s \partial p} \right] \quad (6.30)$$

$$\left(\frac{\partial^2 \rho}{\partial \theta \partial p} \right) = (P_d)^{-1} \left[\frac{\partial^2 P_n}{\partial \theta \partial p} - \frac{\partial \rho}{\partial p} \frac{\partial P_d}{\partial \theta} - \frac{\partial \rho}{\partial \theta} \frac{\partial P_d}{\partial p} - \rho \frac{\partial^2 P_d}{\partial \theta \partial p} \right]. \quad (6.31)$$

¹We thank Trevor McDougall for pointing out this simplification.

The first order partial derivatives of the equation of state functions are given by

$$\left(\frac{\partial P_n}{\partial \theta}\right)_{s,p} = a_1 + 2a_2\theta + 3a_3\theta^2 + a_5s + 2a_8p\theta + 2a_{11}p^2\theta \quad (6.32)$$

$$\left(\frac{\partial P_n}{\partial s}\right)_{\theta,p} = a_4 + a_5\theta + 2a_6s + a_9p \quad (6.33)$$

$$\left(\frac{\partial P_n}{\partial p}\right)_{\theta,s} = a_7 + a_8\theta^2 + a_9s + 2a_{10}p + 2a_{11}p\theta^2 \quad (6.34)$$

$$\begin{aligned} \left(\frac{\partial P_d}{\partial \theta}\right)_{s,p} &= b_1 + 2b_2\theta + 3b_3\theta^2 + 4b_4\theta^3 + b_6s + 3b_7s\theta^2 + 2b_9s^{3/2}\theta \\ &\quad + 3b_{11}p^2\theta^2 + b_{12}p^3 \end{aligned} \quad (6.35)$$

$$\left(\frac{\partial P_d}{\partial s}\right)_{\theta,p} = b_5 + b_6\theta + b_7\theta^3 + (3/2)b_8s^{1/2} + (3/2)b_9s^{1/2}\theta^2 \quad (6.36)$$

$$\left(\frac{\partial P_d}{\partial p}\right)_{\theta,s} = b_{10} + 2b_{11}p\theta^3 + 3b_{12}p^2\theta \quad (6.37)$$

with rearrangement leading to

$$\left(\frac{\partial P_n}{\partial \theta}\right)_{s,p} = a_1 + \theta(2a_2 + 3a_3\theta) + a_5s + 2p\theta(a_8 + a_{11}p) \quad (6.38)$$

$$\left(\frac{\partial P_n}{\partial s}\right)_{\theta,p} = a_4 + a_5\theta + 2a_6s + a_9p \quad (6.39)$$

$$\left(\frac{\partial P_n}{\partial p}\right)_{\theta,s} = a_7 + a_8\theta^2 + a_9s + 2a_{10}p + 2a_{11}p\theta^2 \quad (6.40)$$

$$\begin{aligned} \left(\frac{\partial P_d}{\partial \theta}\right)_{s,p} &= b_1 + \theta(2b_2 + \theta(3b_3 + 4b_4\theta)) + s(b_6 + \theta(3b_7\theta + 2b_9s^{1/2})) \\ &\quad + p^2(3b_{11}\theta^2 + b_{12}p) \end{aligned} \quad (6.41)$$

$$\left(\frac{\partial P_d}{\partial s}\right)_{\theta,p} = b_5 + \theta(b_6 + b_7\theta^2) + (3/2)s^{1/2}(b_8 + b_9\theta^2) \quad (6.42)$$

$$\left(\frac{\partial P_d}{\partial p}\right)_{\theta,s} = b_{10} + 2b_{11}p\theta^3 + 3b_{12}p^2\theta. \quad (6.43)$$

The second order partial derivatives of the equation of state functions are given by

$$\frac{\partial^2 P_n}{\partial \theta^2} = 2a_2 + 6a_3\theta + 2a_8p + 2a_{11}p^2 \quad (6.44)$$

$$\frac{\partial^2 P_n}{\partial s^2} = 2a_6 \quad (6.45)$$

$$\frac{\partial^2 P_n}{\partial s \partial \theta} = a_5 \quad (6.46)$$

$$\frac{\partial^2 P_n}{\partial s \partial p} = a_9 \quad (6.47)$$

$$\frac{\partial^2 P_n}{\partial \theta \partial p} = 2a_8\theta + 4a_{11}p\theta \quad (6.48)$$

$$\frac{\partial^2 P_d}{\partial \theta^2} = 2b_2 + 6b_3\theta + 12b_4\theta^2 + 6b_7s\theta + 2b_9s^{3/2} + 6b_{11}p^2\theta \quad (6.49)$$

$$\frac{\partial^2 P_d}{\partial s^2} = (3/4)b_8s^{-1/2} + (3/4)b_9s^{-1/2}\theta^2 \quad (6.50)$$

$$\frac{\partial^2 P_d}{\partial s \partial \theta} = b_6 + 3b_7\theta^2 + 3b_9s^{1/2}\theta \quad (6.51)$$

$$\frac{\partial^2 P_d}{\partial s \partial p} = 0 \quad (6.52)$$

$$\frac{\partial^2 P_d}{\partial \theta \partial p} = 6b_{11}p\theta^2 + 3b_{12}p^2. \quad (6.53)$$

6.3.2 TEOS-10 equation of state

Documentation of the TEOS-10 equation of state relies on the work of [IOC et al. \(2010\)](#) and the paper by [McDougall et al. \(2012\)](#) in preparation. The form of the equation of state is analogous to the 25 term form of Section 6.3.1, with new polynomial terms needed to better account for a wider range of temperature and salinity.

The TEOS-10 equation of state uses the following prognostic temperature and salinity fields:

- CONSERVATIVE TEMPERATURE: The conservative temperature variable of [McDougall \(2003\)](#).
- PREFORMED SALINITY: The preformed salinity variable detailed in [IOC et al. \(2010\)](#).

Although the salinity variable time stepped by the model is preformed salinity, there is a translation made to absolute salinity before computing the density, since the equation of state is a function of absolute salinity, not preformed salinity.

Caveat

Testing of the TEOS-10 equation of state, and in particular the use of preformed salinity, remains incomplete. Additionally, some of the model diagnostics remain incomplete, such as the diapycnal transport diagnostics detailed in Chapter 36.

DYNAMICAL OCEAN EQUATIONS WITH A NONCONSTANT GRAVITY FIELD

Contents

7.1	Gravitational force: conventional approach	103
7.2	Gravitational force: general approach	105
7.2.1	Momentum equation	105
7.2.2	Primitive equations	107
7.2.3	Depth independent perturbed geopotential	107

The purpose of this chapter is to formulate the dynamical equations of the ocean in the presence of a space and time dependent gravitational acceleration. This formulation has applications for the implementation of astronomical tide forcing (Chapter 8). In addition, inhomogeneities in mass distributions cause the earth's gravity field to be non-spherical. Of increasing interest to climate science is the study of how the ocean responds to changes in mass distributions associated with melting land ice. Given the nontrivial impact that melting land glaciers has on the earth's geoid (Farrell and Clark (1976) and Mitrovica et al. (2001)), we formulate the dynamical equations of a liquid ocean in the presence of a space-time dependent gravity field.

The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_barotropic.F90

7.1 Gravitational force: conventional approach

The effective gravitational force is noncentral due to the Earth's rotation. Hence if the Earth were a homogeneous ideal fluid, matter would flow from the poles toward the equator, thus ensuring that the Earth's surface would everywhere be perpendicular to the effective gravitational acceleration, \mathbf{g} . Indeed, the Earth does exhibit a slight equatorial bulge. However, inhomogeneities in the Earth's composition and surface loading by continents, glaciers, and seawater make its shape differ from the ideal case.

Veronis (1973), Phillips (1973), and Gill (1982) discuss how the Earth's geometry can be well approximated by an oblate spheroid, with the equatorial radius larger than the polar due to centrifugal effects. With this geometry, surfaces of constant geopotential are represented by surfaces with a constant oblate spheroid radial coordinate (page 662 of Morse and Feshbach, 1953). However, the oblate spheroidal metric functions, which determine how to measure distances between points on the spheroid, are less convenient to use for ocean modelling than the more familiar spherical metric functions.

To provide a simpler form of the equations of motion on the Earth, Veronis (1973) and Gill (1982) (see in particular page 91 of Gill) indicate that it is possible, within a high level of accuracy, to maintain the best of both situations. That is, surfaces of constant radius r are interpreted as best fit oblate spheroidal geopotentials, yet the metric functions used to measure distance between points in the surface are approximated as spherical. As the metric functions determine the geometry of the surface, and hence the form of the equations of motion, the equations are exactly those that result when using spherical coordinates on a sphere. Hence, in global ocean climate modelling, one generally considers the geometry of the Earth to be spherical as in Figure 7.1, yet the radial position r represents a surface of constant geopotential, which is approximated by an oblate spheroid.

In summary, the gravitational field traditionally used for ocean climate modelling is an effective gravitational field, which incorporates the effects from the centrifugal force. The effective gravitational field is conservative, so that the gravitational acceleration of a fluid parcel can be represented as the gradient of a scalar,

$$\mathbf{g} = -\nabla\Phi, \quad (7.1)$$

with Φ known as the geopotential. With the mass of a fluid parcel written as ρdV , then

$$\mathcal{P} = (\rho dV)\Phi \quad (7.2)$$

is the gravitational potential energy of a parcel, thus making Φ the gravitational potential energy per mass of a fluid parcel.

In this equation, ρ is the *in situ* density and dV its volume. In most ocean modelling applications, the local vertical direction is denoted by

$$z = r - R, \quad (7.3)$$

with $z = 0$ the geopotential surface corresponding to a resting ocean. The geopotential in this case is given by case

$$\Phi \approx \Phi_0 = gz, \quad (7.4)$$

with $g \approx 9.8 \text{ ms}^{-2}$ the typical value taken in ocean climate models for the acceleration due to gravity at the earth's surface.

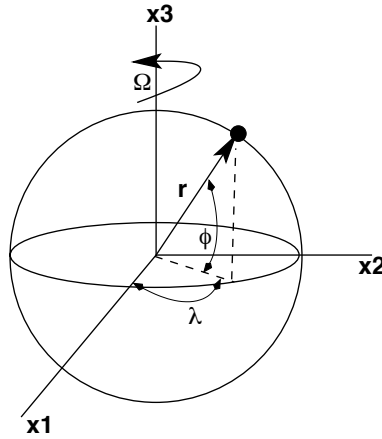


Figure 7.1: The position vector for a point in 3D Euclidean space can be represented in terms of many sets of coordinates, such as the Cartesian coordinates (x^1, x^2, x^3) or the spherical coordinates (λ, ϕ, r) . In a geophysical context, the angular coordinate $0 \leq \lambda \leq 2\pi$ is the longitude, with positive values measured eastward from a meridian passing through Greenwich, England. The angular coordinate ϕ is the latitude, with values $\phi = 0$ at the equator and $\phi = \pi/2(-\pi/2)$ at the north (south) poles. The radial distance r is measured here with respect to the center of the sphere. The coordinate transformation between Cartesian and spherical is given by $(x^1, x^2, x^3) = r(\cos \phi \cos \lambda, \cos \phi \sin \lambda, \sin \phi)$.

7.2 Gravitational force: general approach

Absent changes to gravity and crustal rebound, the melting of Greenland would add about 7 m of water to the ocean's sea level. As shown by [Farrell and Clark \(1976\)](#) and [Mitrovica et al. \(2001\)](#), the melting of land ice over Greenland, Antarctica, or mountain glaciers, creates a sizable perturbation to the present-day *static equilibrium* sea level, which we refer to as the *geoid* in the following. For example, according to [Farrell and Clark \(1976\)](#) and [Mitrovica et al. \(2001\)](#), if all of Greenland melted, much of the northern North Atlantic and Arctic Ocean would see only a fraction of the 7 m rise, with some regions close to Greenland actually seeing a reduction in sea level (see Figure 1A in [Mitrovica et al., 2001](#)), whereas other ocean regions, such as the central and South Pacific, Indian, and South Atlantic, would see more than 7m sea level rise.

It is generally assumed by climate modellers that changes in the geoid can be used *post facto* to renormalize projections of sea level change from global climate models simulated with a fixed geoid. The results from [Farrell and Clark \(1976\)](#) and [Mitrovica et al. \(2001\)](#) prompt us to question this assumption, especially for dramatic changes associated with melting Greenland or Antarctica. Furthermore, changes in the geoid associated with past glacial periods, such as ice ages, would likewise be a nontrivial modification to sea level.

The study of [Kopp et al. \(2010\)](#) represents the first attempt to partially remove the constant geoid assumption for purposes of global ocean climate. In that study, the prognostic sea level from an ocean climate model was combined with an evolving equilibrium sea level determined as a function of changing mass of seawater and land ice. The [Kopp et al. \(2010\)](#) study only partially addresses the main question associated with this issues: namely, will the drastic and rapid changes in the geoid associated with land ice melting have a nontrivial impact on ocean circulation? To address that question requires ocean climate models to be run with an evolving gravity field. It is to this issue that we now turn.

Consider a generalized geopotential written in the form

$$\Phi = \Phi_0(r) + \Phi_1(r, \lambda, \phi, t), \quad (7.5)$$

where $\Phi_0(r)$ is the unperturbed geopotential given by equation (7.4), and Φ_1 incorporates perturbations to the geopotential associated with changes in land ice cover. Within the ocean fluid, the radial dependence of Φ_1 is generally quite weak, though it can be large for regions near the melting land ice. We thus maintain this dependence for purposes of generality, though it will be dropped for certain specialized examples. The calculation of ocean tides arising from astronomical forcing is formulated with a space-time dependent geopotential as in equation (7.5), with the radial dependence of Φ_1 neglected (e.g., Section 9.8 in [Gill, 1982](#)). [Arbic et al. \(2004\)](#) provide a recent discussion of global tide modelling.

7.2.1 Momentum equation

The linear momentum of a fluid parcel is given by

$$\mathbf{P} = \mathbf{v} \rho dV, \quad (7.6)$$

where again ρdV is the mass of the parcel. Through Newton's Second Law of Motion, momentum changes in time due to the influence of forces acting on the parcel. As recently reviewed by [Griffies and Adcroft \(2008\)](#), the equation for linear momentum of a fluid parcel takes the form

$$\rho \frac{d\mathbf{v}}{dt} + 2\boldsymbol{\Omega} \wedge \rho \mathbf{v} = -(\nabla p + \rho \nabla \Phi) + \nabla \cdot \boldsymbol{\tau}. \quad (7.7)$$

The left-hand side of this equation is the material time change for the linear momentum per volume of a parcel, along with the Coriolis force, and the right-hand side is the sum of the pressure, gravitational, and frictional forces.

In writing the momentum equation in the form of (7.7), we have chosen to retain an orientation afforded by the unperturbed geopotential surfaces, which correspond to surfaces of constant depth z . This approach reflects that commonly used to study ocean tides. In the presence of a perturbed geopotential Φ_1 , the "horizontal" directions defined by surfaces of constant z are no longer parallel to geopotential surfaces.

We thus may interpret the sum $\nabla_z p + \rho \nabla_z \Phi$ as an orientation of the pressure gradient along surfaces of constant geopotential, where the geopotential is determined by $\Phi = \Phi_0 + \Phi_1$, rather than just the unperturbed geopotential Φ_0 . This result is familiar to those who have formulated ocean models in generalized vertical coordinates (see, for example, Chapter 6 of [Griffies, 2004](#)).

We next write the momentum equation in component form using spherical coordinates. For this purpose, introduce the orthogonal unit vectors $\hat{\lambda}$, $\hat{\phi}$, and \hat{r} , each moving with the rotating sphere (Figure 7.1). The vector $\hat{\lambda}$ points in the positive longitude direction, $\hat{\phi}$ points in the positive latitude direction, and \hat{r} points radially outward from the center of the sphere. In spherical coordinates, the angular rotation velocity for the sphere takes the form

$$\boldsymbol{\Omega} = \Omega (\hat{r} \sin \phi + \hat{\phi} \cos \phi), \quad (7.8)$$

and the velocity of a fluid parcel is written

$$\begin{aligned} \mathbf{v} &= \hat{\lambda} (r \cos \phi) \frac{d\lambda}{dt} + r \hat{\phi} \frac{d\phi}{dt} + \hat{r} \frac{dr}{dt} \\ &= u \hat{\lambda} + v \hat{\phi} + w \hat{z}, \end{aligned} \quad (7.9)$$

where $\hat{z} = \hat{r}$ is the radial/vertical unit vector, and

$$(u, v, w) = (r \cos \phi \, d\lambda/dt, r \, d\phi/dt, dr/dt) \quad (7.10)$$

are spherical components to the velocity vector. Hence, the Coriolis force per mass is given by

$$\begin{aligned} 2\boldsymbol{\Omega} \wedge \mathbf{v} &= 2\Omega (w \cos \phi - v \sin \phi) \hat{\lambda} \\ &\quad + 2\Omega u \sin \phi \hat{\phi} - 2\Omega u \cos \phi \hat{r}. \end{aligned} \quad (7.11)$$

Introducing the notation

$$f = 2\Omega \sin \phi \quad (7.12)$$

$$f^* = 2\Omega \cos \phi, \quad (7.13)$$

leads to the equation of motion

$$\left(\frac{d}{dt} + (\hat{z} f + \hat{\lambda} f^*) \wedge \right) \mathbf{v} = -(\nabla \Phi + \rho^{-1} \nabla p) + \mathbf{F} \quad (7.14)$$

which takes the component form

$$\frac{du}{dt} - v f + w f^* = -(\Phi_{,x} + \rho^{-1} p_{,x}) + F^{(x)} \quad (7.15)$$

$$\frac{dv}{dt} + u f = -(\Phi_{,y} + \rho^{-1} p_{,y}) + F^{(y)} \quad (7.16)$$

$$\frac{dw}{dt} - u f^* = -(\Phi_{,z} + \rho^{-1} p_{,z}) + F^{(z)}. \quad (7.17)$$

In these equations, we defined

$$\mathbf{F} = \frac{1}{\rho} \nabla \cdot \boldsymbol{\tau} \quad (7.18)$$

as the friction vector per unit mass, wrote

$$\frac{\partial}{\partial x} = \frac{1}{r \cos \phi} \frac{\partial}{\partial \lambda} \quad (7.19)$$

$$\frac{\partial}{\partial y} = \frac{1}{r} \frac{\partial}{\partial \phi} \quad (7.20)$$

$$\frac{\partial}{\partial z} = \frac{\partial}{\partial r} \quad (7.21)$$

as the three-dimensional gradient operator in spherical coordinates, and introduced a comma as a shorthand for partial derivative.

7.2.2 Primitive equations

Large-scale ocean general circulation models are typically based on the hydrostatic primitive equations. Here, the vertical momentum equation is reduced to its static inviscid form with $f^* = 0$

$$\begin{aligned}\partial_z p &= -\rho \partial_z \Phi \\ &= -\rho(g + \partial_z \Phi_1).\end{aligned}\tag{7.22}$$

This *hydrostatic balance* filters motions associated with strong vertical accelerations, such as may occur in regions of gravitational instability. It also filters out the majority of acoustic modes, with only the Lamb Wave remaining (see, for example, [Griffies and Adcroft, 2008](#), for further discussion). The hydrostatic balance is modified from its traditional form for cases where the perturbation geopotential Φ_1 exhibits nontrivial depth dependence. This extra term represents a potential significant modification to the usual primitive equations of an ocean model.

Vertical integration of the hydrostatic balance (7.22) from the ocean surface to an arbitrary depth z leads to the hydrostatic pressure

$$p(x, y, z, t) = p_a + g \int_z^\eta \rho dz + \int_z^\eta \rho \partial_z \Phi_1 dz,\tag{7.23}$$

where p_a is the pressure applied to the ocean surface from the media above the ocean (e.g., the overlying atmosphere or ice).

Consistent with the hydrostatic balance, we drop the $w f^*$ term appearing in the zonal momentum equations, thus reducing the momentum equation (7.14) to the primitive equation set

$$\left(\frac{d}{dt} + \hat{\mathbf{z}} f \wedge \right) \mathbf{u} = -(\nabla_z \Phi_1 + \rho^{-1} \nabla_z p) + \mathbf{F}\tag{7.24}$$

$$\partial_z p = -\rho(g + \partial_z \Phi_1),\tag{7.25}$$

where $\mathbf{u} = (u, v)$ is the horizontal velocity vector, and $\nabla_z = (\partial_x, \partial_y, 0)$ is the horizontal gradient taken on surfaces of constant z . For a volume conserving Boussinesq fluid, such as used for CM2.1 and ESM2M we make one final assumption for the pressure gradient, whereby the momentum equations become

$$\left(\frac{d}{dt} + \hat{\mathbf{z}} f \wedge \right) \mathbf{u} = -\rho_o^{-1} \nabla_z (\rho_o \Phi_1 + p) + \mathbf{F}\tag{7.26}$$

$$\partial_z p = -\rho(g + \partial_z \Phi_1),\tag{7.27}$$

where ρ_o is the constant reference density for a Boussinesq fluid. The Boussinesq form makes the addition of a perturbed geopotential quite straightforward, in which it is gradients in $\rho_o \Phi_1 + p$ that take the place of gradients in pressure p .

7.2.3 Depth independent perturbed geopotential

The simplest case to consider is a depth independent perturbed geopotential

$$\Phi_1 = \Phi_1(x, y, t).\tag{7.28}$$

As stated earlier, this is precisely the form assumed for studies of ocean tides. In this form, we are motivated to write the full geopotential as

$$\Phi = g(z - h)\tag{7.29}$$

where

$$\Phi_1 = -g h\tag{7.30}$$

introduces a perturbed geopotential height field $h = h(x, y, t)$. Rather than $z = 0$, the zero of the geopotential is now set by $z = h$. The impact of the perturbed geopotential is isolated to the depth integrated momentum

equations, which for a Boussinesq fluid¹ takes the form (see Section 10.9)

$$\rho_o (\partial_t + f \hat{\mathbf{z}} \wedge) \mathbf{U} = \mathbf{G} - (H + \eta) \nabla (p_a + p_{\text{surf}} + \rho_o \Phi_1), \quad (7.31)$$

where

$$\mathbf{U} = \int_{-H}^{\eta} \mathbf{u} \, dz \quad (7.32)$$

is the vertically integrated horizontal velocity, \mathbf{G} is the vertical integral of the depth dependent terms on the right hand side of the momentum equation (7.24). \mathbf{G} embodies all contributions which are generally evolving on a slower baroclinic time scale. The surface pressure is given by

$$p_{\text{surf}} = g \rho_{\text{surf}} \eta, \quad (7.33)$$

and it represents the hydrostatic pressure at $z = 0$ associated with water in the region between $z = 0$ and $z = \eta$. The applied pressure p_a is the pressure applied at the top of the ocean arising from media above the ocean, such as the atmosphere and sea ice. To within a good approximation, we can combine the surface pressure and geopotential terms to bring the momentum equation to the form

$$\rho_o (\partial_t + f \hat{\mathbf{z}} \wedge) \mathbf{U} = \mathbf{G} - (H + \eta) \nabla [p_a + g \rho_{\text{surf}} (\eta - h)]. \quad (7.34)$$

In this way, modifications to the geopotential, embodied by the perturbed geopotential height field $h = h(x, y, t)$, are isolated to their impacts on the horizontal pressure gradients acting on the vertically integrated momentum field. As stated earlier, this formulation is identical to that associated with the study of ocean tides, where in the case of tides, fluctuations in h arise from astronomical perturbations to the earth's gravity field. For our present considerations, h arises from perturbations in terrestrial masses, such as the melting land ice on Greenland or Antarctica. In contrast to ocean tides, geoid perturbations associated with melting land ice are not periodic. Furthermore, as evidenced by Figure 1 in [Mitrovica et al. \(2001\)](#), the amplitude of geoid perturbations can be far greater than typical open ocean tide fluctuations.

Changes in the geoid associated with nontrivial $h = h(x, y, t)$ will propagate throughout the vertically integrated momentum field on a rapid barotropic time scale. Consequently, the ocean's free surface will adjust within a few days of geoid perturbations, just as it does for ocean tides. In contrast, it is unclear how the depth dependent ocean circulation will adjust, with a general circulation model a useful tool for considering the slower baroclinic adjustment processes.

We note that when coupling to a sea ice model, it is the effective sea level given by

$$\eta_{\text{eff}} = \eta + \left(\frac{p_a}{g \rho_{\text{surf}}} \right) - h \quad (7.35)$$

that is to be passed from the ocean model to the sea ice model for the purpose of computing horizontal pressure gradients acting on the ice.

¹See Section 8.2 for the non-Boussinesq mass conserving form.

TIDAL FORCING FROM THE MOON AND SUN

Contents

8.1	Tidal constituents and tidal forcing	109
8.2	Formulation in non-Boussinesq models	110
8.3	Implementation in MOM	110

The purpose of this chapter is to describe the formulation of lunar and solar tidal forcing implemented in MOM. This chapter was written by Harper Simmons (hsimmons@iarc.uaf.edu) with some additions and edits by Stephen.Griffies@noaa.gov.

The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_barotropic.F90

8.1 Tidal constituents and tidal forcing

As formulated in Chapter 7 (see also [Marchuk and Kagan \(1989\)](#)), tidal forcing appears in the momentum equations as a depth independent acceleration. Consequently, tide dynamics can be isolated in the vertically integrated momentum budget. As shown in Section 10.9.3, the equation for the vertically integrated transport \mathbf{U} in a Boussinesq version of MOM takes the form (equation (10.137))

$$\rho_o (\partial_t + f \hat{\mathbf{z}} \wedge) \sum (dz \mathbf{u}) = \mathbf{G} - (H + \eta) \nabla (p_a + p_{\text{surf}}). \quad (8.1)$$

In this equation, \mathbf{G} is the vertically integrated forcing arising from baroclinic effects, p_s is the pressure associated with undulations of the surface height, p_a is the applied pressure from the atmosphere and sea ice, H is the depth of the ocean, and η is the surface height deviation from a resting state with $z = 0$. Our goal is to modify this equation to account for gravitational forcing that give rise to ocean tides.

Tidal forcing arising from the eight primary constituents (M2, S2, N2, K2, K1, O1, P1, Q1) (see [Gill \(1982\)](#)) can be added to the forcing for \mathbf{U} in MOM. The formulation follows [Marchuk and Kagan \(1989\)](#), by considering a tide generating potential ($g\eta_{\text{eq}}$) with corrections due to both the earth tide ($1 + k - h$) and self-attraction and loading (α). In this approach, the depth independent pressure gradient acceleration is modified to the form

$$\rho_o^{-1} \nabla (p_s + p_a) \rightarrow \rho_o^{-1} \nabla (p_s + p_a) + g \nabla [\alpha \eta - (1 + k - h) \eta_{\text{eq}}]. \quad (8.2)$$

The term η_{eq} is known as the *equilibrium tide*, and it arises from the astronomically derived gravity producing forces. It is modified by several factors. The Love numbers, k and h , named for the physicist A.L. Love, account for the reduction of the ocean tide because of the deformation of the solid earth by tidal forces. The Love numbers are frequency dependent, with $1 + k - h$ generally close to 0.7 ([Wahr \(1998\)](#)).

The term α in equation (8.2) accounts for a modification of the ocean's tidal response as a result of self-attraction and loading (SAL) (Hendershott (1972)). Self attraction is the modification of the tidal potential as a result of the redistribution of the earth and ocean due to the equilibrium tidal forcing. Loading refers to the depression of the earth's crust by the mounding of tides. Calculation of the SAL term requires an extremely cumbersome integration over the earth surface, rendering equation (8.2) an integro-differential equation (Ray (1998)).

Instead of solving the integro-differential form of equation (8.2), MOM4 uses the scalar approximation to SAL. We feel this is justified since our purpose in introducing tidal forcing is to study the *effects* of tides on the general circulation, not the details of the tides themselves. The conjecture is that precise calculation of the SAL term is not needed for to understand tidal effects on the general circulation. For the scalar approximation, α is usually set between 0.940 – 0.953. MOM4 uses $\alpha = 0.948$. Limitations of the scalar approximation to SAL are discussed by Ray (1998), who concluded that the scalar approximation introduces phase errors of up to 30° and amplitude errors of 10% into a global scale tidal simulation.

8.2 Formulation in non-Boussinesq models

The horizontal acceleration from pressure gradients is given by the two terms (see Section 10.9.4, where we drop here the tilde notation used in that section)

$$\rho^{-1} (\nabla_z p)_{\text{without tidal forcing}} = \rho^{-1} \nabla_s p + \nabla_s \Phi. \quad (8.3)$$

In this equation, p is the hydrostatic pressure at a grid point, Φ is the geopotential at this point, and s is the generalized vertical coordinate. The ρ^{-1} factor is set to ρ_o^{-1} for Boussinesq models, but remains nontrivial for non-Boussinesq, pressure-based vertical coordinates in MOM. As noted in Section 8.1, gravitational forces giving rise to ocean tides can be incorporated into MOM by adding a depth independent acceleration throughout the water column. Following the approach used for the Boussinesq case, we add to the non-Boussinesq pressure gradient a modification to the geopotential due to tidal acceleration

$$\rho^{-1} (\nabla_z p)_{\text{with tidal forcing}} = \rho^{-1} \nabla_s p + \nabla_s \Phi + g \nabla [\alpha \eta - (1 + k - h) \eta_{\text{eq}}], \quad (8.4)$$

where the tidal term is taken from equation (8.2). Inserting this modified acceleration into the vertically integrated momentum equation (10.145) yields

$$(\partial_t + f \hat{\mathbf{z}} \wedge) \sum (dz \rho \mathbf{u}) = \mathbf{G} - \frac{p_b - p_a}{g \rho_o} \nabla (p_b + \rho_o \Phi_b + g \rho_o h_{\text{tide}}), \quad (8.5)$$

where

$$h_{\text{tide}} = \alpha \eta - (1 + k - h) \eta_{\text{eq}} \quad (8.6)$$

is shorthand for the tidal term, p_b is the pressure at the ocean bottom, and $\Phi_b = -gH$ is the geopotential at the bottom.

8.3 Implementation in MOM

The equilibrium tide is written for the n^{th} diurnal tidal constituent as

$$\eta_{\text{eq},n} = H_n \sin 2\phi \cos(\omega_n t + \lambda), \quad (8.7)$$

and for the n^{th} semi-diurnal constituent as

$$\eta_{\text{eq},n} = H_n \cos^2 \phi \cos(\omega_n t + 2\lambda), \quad (8.8)$$

where ϕ is latitude and λ is longitude. Recognizing that equations (8.7) and (8.8) require the evaluation of trigonometric functions at every grid point and every time-step, tidal forcing is introduced into MOM4 in the following mathematically equivalent form. Making use of the identity

$$\cos(A + B) = \cos(A) \cos(B) - \sin(A) \sin(B), \quad (8.9)$$

constit	name	origin	ω (1/day)	$1 + k - h$	a (m)
1	K_1	Luni-solar declinational	0.7292117	0.736	0.141565
2	O_1	Principal lunar declinational	0.6759774	0.695	0.100661
3	P_1	Principal solar declinational	0.7252295	0.706	0.046848
4	Q_1	Larger lunar elliptic	0.6495854	0.695	0.019273
5	M_2	Principal lunar	1.405189	0.693	0.242334
6	S_2	Principal solar	1.454441	0.693	0.112743
7	N_2	Larger lunar elliptic	1.378797	0.693	0.046397
8	K_2	Luni-solar declinational	1.458423	0.693	0.030684

Table 8.1: Frequencies, Love numbers, and amplitude functions for the eight principle constituents of tidal forcing available in MOM4.

we can write the eight tidal forcing constituents as

$$\eta_{\text{eq}} = \sum_{n=1}^4 \left[\beta_n a_n \cos^2 \phi [\cos(\omega_n t) \cos 2\lambda - \sin(\omega_n t) \sin 2\lambda] + \beta_{n+4} a_{n+4} \sin 2\phi [\cos(\omega_{n+4} t) \cos 2\lambda - \sin(\omega_{n+4} t) \sin 2\lambda] \right], \quad (8.10)$$

which allows all the trigonometric functions of ϕ and λ to be precomputed. Note that we have written $\beta_n = 1 + k_n - h_n$. The frequencies (ω_n) , amplitudes (a_n) and Love numbers are listed in Table 8.1.

Numerical formulations

The purpose of this part of the document is to describe algorithms used to numerically solve the ocean primitive equations in MOM. We address discretization issues for both space and time stepping the continuum equations.

B AND C GRID DISCRETIZATIONS

Contents

9.1	B and C grids used in MOM	115
9.1.1	Variables on the B-grid	116
9.1.2	Variables on the C-grid	116
9.2	Describing the horizontal grid	117
9.2.1	Four basic grid points and corresponding cells	118
9.2.2	Horizontal layout of wet and dry cells	118
9.2.3	Computing the grid distances	119
9.2.4	Grid distances carried by the model	121
9.3	The Murray (1996) tripolar grid	121
9.4	Specifying fields and grid distances within halos	123
9.4.1	Interior domains	123
9.4.2	Exterior domains	126
9.4.2.1	Solid wall boundary conditions	126
9.4.2.2	Periodic boundary conditions	126
9.4.3	The bipolar Arctic grid	128
9.4.3.1	Fields defined at points T,U,N, and E	129
9.4.3.2	Grid distances for horizontal quarter-cells	129
9.4.3.3	Grid distances for horizontal full cells	132
9.4.3.4	Summary of redundancies and halo mappings	132

The purpose of this chapter is to detail the horizontal grids used in MOM as well as the specification of field and grid values in halo regions. Details of the vertical discretization are presented in Section 10.1 (see in particular Figure 10.1). This chapter builds from Chapter 4 in [Griffies et al. \(2004\)](#), with newer material here concerned with the C-grid option now under development in MOM. Further information about the MOM grids and topography can be found in Chapters 16 and 18 of the MOM3 Manual ([Pacanowski and Griffies, 1999](#)).

The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_grid.F90

9.1 B and C grids used in MOM

The continuum partial differential equations of MOM are derived and discussed in Part I, as well as in [Griffies \(2004\)](#). [Bryan \(1969\)](#) cast the discrete version of these equations on an Arakawa B-grid. As sum-

marized in the review article by [Griffies et al. \(2000a\)](#), the B-grid allows for a reasonably accurate representation of geostrophic currents, even when running a coarse grid model. However, many recent applications with other model codes such as GOLD, HYCOM, MITgcm, NEMO, and ROMs exploit the advantages of a C-grid, with the following two advantages notable.

- At resolutions where the baroclinic radius is well resolved, the C-grid presents certain advantages for rotating stratified flow over the B-grid (Section 3.2 of [Griffies et al. \(2000a\)](#)).¹
- For coastal applications, details of the fractal land/sea boundary are critical. Use of the B-grid leads to complexities associated with the need to distinguish between tracer and velocity cells. That is, to have advective flow through a tracer cell requires two adjacent velocity cells. The net effect is that narrow straights may need to be unphysically widened to facilitate advective transport. This situation hinders the utility of the B-grid for representing complex land/sea regions, with particular importance placed on such details for coastal modeling.

It is for these reasons that MOM, which has traditionally used exclusively a B-grid, will soon have a C-grid option. The C-grid option is presently not available for general use, but development during late 2012 will focus nearly exclusively on this layout, given the focus at GFDL on mesoscale eddy permitting and resolving simulations.

9.1.1 Variables on the B-grid

Figure 9.1 illustrates the horizontal arrangement of prognostic model fields used with the B-grid. The B-grid places both horizontal prognostic velocity components at the same point, the corner of the tracer cell. This placement is natural when computing the Coriolis Force. However, it is unnatural for computation of advective tracer transport or the horizontal pressure gradient force acting on velocity. The need to perform an averaging operation when computing the horizontal pressure gradient leads to the computational mode associated with gravity waves on the B-grid (see Section 31.1 and references cited there).

MOM follows a *northeast convention*, whereby the velocity is positioned at the northeast corner of the corresponding tracer cell. With half-integer notation, the velocity U-point lives at $(i+1/2, j+1/2)$ with the T-point at (i, j) . There are good reasons to employ the half-integer convention when representing discrete quantities on a grid. However, we choose to avoid such notation, preferring instead to keep the grid variable placements implied by use of the *northeast convention*.

The B-grid placement leads to the following placements for the discrete fields realized in MOM on the grid.

- As density is a function of temperature, salinity, and pressure, density is naturally defined at the tracer point. Correspondingly, so is hydrostatic pressure and the surface ocean height.
- For each tracer cell there is a corresponding velocity cell, as depicted in Figure 9.1. Fluxes through the faces of the velocity cell are related to those through the faces of the tracer cell via remapping operations as detailed in Chapter 15.
- The vertical velocity component is defined according to the requirements of continuity across the tracer and velocity cells. Hence, the vertical velocity component lives at the bottom face of the corresponding tracer or velocity cell. Once the horizontal grid placement is defined, the vertical position is specified for both the grid point and the vertical velocity position. Chapter 16 of **The MOM3 Manual** provides further details of the vertical grid.

9.1.2 Variables on the C-grid

Figure 9.2 illustrates the horizontal arrangement of prognostic model fields used with the C-grid. The C-grid places the zonal velocity component on the zonal tracer cell face, and meridional velocity component

¹As pointed out by [Webb et al. \(1998\)](#), there will potentially always be important unresolved baroclinic modes, such as in the equatorial region. Hence, it will be very useful to have both B and C grid options in MOM to better examine the pros and cons for any particular application.

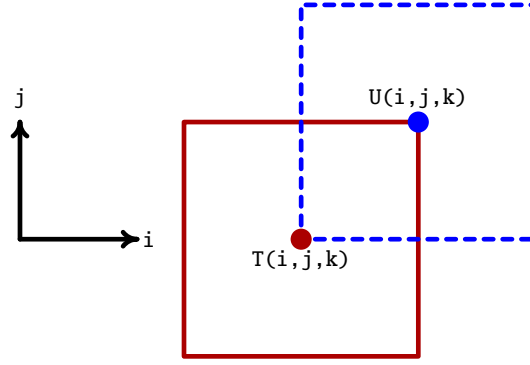


Figure 9.1: Illustration of how fields are placed on the horizontal B-grid used in MOM using a *northeast convention*. Velocity points $U(i, j, k)$ are placed to the northeast of tracer points $T(i, j, k)$. Both horizontal velocity components $u_{i,j,k}$ and $v_{i,j,k}$ are placed at the velocity point $U(i, j, k)$. Both the tracer point and velocity point have a corresponding grid cell region, denoted by the solid and dashed squares.

on the meridional tracer cell face. This placement is suited for computation of advective tracer transport. It is also suited for computing the stress tensor and the horizontal pressure gradient force acting on velocity components. However, it is not natural for computation of the Coriolis Force. The need to perform an averaging operation to compute the Coriolis Force leads to the presence of a computational null mode associated with geostrophically balanced flow (Adcroft et al., 1999).

Following a *northeast convention*, MOM places its zonal velocity component $u_{i,j,k}$ on the east face of the tracer cell $T(i, j)$, and the meridional velocity component $v_{i,j,k}$ at the north face of the same tracer cell. With half-integer notation, the zonal velocity component $u_{i,j,k}$ lives at the $(i+1/2, j)$ point whereas the meridional velocity component $v_{i,j,k}$ lives at the $(i, j+1/2)$ point. This C-grid convention leads to the following placements for the discrete fields realized in MOM.

- As density is a function of temperature, salinity, and pressure, density is naturally defined at the tracer point. Correspondingly, so is hydrostatic pressure and the surface ocean height. Indeed, all tracer quantities from the B-grid are correspondingly on the same tracer cell using the C-grid. This agreement means that nearly all processes associated with tracer transport have direct correspondence across the B and C grids, without any changes required for the code.
- The vertical velocity component is defined according to the requirements of continuity across the tracer cell. Hence, the vertical velocity component lives at the bottom face of the tracer cell. Additionally, it is necessary to prescribe a means to compute the vertical velocity used to advect zonal and meridional velocity. This velocity component is prescribed in terms of averages of the corresponding tracer grid vertical velocity component.
- The B-grid velocity point, which sits at the corner of the tracer cell, is the natural position for the vertical component of vorticity

$$\zeta = \hat{\mathbf{z}} \cdot (\nabla \wedge \mathbf{v}) = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}. \quad (9.1)$$

When writing the velocity equation in a vector-invariant form (see Section 4.4.4 of Griffies (2004)), as in GOLD, MITgcm, and NEMO, the vorticity point is also the natural location for defining the Coriolis parameter, f , for use in computing the total vorticity $\zeta + f$.

9.2 Describing the horizontal grid

With the use of generalized horizontal coordinates in MOM, there are many grid distances required to compute discrete derivatives, integrals, and areas. When constructing the grid distances in MOM, we

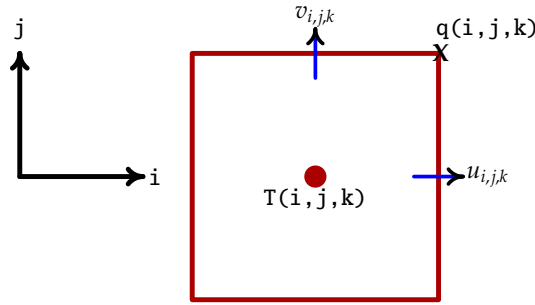


Figure 9.2: Illustration of how fields are placed on the horizontal C-grid used in MOM. As for the B-grid, MOM’s convention is that the zonal velocity component $u_{i,j,k}$ sits at the east face of the tracer cell $T(i, j)$, and the meridional velocity component $v_{i,j,k}$ sits at the north face of the tracer cell $T(i, j, k)$. This convention follows the *northeast* convention also used for the B-grid. However, note that the notion of a corresponding “velocity cell” that surrounds each velocity component is less tenable for the C-grid.

aimed to design a structure useful for both B and C-grids. It is with this goal in mind that the names for the grid distances in the *grid_generator* module are distinct from grid distances used in MOM’s grids module. We note the mapping between the two grid conventions in the following.

9.2.1 Four basic grid points and corresponding cells

On both the B and C grids, it is useful to consider the tracer cell as the basic cell, and all other cells in their relation to the tracer cell. Given this convention, there are four basic grid points and corresponding grid cells that can be identified: $T(i, j)$, $E(i, j)$, $C(i, j)$, and $N(i, j)$. Figure 9.3 illustrates these points as oriented according to the tracer cell.

- $T(i, j)$ is the usual tracer point that is surrounded by a tracer cell region.
- $C(i, j)$ sits at the northeast corner of the tracer cell, and so is equivalent to the B-grid velocity point $U(i, j)$ and the C-grid vorticity point $q(i, j)$.
- $E(i, j)$ sits at the east face of the tracer cell and so is where the zonal velocity component $u_{i,j,k}$ sits on the C-grid.
- $N(i, j)$ sits at the north face of the tracer cell and so is where the meridional velocity $v_{i,j,k}$ sits on the C-grid.

The geographical coordinates of these four points is sufficient to place them on the discrete lattice.

9.2.2 Horizontal layout of wet and dry cells

The ocean land-sea boundary is fractal in nature, with each refinement in resolution introducing new smaller scale features. The representation of the land-sea boundary is thus fundamental to the utility of an ocean model for realistic simulations. We outline in this section the arrangement of grid variables on both the B and C grids of MOM, with reference made to Figure 9.4.

Figure 9.4 depicts an array of tracer cells, each with a corresponding northeast corner point denoted by an **X**. On the B-grid, the northeast corner is where both velocity components, $u_{i,j,k}$ and $v_{i,j,k}$ are located, whereas for the C-grid this is where the vorticity $\zeta_{i,j,k}$ sits. Any corner point that touches a land cell will have both components of the B-grid velocity set to zero. Arrows crossing the zonal and meridional faces of a cell depict the tracer flux moving across the cell faces. There are no arrows drawn entering land, due to the no-normal flow boundary condition. On the C-grid, arrows also represent the horizontal velocity components. For both the B and C grids, arrows depict the advective and diffusive tracer flux components.

A fundamental distinction between the B and C grids is their treatment of narrow straights and through-flows. We illustrate this distinction by examining the advective tracer transport through tracer cells $T(3, 2)$

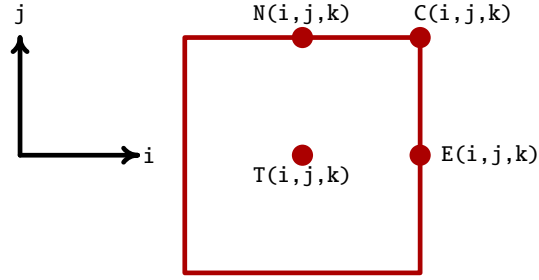


Figure 9.3: The four basic grid points for the B and C grids that surround the fundamental tracer cell. $T(i, j, k)$ is the usual tracer point; $C(i, j, k)$ is the corner point; $E(i, j, k)$ is on the east side; and $N(i, j, k)$ is on the north side. The corner point is the position for the two horizontal velocity components for the B-grid, whereas it is the vorticity position on the C-grid. The east and north points are the position of the zonal and meridional velocity components on the C-grid.

and $T(4, 2)$. For these cells, the B-grid horizontal velocity components are zero, since the corner points touch land. Hence, there is zero zonal advective flux through these cells for the B-grid. We conclude that to allow advective tracer transport on the B-grid requires no less than two adjacent ocean tracer cells. In contrast, the C-grid allows for advective tracer transport through a single tracer cell, and so has nonzero advective tracer transport through tracer cells $T(3, 2)$ and $T(4, 2)$.

9.2.3 Computing the grid distances

To support a discrete calculus for casting the model equations on a grid, we must specify distances between grid points and the grid cells. Knowing the geographical position of the four basic grid points as well as the vertices of their corresponding grid cells is not sufficient. In addition, we need information regarding the metric or stretching functions specific to the coordinate system used to tile the sphere.

The traditional approach is to use spherical coordinates for tiling the sphere. In this method, the distance between two points zonally displaced a finite distance from one another is given by the analytic formula

$$\Delta x[a, b] = R \cos \phi \int_{\lambda_a}^{\lambda_b} d\lambda = (R \cos \phi)(\lambda_b - \lambda_a), \quad (9.2)$$

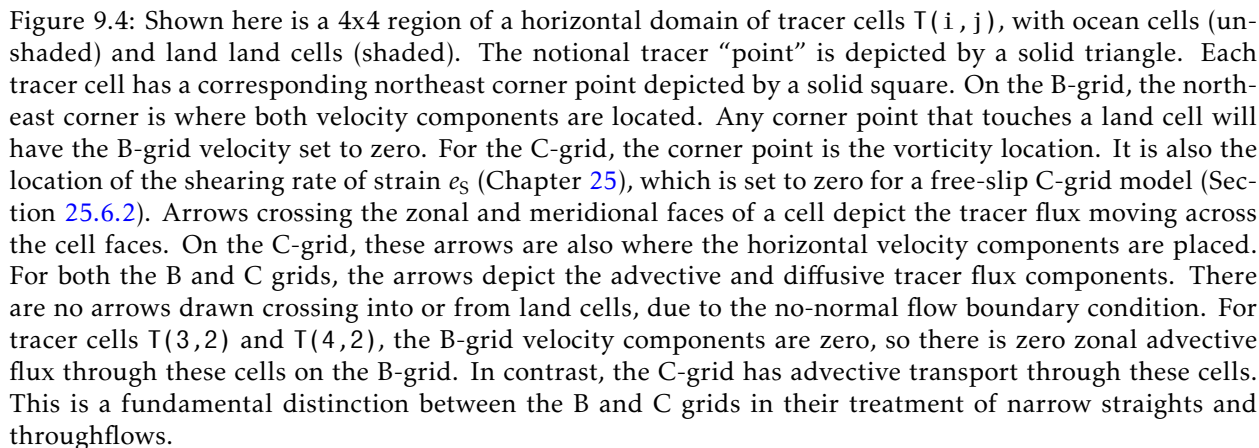
and the distance between two points along a line of constant longitude is given by

$$\Delta y[a, b] = R \int_{\phi_a}^{\phi_b} d\phi = R(\phi_b - \phi_a). \quad (9.3)$$

Writing these expressions in a general manner leads to the generalized zonal and generalized meridional distance given by

$$\Delta x[a, b] = \int_{\xi_1^{(b)}}^{\xi_1^{(a)}} h_1 d\xi_1 \quad (9.4)$$

$$\Delta y[a, b] = \int_{\xi_2^{(b)}}^{\xi_2^{(a)}} h_2 d\xi_2, \quad (9.5)$$



where (ξ_1, ξ_2) represent generalized orthogonal coordinates, and (h_1, h_2) are the stretching functions specific to the coordinate system. They determine the distance between two infinitesimally close points via the line element formula

$$(ds)^2 = (h_1 d\xi_1)^2 + (h_2 d\xi_2)^2. \quad (9.6)$$

With $dx = h_1 d\xi_1$ and $dy = h_2 d\xi_2$, the line element formula takes the form of the usual Cartesian expression

$$(ds)^2 = (dx)^2 + (dy)^2. \quad (9.7)$$

MOM makes use of dx and dy , with units of metre, to allow for cleaner expressions of length, area, and volume.

It is not possible to perform the distance integrals analytically for an arbitrary general orthogonal coordinate system. Therefore, approximations must be made. Indeed, in MOM3 the analytical form for the zonal distance was actually approximated according to

$$\Delta x \approx R \cos \bar{\phi} \quad (9.8)$$

where $\bar{\phi} = (\phi_1 + \phi_2)/2$ (see discussion in Section 39.6 of [Pacanowski and Griffies \(1999\)](#)). Assuming information is available only at the grid points and at the cell vertices, MOM chooses to compute the distance between two points along a generalized zonal direction (i -line) as

$$\Delta x[a, b] = \left| \xi_1^{(a)} - \xi_1^{(b)} \right| \left(h_1^{(a)} + h_1^{(b)} \right) / 2. \quad (9.9)$$

Likewise, the distance along a generalized meridional direction (j -line) is computed as

$$\Delta y[a, b] = \left| \xi_2^{(a)} - \xi_2^{(b)} \right| \left(h_2^{(a)} + h_2^{(b)} \right) / 2. \quad (9.10)$$

9.2.4 Grid distances carried by the model

Given coordinates for the grid points and grid vertices, as well as the stretching functions evaluated at these points, we can use the approximate expressions (9.9) and (9.10) to compute distances between the T,U,N, and E points. Figure 9.5 shows the notation for the grid distances that define four quarter-cells splitting up each tracer and velocity cell. Shown is the notation used in the *grid descriptor* module as well as that used in MOM. The full dimensions of the tracer and velocity cells are shown in Figure 9.6, where again the distances computed in the *grid descriptor* module are translated into the grid distances used in MOM. Finally, Figure 9.7 shows the distances specifying the separation between adjacent tracer and velocity points.

9.3 The Murray (1996) tripolar grid

The [Murray \(1996\)](#) tripolar grid (see his Figure 7) has been a focus of ocean climate model development with MOM and GOLD during 2001-2002. This grid is comprised of the usual spherical coordinate grid southward of a chosen latitude circle, typically taken at $65^\circ N$. This part of the grid has a single pole over Antarctica, which is of no consequence to the numerical ocean climate model. In the Arctic region, the Murray grid places a bipolar region with two poles situated over land, and so these poles are also of no consequence to the numerical ocean model.

Figure 9.8 illustrates the grid lines used to discretize the ocean equations in the Arctic using Murray's grid. The placement of discrete model tracer and velocity points along the bipolar grid lines is schematically represented in Figure 9.9. The arrangement of northern and eastern vector components centered on the tracer cell faces is shown in Figure 9.10. Details for how to transfer information across the bipolar *prime meridian* located along the $j = nj$ line are provided in Section 9.4.

Motivation for choosing the [Murray \(1996\)](#) grid includes the following:

- It removes the spherical coordinate singularity present at the geographical north pole.

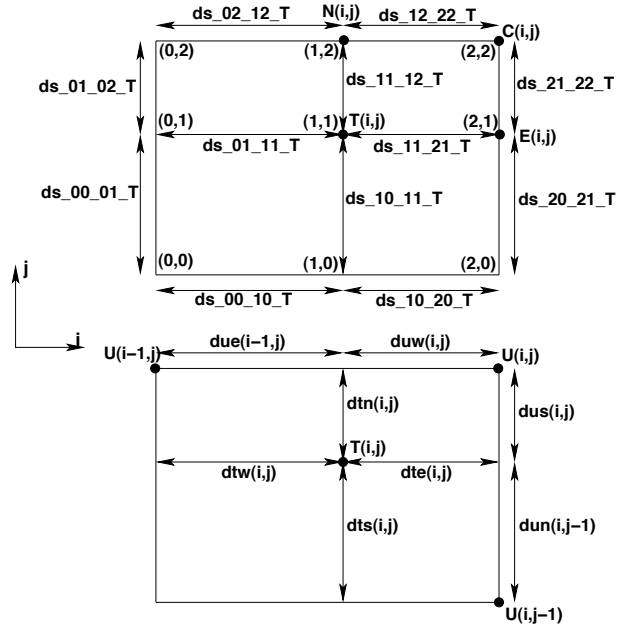


Figure 9.5: Upper panel: Grid distances used to measure the distance between the four fundamental grid points shown in Figure 9.3. These distances are computed in the FMS *grid descriptor* module. The naming convention is based on a Cartesian grid with the origin at the lower left corner of the tracer cell at (0,0), the upper right hand corner is (2,2), the center at (1,1), and all other points set accordingly. The distances are then named as distances between these grid points. Note that each tracer cell has a local Cartesian coordinate set as here, and so there is redundancy in the various grid distances. Lower panel: when read into MOM, the grid distances set the distance between the tracer and velocity points used in the model (Figure 9.1) and the sides of the corresponding grid cells. A translation of the upper panel distance names to those used in MOM is made within the module `ocean_core/ocean_grid.F90`. Note that the names in the lower panel are chosen to correspond to a B-grid, though the lengths are used for both B and C grid calculations.

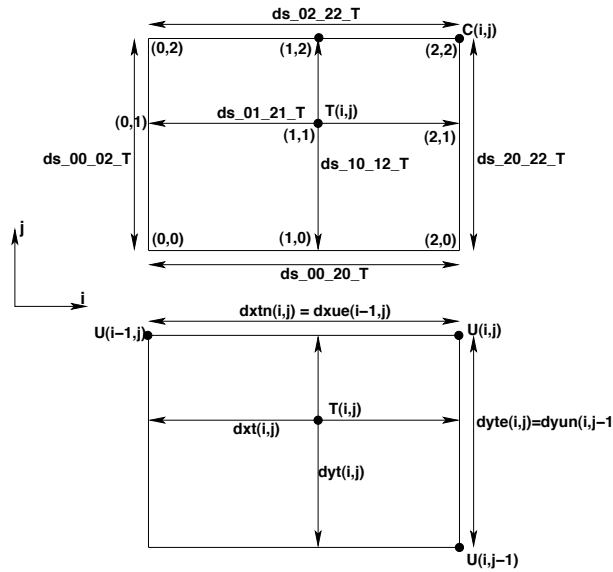


Figure 9.6: Grid cell distances used for computing the area of a grid cell. These dimensions are related to the fundamental quarter-cell dimensions shown in Figure 9.5. Upper panel: distances computed in the FMS *grid_descriptor* module. Lower panel: names of the distances used in MOM. Note that the names in the lower panel are chosen to correspond to a B-grid, though the lengths are used for both B and C grid calculations.

- It maintains the usual spherical coordinate grid lines for latitudes southward of the Arctic region, thus simplifying analysis.
- The grid resolution in the Arctic is more isotropic than the alternative approach of a displaced pole used in simulations with POP (Smith et al., 1995), with isotropic grids generally preferred for numerical accuracy.
- It is locally orthogonal, and so can be used with the MOM generalized horizontal coordinates.
- A similar global grid has been successfully run by the GOLD model code at GFDL (Dunne et al., 2012), as well as the European NEMO modeling group (Madec and Imbard, 1996).

9.4 Specifying fields and grid distances within halos

MOM has been designed to run on multiple parallel processors. The computation of finite derivative operators requires the passage of information across processor boundaries. In particular, the decomposition of the model's *global* domain into multiple *local* domains requires that fields and grid information from one local domain be mapped to halos of adjacent local domains. For second order numerics, the calculation of derivatives on the boundary of a local domain requires information within one grid row *halo* surrounding the local domain. Higher order numerics require larger halos.

9.4.1 Interior domains

Within the interior of the ocean model, away from global boundaries, the mapping between domains is performed using an FMS utility that fills the halo points for one local domain using information available to another local domain. Figure 9.11 illustrates this basic point. Shown is a central processor, arbitrarily labelled PE(0), and a surrounding hatched region representing halo points. The width of the halo is a function of the numerics used in the model. For second order numerics, a halo width of a single point is

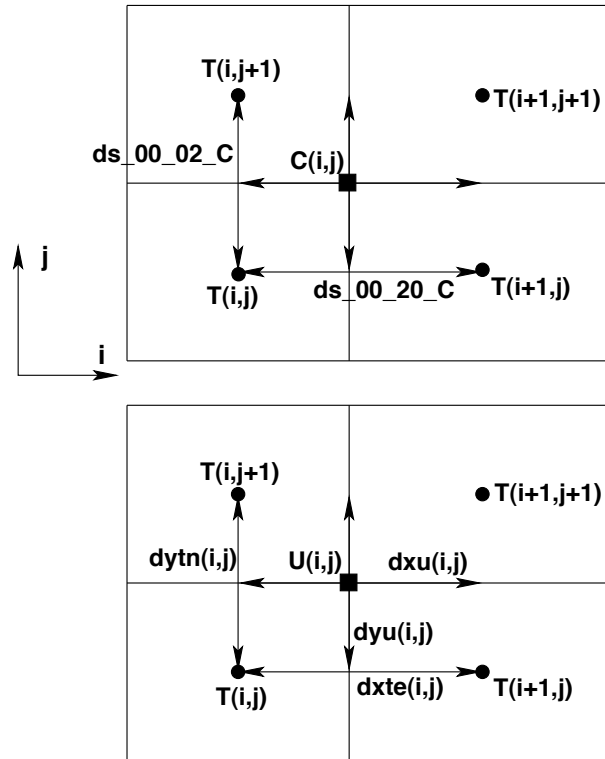


Figure 9.7: Distances between fundamental grid points (upper panel) as computed by the *grid descriptor* module. These distances are taken into MOM and used to set the distances between tracer and velocity points (lower panel). Note that the names are chosen to correspond to a B-grid, though the lengths are used for both B and C grid calculations.

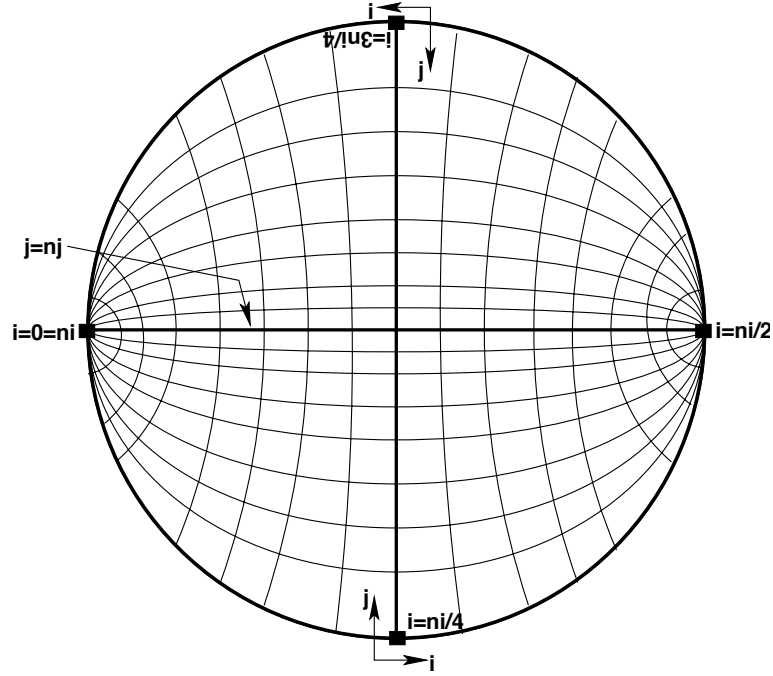


Figure 9.8: Illustration of the grid lines forming the bipolar region in the Arctic. This figure is taken after Figure 7 of Murray (1996). The thick outer boundary is a line of constant latitude in the spherical coordinate grid. This latitude is typically at the latitude nearest to 65°N . As in the spherical coordinate region, lines of constant i move in a generalized eastward direction. They start from the bipolar south pole at $i = 0$, which is identified with $i = ni$. The bipolar north pole is at $i = ni/2$. As shown in Figure 9.9, the poles are centered at a velocity point. Lines of constant j move in a generalized northward direction. The bipolar prime-meridian is situated along the j -line with $j = nj$. This line defines the *bipolar fold* that bisects the tracer grid. Its fold topology causes the velocity points centered along $j = nj$ to have a two-fold redundancy (see Figure 9.9 for more details).

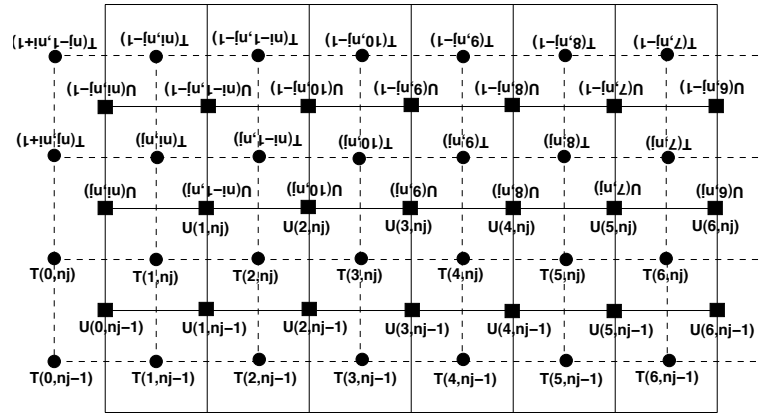


Figure 9.9: Schematic representation of the tracer and velocity cells on the bipolar grid shown in Figure 9.8. The global computational domain consists of $ni = 12$ i -points for this example. The $j = nj$ line bisects the tracer grid, which means there are redundant velocity points along this line. Along an i -line of velocity points, velocity cells with $i = ni/2$ live at the bipolar north pole, whereas velocity cells with $i = 0 = ni$ live at the bipolar south pole.

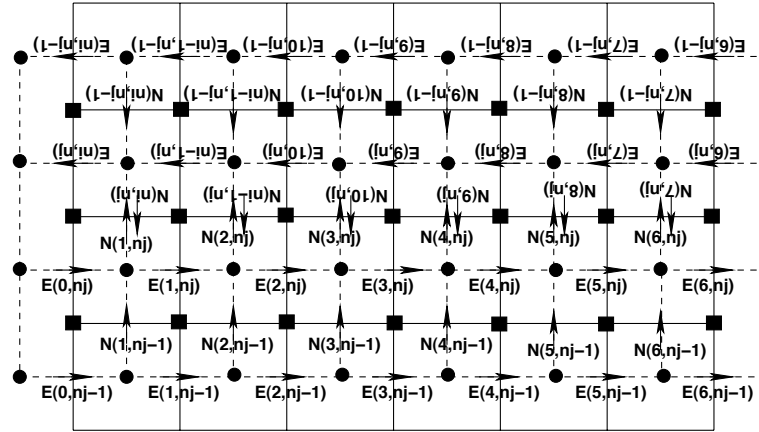


Figure 9.10: Schematic representation of fields living at the north and east faces of the tracer cells as configured using the bipolar grid shown in Figure 9.8. Typical fields of this sort are diffusive and advective tracer flux components, and so they are components to a vector field, hence the vector notation. The global computational domain consists of $n_i = 12$ i -points for this example. The $j = n_j$ line bisects the tracer grid, which means there are redundant velocity points along this line. Along an i -line of velocity points, velocity cells with $i = n_i/2$ live at the bipolar north pole, whereas velocity cells with $i = 0 = n_i$ live at the bipolar south pole.

sufficient, whereas fourth order numerics requires two grid points in a halo. The values of fields and grid factors within the halo are transmitted from the surrounding processors to PE(0) in order for PE(0) to time step its portion of the ocean equations discretized on its local domain.

9.4.2 Exterior domains

For processors whose boundary touches the global model boundary, it is necessary to specify whether the global boundary is a solid wall as in a sector model, periodic as in a zonal channel, or folded as in the bipolar grid of Murray (1996). Each of these three topologies requires special consideration of the mpp code used for transmitting information across processor boundaries. The information about grid topology is defined in the grid specification file during the preprocessing step used to create the grid. We focus here on the three common topologies supported by MOM. A fourth case, radiating open boundaries, is discussed separately in chapter 16 (see also Herzfeld et al. (2011)).

9.4.2.1 Solid wall boundary conditions

For a solid wall boundary condition, all fluxes passing across the walls are zeroed out via masks, and fields within the solid wall are either trivial or masked. Hence, no halo updates are necessary for fields and fluxes at solid walls. However, it is important to specify self-consistent grid distances separating points within the solid wall from those within the model's computational domain. The reason is that various remapping operators require grid distances be well defined for all points within the computational domain, including those distances reaching into the halo. See Chapter 15 for details of remapping operators. For this reason, we extend the grid into the solid wall halo so that resolution in this region is given by the resolution between the two nearest interior points.

9.4.2.2 Periodic boundary conditions

Zonally periodic channels (x -cyclic) are commonly run for idealized studies. Additionally, for realistic global domains, the zonal direction is periodic. Meridionally periodic (y -cyclic) domains may also be of interest for simulations on an f -plane or β -plane. For these reasons, we need to specify grid factors within the halo assuming periodicity at the global domain boundary.

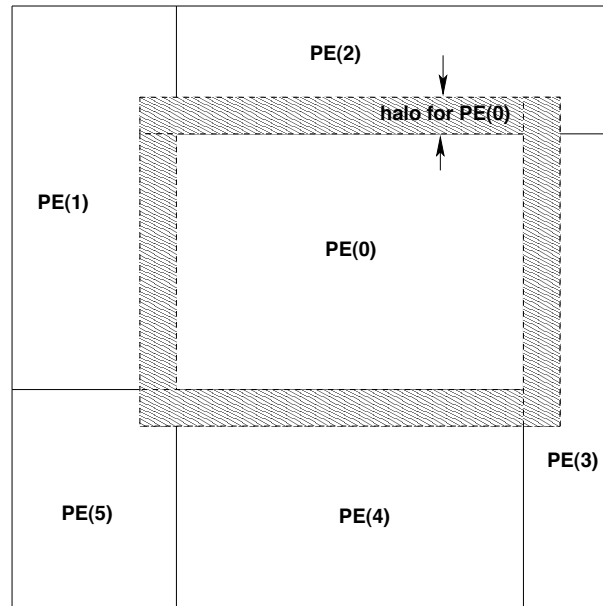


Figure 9.11: Elements of halos needed for computing finite difference operators on a local or *computational* domain. The hatched region is comprised of halo points needed for the processor labelled PE(0) in order to time step its equations. The halo values must be transmitted from the surrounding processors, since they live outside of PE(0)'s local or computational domain. The union of the halo region plus the computational domain defines the *data domain*. Fields that must be known in both the halo region and computational domain have their array sizes set by the data domain. Most fields in MOM are routinely dimensioned over the data domain, even if their halos values are never required.

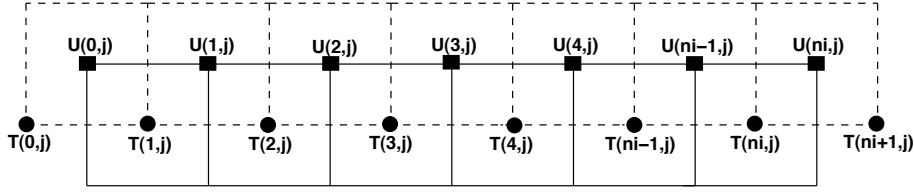


Figure 9.12: A zonally periodic array of tracer and velocity (assuming B-grid placements) points with a single halo point. In this example there are $ni = 6$ points in the global computational domain, and $halo = 1$ point in the surrounding halo region. The cyclic mapping leads us to specify halo points with values $T_{i=0,j} = T_{i=ni,j}$, $T_{i=ni+1,j} = T_{i=1,j}$, and $U_{i=0,j} = U_{i=ni,j}$.

We focus here on the needs of the more common zonally periodic boundary conditions, and refer to Figure 9.12. The same considerations hold for y-cyclic conditions. For either case, we envision the grid wrapped onto itself in the appropriate direction. With second order numerics, computation of the prognostic tracer in grid cells $T_{i=1,j}$ requires information regarding $T_{i=0,j}$. Likewise, $T_{i=ni,j}$ requires information about $T_{i=ni+1,j}$. Higher order numerics will need to reach out further.

First consider the eastern boundary of the domain where $i = ni$. For a single grid halo, we need to specify values of fields living at the T , E , N , and C points at $i = ni + 1$ (recall Figures 9.1 and 9.3 where the C point is equivalent to the B-grid U point). Zonal periodicity renders the equalities

$$T_{ni+1,j} = T_{1,j} \quad (9.11)$$

$$E_{ni+1,j} = E_{1,j} \quad (9.12)$$

$$N_{ni+1,j} = N_{1,j} \quad (9.13)$$

$$C_{ni+1,j} = C_{1,j}. \quad (9.14)$$

More generally, halo points with $ni < i \leq ni + halo$ acquire the x-cyclic mapping

$$T_{i,j} = T_{i-ni,j} \quad (9.15)$$

$$E_{i,j} = E_{i-ni,j} \quad (9.16)$$

$$N_{i,j} = N_{i-ni,j} \quad (9.17)$$

$$C_{i,j} = C_{i-ni,j}. \quad (9.18)$$

At the western boundary, similar considerations lead to halo points $1 - halo \leq i < 1$ mapped to interior points according to

$$T_{i,j} = T_{i+ni,j} \quad (9.19)$$

$$E_{i,j} = E_{i+ni,j} \quad (9.20)$$

$$N_{i,j} = N_{i+ni,j} \quad (9.21)$$

$$C_{i,j} = C_{i+ni,j}. \quad (9.22)$$

9.4.3 The bipolar Arctic grid

The ideas considered for the cyclic case are now generalized to the more complex topology of the Murray (1996) bipolar grid shown in Figures 9.8 and 9.9. In particular, Figures 9.9 and 9.10 allow us to deduce the mappings between related points on the grid. We focus here on the B-grid naming conventions, and assume that both horizontal velocity components both sit at the corner point C . However, C-grid relations follow by placing the zonal velocity component at the east point, E , and meridional velocity component at the north point, N .

9.4.3.1 Fields defined at points T,U,N, and E

The generalized zonal direction (along a constant i -line) is treated with the x-cyclic conditions shown Figure 9.12. It is the bipolar prime meridian along the j -line with $j = nj$ that introduces the most subtle issues. This line bisects the tracer grid. Relating points across the prime meridian requires knowledge of the tensorial nature of the field being considered. In particular, scalar fields map without a change in sign, whereas components of a vector field have a sign change.

The U -points contain a two-fold redundancy of points along the $j = nj$ line. For scalars living at these points, such as some grid factors, we have the identity

$$U_{i,nj} = U_{ni-i,nj}. \quad (9.23)$$

Likewise, scalars living at the northern face of a tracer cell contain a two-fold redundancy of points along the $j = nj$ line so that

$$N_{i,nj} = N_{ni-i+1,nj}. \quad (9.24)$$

For vector components living at U -points, such as the B-grid horizontal velocity field, we associate transition across the $j = nj$ meridian with a sign change

$$u_{i,nj} = -u_{ni-i,nj} \quad (9.25)$$

$$v_{i,nj} = -v_{ni-i,nj}. \quad (9.26)$$

This sign change takes the right handed orientation into a right handed orientation across the meridian. Likewise, for components of vector fluxes living at the north face of a tracer cell, we have

$$F_{N_{i,nj}}^n = -F_{N_{ni-i+1,nj}}^n. \quad (9.27)$$

Note that numerical roundoff may compromise these equalities in the model. Such compromise will generally make the model energetics appear to be larger than when running with the spherical grid, or with the tripolar grid with the fold closed (*debug_tripolar = .true.*).

Moving along a j -line, halo points for scalar fields with $nj < j \leq nj + halo$ are evaluated according to the following rules

$$\left. \begin{aligned} T_{i,j} &= T_{ni-i+1,2nj-j+1} \\ U_{i,j} &= U_{ni-i,2nj-j} \\ N_{i,j} &= N_{ni-i+1,2nj-j} \\ E_{i,j} &= E_{ni-i,2nj-j+1} \end{aligned} \right\} \text{for } nj < j \leq nj + halo \quad (9.28)$$

Vector components living at these points have the same index mapping along with a sign flip for the field values.

9.4.3.2 Grid distances for horizontal quarter-cells

Grid distances must also be specified in halo points. Some distances also maintain redundancy relations. Since grid distances are taken between T,U,N, or E points, their redundancy relations and halo mappings are determined by those of their endpoints. We start by considering the grid factors defining the dimensions of quarter-cells defined in Figure 9.5. These require the most care. Figure 9.13 illustrates the placement of these factors on the bipolar grid. Immediately we see that the two-fold redundancy in the velocity cells $U_{i,nj}$ leads to the two-fold redundancy in grid cell distances

$$due_{i,nj} = duw_{ni-i,nj} \quad (9.29)$$

$$duw_{i,nj} = due_{ni-i,nj} \quad (9.30)$$

$$dun_{i,nj} = dus_{ni-i,nj} \quad (9.31)$$

$$dus_{i,nj} = dun_{ni-i,nj}. \quad (9.32)$$

Now consider the mappings needed to evaluate distances within halos. First consider the distances associated with the tracer cells. By definition, $dte_{i,j}$ measures the distance between the tracer point $T_{i,j}$

and its “eastern” neighbor $E_{i,j}$, and $dtw_{i,j}$ is the distance between $T_{i,j}$ with its “western” neighbor $E_{i-1,j}$, where “eastern” and “western” are in a generalized sense. Mathematically, these distances are

$$\Delta x(T_{i,j}, E_{i,j}) = dte_{i,j} \quad (9.33)$$

$$\Delta x(T_{i,j}, E_{i-1,j}) = dtw_{i,j} \quad (9.34)$$

where $\Delta x(A, B)$ is the distance between points A and B computed according to the generalized zonal distance in equation (9.9). The question is how to map these distances across the bipolar fold. To do so, we note that if we are in a halo region where $nj < j \leq nj + halo$, then the scalar mappings given by equation (9.28) lead to

$$\Delta x(T_{i,j}, E_{i,j}) = \Delta x(T_{ni-i+1, 2nj-j+1}, E_{ni-i, 2nj-j+1}) \quad (9.35)$$

$$\Delta x(T_{i,j}, E_{i-1,j}) = \Delta x(T_{ni-i+1, 2nj-j+1}, E_{ni-i+1, 2nj-j+1}). \quad (9.36)$$

Comparison of these equalities with the definitions of dte and dtw then leads to the halo cell relations

$$\left. \begin{aligned} dte_{i,j} &= dtw_{ni-i+1, 2nj-j+1} \\ dtw_{i,j} &= dte_{ni-i+1, 2nj-j+1} \end{aligned} \right\} \text{for } nj < j \leq nj + halo \quad (9.37)$$

Distances to the northern and southern faces of the tracer cell, dtn and dts , are defined by

$$\Delta y(T_{i,j}, N_{i,j}) = dtn_{i,j} \quad (9.38)$$

$$\Delta y(T_{i,j}, N_{i,j-1}) = dts_{i,j} \quad (9.39)$$

where Δy is the generalized meridional distance given by equation (9.10). Equation (9.28) indicate that within the halo region $nj < j \leq nj + halo$,

$$\Delta y(T_{i,j}, N_{i,j}) = \Delta y(T_{ni-i+1, 2nj-j+1}, N_{ni-i+1, 2nj-j}) \quad (9.40)$$

$$\Delta y(T_{i,j}, N_{i,j-1}) = \Delta y(T_{ni-i+1, 2nj-j+1}, N_{ni-i+1, 2nj-j+1}). \quad (9.41)$$

Comparison of these equalities with the definitions of dtn and dts leads to the halo cell relations

$$\left. \begin{aligned} dtn_{i,j} &= dts_{ni-i+1, 2nj-j+1} \\ dts_{i,j} &= dtn_{ni-i+1, 2nj-j+1} \end{aligned} \right\} \text{for } nj < j \leq nj + halo \quad (9.42)$$

Velocity cell distances are defined by

$$\Delta x(U_{i,j}, N_{i+1,j}) = due_{i,j} \quad (9.43)$$

$$\Delta x(U_{i,j}, N_{i,j}) = duw_{i,j} \quad (9.44)$$

$$\Delta y(U_{i,j}, E_{i,j+1}) = dun_{i,j} \quad (9.45)$$

$$\Delta y(U_{i,j}, E_{i,j}) = dus_{i,j} \quad (9.46)$$

Equation (9.28) indicate that within the halo region $nj < j \leq nj + halo$,

$$\Delta x(U_{i,j}, N_{i+1,j}) = \Delta x(U_{ni-i, 2nj-j}, N_{ni-i, 2nj-j}) \quad (9.47)$$

$$\Delta x(U_{i,j}, N_{i,j}) = \Delta x(U_{ni-i, 2nj-j}, N_{ni-i+1, 2nj-j}) \quad (9.48)$$

$$\Delta y(U_{i,j}, E_{i,j+1}) = \Delta x(U_{ni-i, 2nj-j}, E_{ni-i, 2nj-j}) \quad (9.49)$$

$$\Delta y(U_{i,j}, E_{i,j}) = \Delta x(U_{ni-i, 2nj-j}, E_{ni-i, 2nj-j+1}), \quad (9.50)$$

which then leads to the halo cell relations

$$\left. \begin{aligned} due_{i,j} &= duw_{ni-i, 2nj-j} \\ duw_{i,j} &= due_{ni-i, 2nj-j} \\ dun_{i,j} &= dus_{ni-i, 2nj-j} \\ dus_{i,j} &= dun_{ni-i, 2nj-j} \end{aligned} \right\} \text{for } nj < j \leq nj + halo \quad (9.51)$$

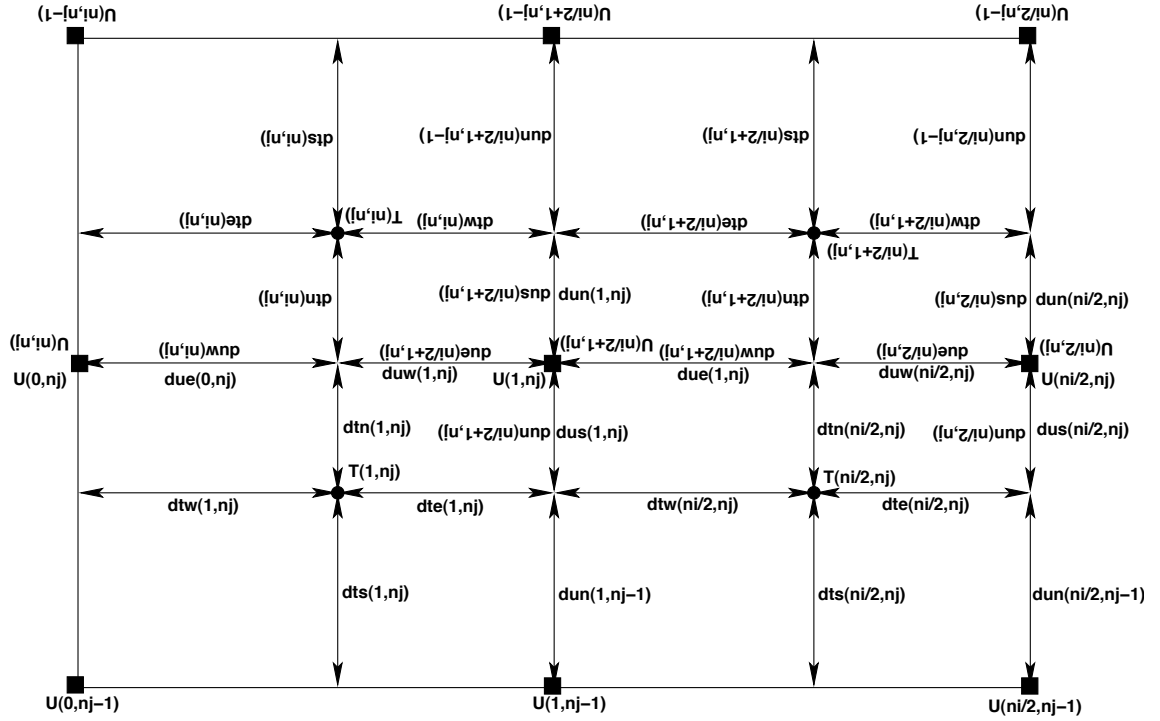


Figure 9.13: Placement of quarter-cells distances at the bipolar fold. For this example, there are $ni = 4$ points in the generalized zonal computational domain. Equivalence of grid factors on the fold leads to the two-fold redundancy for velocity cell distances $due_{i,nj} = duw_{ni-i,nj}$ and $dus_{i,nj} = dun_{ni-i,nj}$.

9.4.3.3 Grid distances for horizontal full cells

Inspection of Figures 9.6 and 9.7, with the definitions of grid points shown in Figure 9.3, leads to the definitions of distances for full horizontal cells

$$\Delta x(E_{i-1,j}, E_{i,j}) = dx_{t_{i,j}} \quad (9.52)$$

$$\Delta y(N_{i,j-1}, N_{i,j}) = dy_{t_{i,j}} \quad (9.53)$$

$$\Delta x(U_{i-1,j}, U_{i,j}) = dx_{tn_{i,j}} \quad (9.54)$$

$$\Delta y(U_{i,j}, U_{i,j-1}) = dy_{te_{i,j}} \quad (9.55)$$

$$\Delta x(T_{i,j}, T_{i+1,j}) = dx_{te_{i,j}} \quad (9.56)$$

$$\Delta y(T_{i,j}, T_{i,j+1}) = dy_{tn_{i,j}} \quad (9.57)$$

$$\Delta x(N_{i,j}, N_{i+1,j}) = dx_{u_{i,j}} \quad (9.58)$$

$$\Delta y(E_{i,j}, E_{i,j+1}) = dy_{u_{i,j}} \quad (9.59)$$

Figures 9.14, 9.15, and 9.16 show these distances for regions surrounding the bipolar fold. To generate the redundancy conditions and halo mappings, we again use the scalar mappings given by equation (9.28). Using these relations we see that redundancy is satisfied by the distances

$$dx_{tn_{i,nj}} = dx_{tn_{ni-i+1,nj}} \quad (9.60)$$

$$dy_{tn_{i,nj}} = dy_{tn_{ni-i+1,nj}} \quad (9.61)$$

$$dx_{u_{i,nj}} = dx_{u_{ni-i,nj}} \quad (9.62)$$

$$dy_{u_{i,nj}} = dy_{u_{ni-i,nj}} \quad (9.63)$$

Equation (9.28) indicates that within the halo region $nj < j \leq nj + halo$,

$$\Delta x(E_{i-1,j}, E_{i,j}) = \Delta x(E_{ni-i+1,2nj-j+1}, E_{ni-i,2nj-j+1}) \quad (9.64)$$

$$\Delta y(N_{i,j-1}, N_{i,j}) = \Delta y(N_{ni-i+1,2nj-j+1}, N_{ni-i+1,2nj-j}) \quad (9.65)$$

$$\Delta x(U_{i-1,j}, U_{i,j}) = \Delta x(U_{ni-i+1,2nj-j}, U_{ni-i,2nj-j}) \quad (9.66)$$

$$\Delta y(U_{i,j}, U_{i,j-1}) = \Delta y(U_{ni-i,2nj-j}, U_{ni-i,2nj-j+1}) \quad (9.67)$$

$$\Delta x(T_{i,j}, T_{i+1,j}) = \Delta x(T_{ni-i+1,2nj-j+1}, T_{ni-i,2nj-j+1}) \quad (9.68)$$

$$\Delta y(T_{i,j}, T_{i,j+1}) = \Delta y(T_{ni-i+1,2nj-j+1}, T_{ni-i+1,2nj-j}) \quad (9.69)$$

$$\Delta x(N_{i,j}, N_{i+1,j}) = \Delta x(N_{ni-i+1,2nj-j}, N_{ni-i,2nj-j}) \quad (9.70)$$

$$\Delta y(E_{i,j}, E_{i,j+1}) = \Delta y(E_{ni-i,2nj-j+1}, E_{ni-i,2nj-j}) \quad (9.71)$$

which then leads to the halo cell relations

$$\left. \begin{aligned} dx_{t_{i,j}} &= dx_{t_{ni-i+1,2nj-j+1}} \\ dy_{t_{i,j}} &= dy_{t_{ni-i+1,2nj-j+1}} \\ dx_{tn_{i,j}} &= dx_{tn_{ni-i+1,2nj-j}} \\ dy_{te_{i,j}} &= dy_{te_{ni-i,2nj-j+1}} \\ dx_{te_{i,j}} &= dx_{te_{ni-i,2nj-j+1}} \\ dy_{tn_{i,j}} &= dy_{tn_{ni-i+1,2nj-j}} \\ dx_{u_{i,j}} &= dx_{u_{ni-i,2nj-j}} \\ dy_{u_{i,j}} &= dy_{u_{ni-i,2nj-j}} \end{aligned} \right\} \text{for } nj < j \leq nj + halo \quad (9.72)$$

9.4.3.4 Summary of redundancies and halo mappings

Table 9.1 summarizes the halo relations and redundancies realized at the bipolar fold. Notice that those distances exhibiting a redundancy have their halo relations reduce to their redundancy relations for $j = nj$. Additionally, the quarter-cell distances all transform from a right handed system to a right handed system. In general, this table should be sufficient to deduce relations for any derived fields, fluxes, etc., computed in the model.

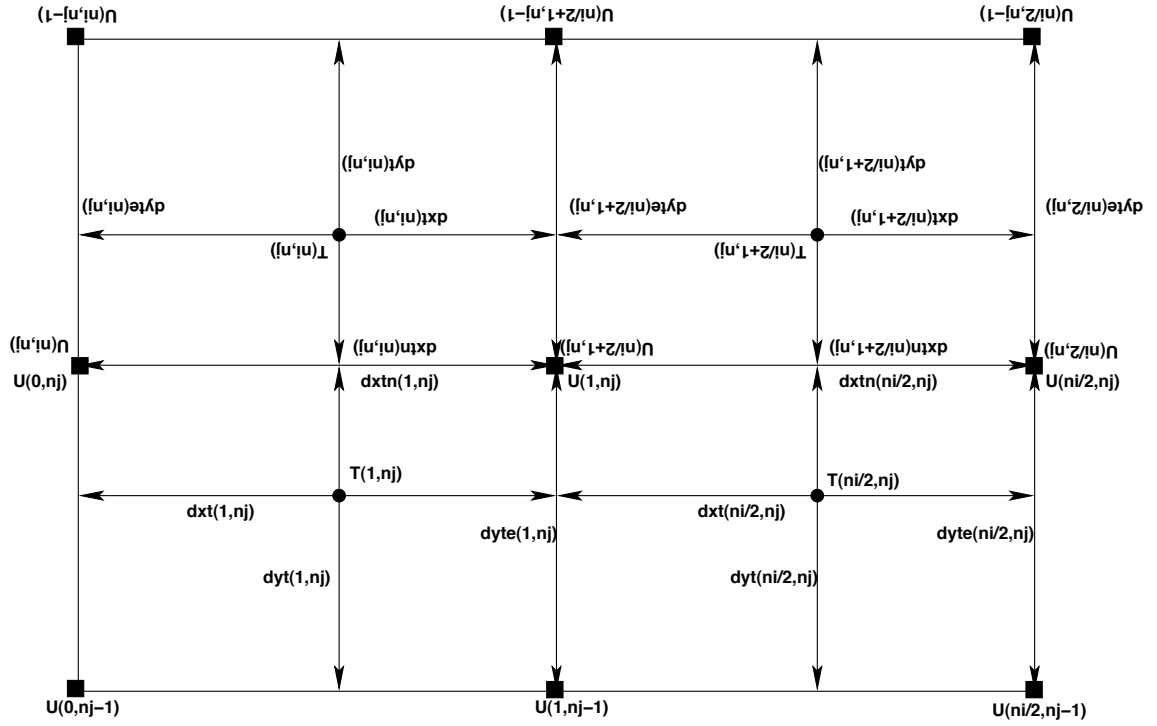


Figure 9.14: Placement of tracer cell dimensions at the bipolar fold. For this example, there are $ni = 4$ points in the generalized zonal computational domain. Equivalence of grid factors on the fold leads to the two-fold redundancy $dxtn_{i,nj} = dxtn_{ni-i+1,nj}$.

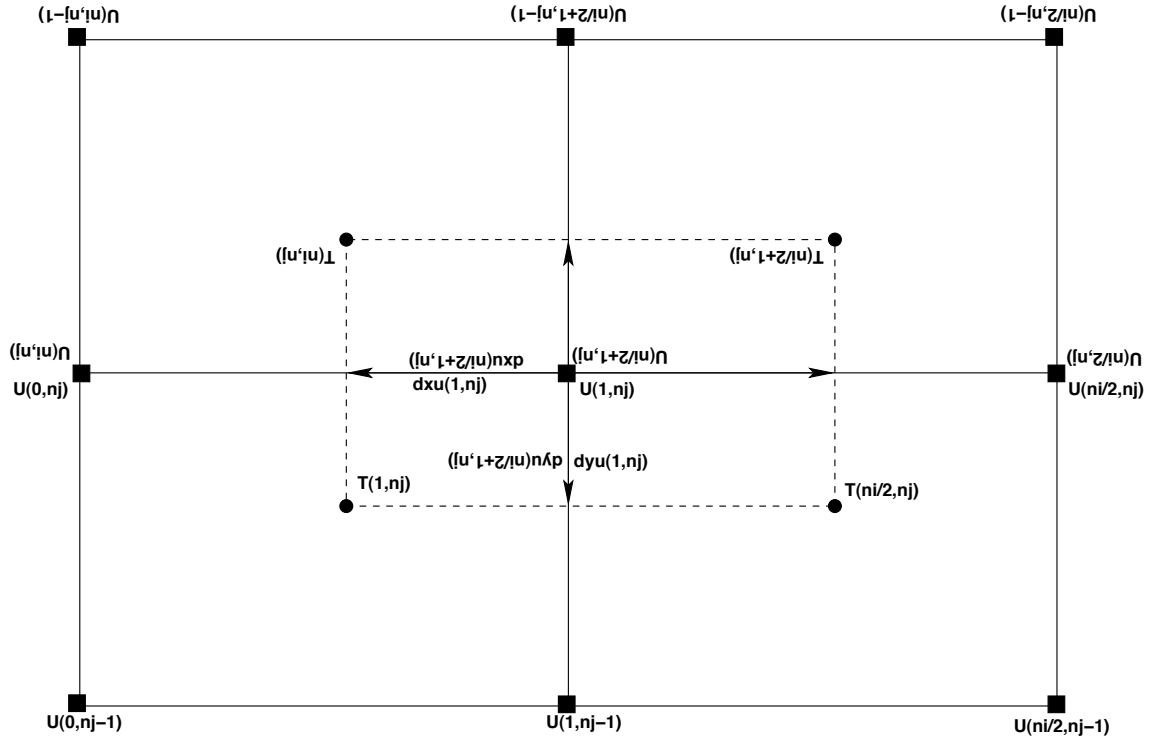


Figure 9.15: Velocity cell distances at the bipolar fold.

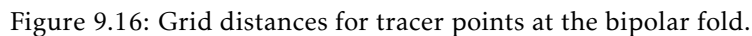


Table 9.1: Summary of the halo mappings and redundancies realized at the bipolar fold. The symbol ε is 1 for scalar fields, and -1 for horizontal components of vector fields.

QUASI-EULERIAN ALGORITHMS FOR HYDROSTATIC MODELS

Contents

10.1	Pressure and geopotential at tracer points	136
10.1.1	Pressure at tracer point: energetic method	137
10.1.2	Pressure at a tracer point: finite volume considerations	137
10.1.3	Discrete geopotential based on energetic considerations	138
10.1.4	Discrete geopotential based on finite volume considerations	138
10.2	Initialization issues	139
10.2.1	Modification of dst	139
10.2.2	Modification of the density field	139
10.2.3	Modification of the bottom topography	139
10.3	Vertical dimensions of grid cells	139
10.3.1	Thickness of a grid cell	140
10.3.2	Vertical distance between tracer points	140
10.3.2.1	Energetic based approach	141
10.3.2.2	Finite volume approach	141
10.4	Summary of vertical grid cell increments	141
10.4.1	Geopotential vertical coordinate	141
10.4.2	z^* vertical coordinate	142
10.4.3	Terrain following $\sigma^{(z)}$ vertical coordinate	142
10.4.4	Non-terrain following pressure vertical coordinate	142
10.4.5	p^* vertical coordinate	145
10.4.6	Steps to initialize pressure and p^* based models	145
10.4.7	Terrain following $\sigma^{(p)}$ coordinate	146
10.5	Surface height and bottom pressure diagnosed	147
10.5.1	Surface height diagnosed in pressure based models	147
10.5.1.1	Concerning nonzero areal average	147
10.5.1.2	Concerning small scale features	147
10.5.2	Bottom pressure diagnosed in depth based models	148
10.6	Vertically integrated volume/mass budgets	148
10.6.1	Vertically integrated volume budget	148
10.6.2	Vertically integrated mass budget	149
10.6.3	Summary of the vertically integrated volume/mass budgets	149

10.7	Compatibility between tracer and mass	150
10.8	Diagnosing the dia-surface velocity component	150
10.8.1	Depth based vertical coordinates	151
10.8.1.1	Depth coordinate	151
10.8.1.2	Depth deviation coordinate	151
10.8.1.3	Zstar coordinate	152
10.8.1.4	Depth-sigma coordinate	152
10.8.1.5	General expression for dia-surface velocity component	152
10.8.2	Pressure based vertical coordinates	153
10.8.2.1	Pressure coordinate	153
10.8.2.2	Pressure deviation coordinate	154
10.8.2.3	Pstar coordinate	154
10.8.2.4	Pressure sigma coordinate	155
10.8.2.5	General expression for the dia-surface velocity component	155
10.8.3	Comments about diagnosing the dia-surface velocity component	155
10.9	Vertically integrated horizontal momentum	156
10.9.1	Budget using contact pressures on cell boundaries	156
10.9.2	Budget using the pressure gradient body force	156
10.9.3	Depth based vertical coordinates	157
10.9.4	Pressure based vertical coordinates	158

Adcroft and Hallberg (2006) characterize two types of primitive equation ocean models. Eulerian vertical coordinate algorithms, such as used in MOM, ROMS, and NEMO, diagnose the dia-surface velocity component from the continuity equation. Lagrangian vertical coordinate algorithms, such as used in GOLD and HYCOM, specify the dia-surface velocity component (e.g., zero diapycnal velocity in adiabatic simulations with isopycnal coordinates). Eulerian in this context does not mean that a grid cell has a time constant vertical position. Hence, the term quasi-Eulerian is used in this chapter.

In this chapter, we develop the semi-discrete budgets of a hydrostatic ocean model and present quasi-Eulerian solution algorithms. Notably, as implemented in MOM, the quasi-Eulerian algorithms are formulated assuming a time independent number of grid cells. That is, MOM does not allow for vanishing cell thickness. This assumption simplifies the algorithms in many ways, but in turn limits the extent to which this code can be used for simulations where water masses change in a nontrivial manner.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_advection_velocity.F90
ocean_core/ocean_barotropic.F90
ocean_core/ocean_pressure.F90
ocean_core/ocean_thickness.F90
ocean_core/ocean_velocity.F90
ocean_core/ocean_velocity_advect.F90
ocean_tracers/ocean_tracer.F90
ocean_tracers/ocean_tracer_advect.F90
```

10.1 Pressure and geopotential at tracer points

We discussed the discrete pressure gradient body force appropriate for a finite difference discretization in Sections 3.2 and 3.3. We require the anomalous hydrostatic pressure in the depth based models, and the anomalous geopotential height in the pressure based models. That is, for depth based vertical coordinate

models, we need a discretization of the anomalous hydrostatic pressure (equation (3.20))

$$p' = g \int_z^0 \rho' dz \quad \text{for } s = z \quad (10.1)$$

$$p' = g \int_z^\eta \rho' dz \quad \text{for } s = z^*, \sigma^{(z)}. \quad (10.2)$$

For pressure based vertical coordinate models, we need a discretization of the anomalous geopotential (equation (3.25))

$$\Phi' = -(g/\rho_o) \int_{-H}^z \rho' dz. \quad (10.3)$$

The vertical integrals involve some ambiguity for the finite difference formulation, since the tracer point is not vertically centred within the tracer cell for the case of a vertically nonuniform grid. In this case, we may choose to compute the pressure and geopotential at the tracer point using a more accurate vertical integration that accounts for the non-centred placement of the tracer point.

The purpose of this section is to describe two methods used for the calculation of the pressure and geopotential at the tracer grid point. Details of this discretization affect the manner used for diagnosing the pressure conversion to buoyancy work, as described in Sections 14.6, 14.7, 14.8, and 14.9. The MOM code provides both choices, with both producing analogous results for the surface height and bottom pressure.

10.1.1 Pressure at tracer point: energetic method

If the equation of state is linear, and both density and velocity are advected with second order centered differences, then the conversion of pressure work to buoyancy work will balance potential energy changes. This equality led Bryan (1969) to formulate the hydrostatic pressure calculation according to

$$p'_{k+1} = p'_k + g \, dzwt_k \, \overline{\rho'}^z_k. \quad (10.4)$$

That is, anomalous hydrostatic pressure is computed given knowledge of the thicknesses $dzwt$ and the density ρ'_k . In this equation, primes refer to anomalies relative to the reference Boussinesq density

$$\rho' = \rho - \rho_o \quad (10.5)$$

and

$$\overline{\rho'}^z_k = (\rho'_k + \rho'_{k+1})/2 \quad (10.6)$$

is the simple vertical average of density. This average is the same as a finite volume average only if the grid cell thicknesses are uniform. With stretched vertical grids, the simple average differs from the finite volume average presented in Section 10.1.2. At the ocean surface, no average is available, so we use the finite volume value for the pressure

$$p'_{k=1} = g \, dzwt_{k=0} \, \rho'_{k=1}. \quad (10.7)$$

Given this surface value, we then integrate downwards according to equation (10.4) to diagnose the anomalous hydrostatic pressure at each discrete k -level.

10.1.2 Pressure at a tracer point: finite volume considerations

Although the finite volume method for computing the pressure force requires the pressure and geopotential to be computed at the bottom of the tracer cells, we may choose to use a finite volume motivated approach for computing the pressure and geopotential at the tracer point. Referring to the right hand panel in Figure 10.1, a finite volume motivated computation of hydrostatic pressure at a tracer point is given by

$$p'_{k=1} = g \, dz \, \text{tup}_{k=1} \, \rho'_{k=1} \quad (10.8)$$

$$p'_{k+1} = p'_k + g \, dz \, \text{tlo}_k \, \rho'_k + g \, dz \, \text{tup}_{k+1} \, \rho'_{k+1}. \quad (10.9)$$

The pressure at $k = 1$ is the same as prescribed in the energetic method. However, for stretched vertical grid cells, the interior cells have a different discrete pressure from that computed in the energetic method. The finite volume approach is more accurate for stretched vertical grids.

10.1.3 Discrete geopotential based on energetic considerations

Following in a manner analogous to the anomalous hydrostatic pressure in Section 10.1.1, we have the discretized anomalous geopotential

$$\Phi'_{k=kbot} = -(g/\rho_o) dzwt_{k=kbot} \rho'_{k=kbot} \quad (10.10)$$

$$\Phi'_k = \Phi'_{k+1} - (g/\rho_o) dzwt_k \overline{\rho'_k}^z. \quad (10.11)$$

Iteration starts from the bottom at $k = kbot$ using the finite volume expression, and moves upward in the column towards the surface.

10.1.4 Discrete geopotential based on finite volume considerations

Following in a manner completely analogous to the anomalous hydrostatic pressure in Section 10.1.2, we have the discretized anomalous geopotential

$$\Phi'_{k=kbot} = -(g/\rho_o) dztl_{kbot} \rho'_{k=kbot} \quad (10.12)$$

$$\Phi'_k = \Phi'_{k+1} - (g/\rho_o) dzstup_{k+1} \rho'_{k+1} - (g/\rho_o) dztl_k \rho'_k. \quad (10.13)$$

Iteration starts from the bottom at $k = kbot$ using the finite volume expression, and moves upward in the column towards the surface.

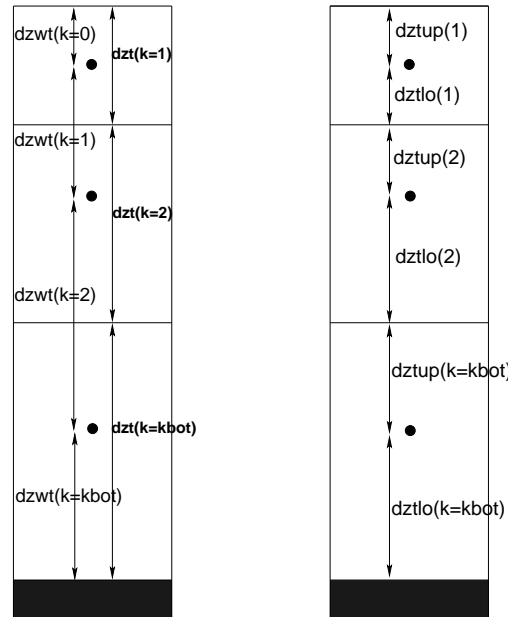


Figure 10.1: A vertical column of three tracer cells and the corresponding vertical cell dimensions. In MOM, the vertical spacing is related by $dzt_k = (dzwt_{k-1} + dzwt_k)/2$. With this specification, the average tracer $\bar{T}^z = (T_k + T_{k+1})/2$ lives at the bottom of the tracer cell T_k and so is co-located with the dia-surface velocity component w_{bt_k} . The right column exposes the half-distances, which measure the distance from the tracer cell point to the top and bottom faces of the tracer cell. The half-distances are used in the finite volume formulation of pressure and geopotential computed at the tracer points (Sections 10.1.2 and 10.1.4), whereas the grid spacing $dzwt$ is used for the energetically based computation of pressure and geopotential computed at the tracer points (Sections 10.1.1 and 10.1.3).

10.2 Initialization issues

When initializing a Boussinesq model, we place a fluid with initial *in situ* density ρ^{init} onto a grid with vertical increments dzt . Hence, both the density and volume of the grid cells are specified. The initial mass of fluid is implied by this initialization method. Furthermore, by definition, the surface elevation η is zero.

For the non-Boussinesq model, we place a fluid with initial *in situ* density ρ^{init} onto a grid with vertical pressure increments dst . Hence, both the density and mass of the grid cells are specified. The initial volume of fluid is thus implied from this initialization method. Furthermore, by definition, the bottom pressure anomaly, $\text{pbot_t} - \text{pbot0}$, is zero if we choose pbot0 as the initial bottom pressure.

The initialization methods are isomorphic. Notably, when initializing the Boussinesq model, there is no guarantee that its bottom pressure anomaly will be initially zero. Likewise, there is no guarantee that the surface elevation η will be zero with the non-Boussinesq initialization. For many applications, the nonzero sea level may be of little concern, with the sea level adjusting rapidly on a barotropic time scale. Nonetheless, we next outline three possible means to ensure a zero sea level results from initializing a non-Boussinesq model. Such may be of interest for careful comparisons between Boussinesq and non-Boussinesq simulations, such as considered by [Losch et al. \(2004\)](#).

10.2.1 Modification of dst

There are three general ways to approach non-Boussinesq initialization. First, we can modify the vertical pressure increments dst of the grid cells to accommodate the initial density and to retain a zero surface height. This approach generally requires nontrivial horizontal deviations in the dst array, so that it has full grid dependence $\text{dst}(i, j, k)$. Such dependence is generally acceptable for the bottom, where partial cells introduce three-dimensional dependence to the vertical grid increments. However, with this added dependence in the ocean interior, there is a possibility for introducing pressure gradient errors, depending on the magnitude of the horizontal variations. If the variations are minor, then this approach may be acceptable.

10.2.2 Modification of the density field

A second approach is to modify the initial density field. This approach, however, may fail after some time integration, depending on the surface forcing. That is, over time the model may be forced towards a density structure similar to the initial structure, in which case the possibility exists for losing the bottom cell in the model if the evolved bottom pressure becomes lighter than the pressure at the top of the bottom cell.

10.2.3 Modification of the bottom topography

A third approach is motivated by one used with the MITgcm. Here, we deepen the bottom topography so that the initial mass (as set by the pressure increments) and density result in vertical columns with zero initial surface height. This approach may appear to be the least desirable, as we know the bottom topography generally more accurately than the initial density. Yet depending on details of the initial density field and the pressure increments, the changes in the bottom topography are often quite minor. We detail this approach in [Section 10.4.4](#).

10.3 Vertical dimensions of grid cells

The density weighted thickness of a grid cell is of fundamental importance in the formulation presented in this document. In particular, density weighted thickness of a tracer cell is a basic ingredient and the values on a velocity cell are diagnosed according to the minimum surrounding tracer cell values. Given these fields, most of the equations for the ocean model retain the same appearance for arbitrary vertical coordinates. The technology of generalized vertical coordinates then resides in the module specifying $\rho \, dz$

COORD	DEFINITION	CELL THICKNESS
geopotential	z	dz
zstar	$z^* = H(z - \eta)/(H + \eta)$	$dz = (1 + \eta/H)dz^*$
z-sigma	$\sigma^{(z)} = (z - \eta)/(H + \eta)$	$dz = (H + \eta)d\sigma^{(z)}$
pressure	p	$dz = -(\rho g)^{-1} dp$
pstar	$p^* = p_b^0(p - p_a)/(p_b - p_a)$	$dz = -[(p_b - p_a)/(\rho g p_b^0)] dp^*$
p-sigma	$\sigma^{(p)} = (p - p_a)/(p_b - p_a)$	$dz = -[(p_b - p_a)/(\rho g)] d\sigma^{(p)}$

Table 10.1: Table of vertical thicknesses dz for grid cells as determined on the tracer grid using the vertical coordinates discussed in Chapter 5. The vertical coordinate increments are specified, and the vertical thicknesses dz are diagnosed.

(the MOM module `ocean_core/ocean_thickness` module), with extra work also needed for the pressure and grid modules.

In addition to the density weighted thicknesses, we are in need of the depth of a grid cell center, depth of the grid cell bottom, and vertical dimensions within the grid cell. Information is needed for these distances both in depth space (z -coordinate), and coordinate space (s -coordinate). These needs introduce new time dependent arrays that are updated and saved for restarts.

Figure 10.1 defines notation for the grid cell thicknesses used in MOM. Here, the left figure exposes the vertical dimensions of the tracer grid cell, dzt and the distance between the T-cell points, dzt . The right figure exposes the half-distances, which measure the distance from the T-cell point to the upper face of the cell, $dztlo$, and the lower face, $dztlo$.

10.3.1 Thickness of a grid cell

The thickness of a grid cell is written

$$dz = z_{,s} ds. \quad (10.14)$$

For a tracer cell, this expression is written in the MOM codes as

$$dzt = dzt_dst * dst. \quad (10.15)$$

Inspection of the results from Tables 5.1 and 5.2 lead to the thicknesses given in Table 10.1, which are again applied to the tracer grid. The corresponding velocity cell thicknesses are diagnosed based on the tracer cell values.

For the finite volume approach to computing the pressure and geopotential, as discussed in Section 10.1.2, we need a method to compute the half-thicknesses. For this purpose, we assume the specific thickness factor `dzt_dst` is constant across the thickness of a tracer cell. We also assume knowledge of the half- s -thicknesses `dstlo` and `dstup`, thus leading to

$$dztlo = dzt_dst * dstlo \quad (10.16)$$

$$dztup = dzt_dst * dstup. \quad (10.17)$$

The full cell thickness is then recovered by setting

$$dzt = dztlo + dztup, \quad (10.18)$$

where

$$dst = dstlo + dstup. \quad (10.19)$$

10.3.2 Vertical distance between tracer points

Through summation from the ocean surface, knowledge of the tracer cell thicknesses dz_k within a vertical column provides the depth of the bottom of any tracer cell within the column. For many purposes, it is also

important to know the depth where the tracer point is located. This information is obtained via vertical summation from the distance between two vertically adjacent tracer cell points. As seen in Section 10.1 when discussing the hydrostatic pressure and the geopotential, the vertical distance between tracer points is known as $dzwt$ and the corresponding velocity cell vertical distance is $dzwu$.

10.3.2.1 Energetic based approach

For depth based vertical coordinates, $dzwt$ is computed according to the results in Table 10.1 given the corresponding coordinate thicknesses $dswt$. For pressure based vertical coordinates using the energetic approach from Section 10.1.1, we are guided by the result (10.4) for the hydrostatic pressure computed in a depth based vertical coordinate model. In general, this expression takes the form

$$ds = \overline{(s_z)}^z dz \quad (10.20)$$

where $\overline{a^z} = (a_k + a_{k+1})/2$ is an unweighted discrete vertical average. Introducing model arrays leads to

$$dzwt_k = \left(\frac{2}{(s_z)_k + (s_z)_{k+1}} \right) dswt_k. \quad (10.21)$$

For example, with $s = p$, this relation takes the form

$$dzwt_k = - \left(\frac{2}{g(\rho_k + \rho_{k+1})} \right) dswt_k, \quad (10.22)$$

where $dswt$ is known and is negative, since pressure decreases upward, whereas geopotential increases upward.

10.3.2.2 Finite volume approach

From the finite volume approach described in Section 10.3.1, we follow expressions (10.16) and (10.17) for the thickness of a grid cell to write

$$dzwt_{k=0} = dztp_{k=1} \quad (10.23)$$

$$dzwt_{k>1} = dztl_{k-1} + dztp_k \quad (10.24)$$

$$dzwt_{k=kbot} = dztl_{k=kbot}. \quad (10.25)$$

10.4 Summary of vertical grid cell increments

We now summarize the results from Section 10.3 for the vertical coordinates z , z^* , $\sigma^{(z)}$, p , p^* , and $\sigma^{(p)}$. The notation used in MOM is used to allow for direct comparison to the model code.

10.4.1 Geopotential vertical coordinate

The geopotential vertical coordinate has the following grid dimensions

$$\begin{aligned} dz_t_dst(i, j, k) &= 1 \\ dzwt(i, j, k=0) &= zt(k=1) + eta_t(i, j) \\ dzt(i, j, k=1) &= zw(k=1) + eta_t(i, j). \end{aligned} \quad (10.26)$$

The initial values of the depth of tracer points, $depth_zt$, remain unchanged in time. However, the thickness of the top cell is time dependent.

10.4.2 z^* vertical coordinate

The z^* coordinate has the grid dimensions

$$\begin{aligned} dz_t_dst(i, j, k) &= 1 + eta_t(i, j)/ht(i, j) \\ dst(i, j, k) &= dz_t(i, j, k)|_{\tau=0} \\ dswt(i, j, k) &= dzwt(i, j, k)|_{\tau=0} \\ dz_t(i, j, k) &= dz_t_dst(i, j, k) * dst(i, j, k). \end{aligned} \quad (10.27)$$

For the energetically based computation of hydrostatic pressure (Section 10.1.1), the distance between tracer points is computed according to

$$\begin{aligned} dzwt(i, j, k = 0) &= \frac{dswt(i, j, k = 0)}{dst_dz_t(i, j, k = 1)} \\ dzwt(i, j, k = 1, kmt - 1) &= \frac{2 dswt(i, j, k)}{dst_dz_t(i, j, k) + dst_dz_t(i, j, k + 1)} \\ dzwt(i, j, k = kmt) &= \frac{dswt(i, j, k = kmt)}{dst_dz_t(i, j, k = kmt)}. \end{aligned} \quad (10.28)$$

For the finite volume based computation of hydrostatic pressure (Section 10.1.2), the distance between tracer points is computed according to equations (10.23)-(10.25). Notice how the s -grid increments are constant in time, and are set by the z -grid increments at the initial model time step.

10.4.3 Terrain following $\sigma^{(z)}$ vertical coordinate

For the terrain following $\sigma^{(z)}$ coordinate, we proceed in a different manner than for the geopotential and z^* coordinates. Here, a dimensionless partition of the $\sigma^{(z)}$ coordinate is prescribed during initialization, and then the vertical grid dimensions deduced from knowledge of the depth field ht . The partitioning of $\sigma^{(z)}$ can be chosen in many ways. We choose to base this partition on the vertical grid dimensions $dz_t(k)$ and $dzw(k)$ available in the Grid derived type. These are the *full cell* grid dimensions, which thus make dst and $dswt$ independent of horizontal position (i, j) .

$$\begin{aligned} dz_t_dst(i, j, k) &= ht(i, j) + eta_t(i, j) \\ dst(i, j, k) &= dz_t(k)/zw(nk) \\ dswt(i, j, k) &= dzw(k)/zw(nk) \\ dz_t(i, j, k) &= dz_t_dst(i, j, k) * dst(i, j, k). \end{aligned} \quad (10.29)$$

For the energetically based computation of hydrostatic pressure (Section 10.1.1), the distance between tracer points is computed according to

$$\begin{aligned} dzwt(i, j, k = 0) &= \frac{dswt(i, j, k = 0)}{dst_dz_t(i, j, k = 1)} \\ dzwt(i, j, k = 1, kmt - 1) &= \frac{2 * dswt(i, j, k)}{dst_dz_t(i, j, k) + dst_dz_t(i, j, k + 1)} \\ dzwt(i, j, k = kmt) &= \frac{dswt(i, j, k = kmt)}{dst_dz_t(i, j, k = kmt)}. \end{aligned} \quad (10.30)$$

For the finite volume based computation of hydrostatic pressure (Section 10.1.2), the distance between tracer points is computed according to equations (10.23)-(10.25).

10.4.4 Non-terrain following pressure vertical coordinate

As described in Section 10.2, initialization of the non-Boussinesq model must take place in a manner different from the Boussinesq model. That is, specifying the vertical grid increments with pressure vertical

coordinates introduces a fundamentally new consideration. Namely, the vertical grid dimensions $dz t$ are a function of the initial *in situ* density ρ^{init} . However, with the present structure of MOM, we only know the initial density after an initial grid structure is established. Furthermore, MOM does not allow for vanishing layers. Hence, there is a possibility that a first guess at a vertical grid layout based on the bottom topography and the initial density, will not provide for a realizable grid in a pressure model absent vanishing layers. This point necessitates a multiple step process in the initialization of the pressure based model. We clarify these points in the following discussion.

The first step of the initialization takes the initial temperature and salinity fields, and initial grid specification file, all generated using the familiar MOM4 preprocessing code that assumes geopotential vertical coordinates. From this information, we compute a vertical density profile function

$$\rho_o(k) = \frac{\sum_{i,j} \text{dat}_{i,j} dz t^0(i,j,k) \rho^{\text{init}}}{\sum_{i,j} \text{dat}_{i,j} dz t^0(i,j,k)}. \quad (10.31)$$

Here, the initial density ρ^{init} is assumed to live on the initial grid specified by thicknesses $dz t^0(i,j,k)$ that are created just as if the model vertical coordinate were geopotential (including bottom partial cells). The model is run for a time step to allow for this function to be generated and written to a netCDF file. Then the model is rerun, now reading in this function as an input file for use in subsequent steps of the initialization.

Note that the vertical density profile function $\rho_o(k)$ takes account of the possibility for larger averaged density in the deep ocean, in which case the vertical pressure increments increase at depth even moreso than suggested by the generally larger vertical depth increments towards the deeper ocean. The utility of the density profile for specifying the pressure levels is a function of many model details. For example, in the global one degree model used for CM2.1 (Griffies et al. (2005), Gnanadesikan et al. (2006), Delworth et al. (2006), Wittenberg et al. (2006), and Stouffer et al. (2006a)) and ESM2M (Dunne et al. (2012)), using a reference profile proved to be detrimental to the abyssal flow in the tropics. We hypothesize that the profile produced a vertical grid spacing that was much coarser than otherwise provided with a depth basic vertical coordinate. Another possibility is there is a bug with the nontrivial $\rho_o(k)$ profile. Hence, we recommend the trivial choice

$$\rho_o(k) = \rho_o. \quad (10.32)$$

Other model configurations may find different profiles to be more useful.

We now proceed to generate the vertical grid increments dst . As the model is pressure-based, these increments should be a function only of the vertical grid index k , with the only exception being at the bottom where partial bottom steps allow for i, j dependence

$$dstlo(i,j,k) = -g \rho_o(k) dz tlo^0(i,j,k) \quad (10.33)$$

$$dstup(i,j,k) = -g \rho_o(k) dz t^0(i,j,k) \quad (10.34)$$

$$dst(i,j,k) = dstlo(k) + dstup(k), \quad (10.35)$$

where again

$$dz t^0(i,j,k) = dz tlo^0(i,j,k) + dz t^0(i,j,k) \quad (10.36)$$

are generated by assuming the model is a geopotential model so that the i, j dependence arises just from the bottom partial cell adjustments.

Now that we have the vertical pressure increments $dst(i,j,k)$, $dstlo(i,j,k)$, and $dstup(i,j,k)$, and the initial density ρ^{init} , we recompute the vertical depth increments so that

$$dz tlo^1(i,j,k) = - \left(\frac{dstlo(i,j,k)}{g \rho^{\text{init}}(i,j,k)} \right) \quad (10.37)$$

$$dz t^1(i,j,k) = - \left(\frac{dstup(i,j,k)}{g \rho^{\text{init}}(i,j,k)} \right) \quad (10.38)$$

$$dz t^1(i,j,k) = dz tlo^1(i,j,k) + dz t^1(i,j,k). \quad (10.39)$$

The fundamental question is whether the above procedure allows for the same number of vertical grid cells to exist in a column with the pressure coordinate model as for the analog geopotential model. A

general aim followed here is to include pressure coordinate models in MOM in a manner that represents an overall modest adjustment to what is commonly done for initializing geopotential models. Given this aim, we assume that both the geopotential model and pressure model have the same number of vertical grid cells within each column. That is, the $kmt(i, j)$ array computed for the geopotential model is the same as for the pressure model. This assumption is self-consistent with the same bottom topography array $ht(i, j)$ *only if*

$$\sum_{k=1}^{k=kmt(i,j)} dz t(i, j, k) \leq ht(i, j). \quad (10.40)$$

More stringently, we aim to allow for a nontrivial bottom cell thickness $dztmin$ in the pressure model in order to regularize the numerical calculations in this cell, so that

$$\sum_{k=1}^{k=kmt(i,j)-1} dz t(i, j, k) \leq ht(i, j) - dz tmin. \quad (10.41)$$

If this condition fails, then we are unable to initialize the pressure model with the same density distribution and bottom depths as in the geopotential model. There are two options: modify the density or modify the bottom. Although not commonly applied at GFDL, the option of modifying the bottom has been facilitated in MOM, with documentation given in subroutine `ocean_thickness_init_adjust` in the module `ocean_core/ocean_thickness.F90`. Depending on details of the initial density and $dztmin$, modifications of the bottom have been found to be modest, and mostly localized to shallow ocean shelf regions. There is no general rule, and the researcher may wish to iterate somewhat to refine the choice of bottom topography for use with the pressure model.

To appreciate the problem a bit more, we write the sum (10.41) in the form

$$\begin{aligned} \sum_{k=1}^{k=kmt(i,j)-1} dz t(i, j, k) &= - \sum_{k=1}^{k=kmt(i,j)-1} \frac{dst(i, j, k)}{g \rho^{init}(i, j, k)} \\ &= \sum_{k=1}^{k=kmt(i,j)-1} dz t^0(i, j, k) \frac{\rho_o(k)}{\rho^{init}(i, j, k)}. \end{aligned} \quad (10.42)$$

Thus, if we admit regions of the ocean where density is far less than the profile $\rho_o(k)$, then the vertical column will be relatively thick. Hence, in order to maintain the same number of vertical grid cells in the pressure and geopotential model, we are forced to depress the bottom topography by some nonzero amount.

Assuming the bottom topography is chosen according to one of the above conventions, we have the following means for computing the grid increments with the pressure vertical coordinate model. Here are the equations that summarize this step

$$\begin{aligned} dz t_dst(i, j, k) &= -(g * rho(i, j, k))^{-1} \\ dswt(i, j, k = 1, kmt - 1) &= -g * rho_o(k) * dzwt(i, j, k)|_{\tau=0} \\ dst(i, j, k = 2, kmt - 1) &= -g * rho_o(k) * dz t(i, j, k)|_{\tau=0} \\ dswt(i, j, k = 0) &= -st(i, j, k = 1) + patm(i, j) \\ dswt(i, j, k = kmt) &= st(i, j, k = kmt) - pbot(i, j) \\ dst(i, j, k = 1) &= -sw(i, j, k = 1) + patm(i, j) \\ dst(i, j, k = kmt) &= sw(i, j, k = kmt - 1) - pbot(i, j) \\ dz t(i, j, k) &= dz t_dst(i, j, k) * dst(i, j, k). \end{aligned} \quad (10.43)$$

For the energetically based computation of hydrostatic pressure (Section 10.1.1), the distance between

tracer points is computed according to

$$\begin{aligned} dzwt(i, j, k = 0) &= \frac{dswt(i, j, k = 0)}{dst_dz t(i, j, k = 1)} \\ dzwt(i, j, k = 1, kmt - 1) &= \frac{2 * dswt(i, j, k)}{dst_dz t(i, j, k) + dst_dz t(i, j, k + 1)} \\ dzwt(i, j, k = kmt) &= \frac{dswt(i, j, k = kmt)}{dst_dz t(i, j, k = kmt)}. \end{aligned} \quad (10.44)$$

For the finite volume based computation of hydrostatic pressure (Section 10.1.2), the distance between tracer points is computed according to equations (10.23)-(10.25).

10.4.5 p^* vertical coordinate

The same initialization procedure is followed for p^* as for pressure. Following the initialization, the model employs the following equations for setting the vertical grid increments

$$\begin{aligned} dz t_dst(i, j, k) &= - \left(\frac{pbot(i, j) - patm(i, j)}{g * rho(i, j, k) * pbot0(i, j)} \right) \\ dswt(i, j, k) &= dswt(i, j, k)|_{\tau=0} \\ dst(i, j, k) &= dst(i, j, k)|_{\tau=0} \\ dz t(i, j, k) &= dz t_dst(i, j, k) * dst(i, j, k) \end{aligned} \quad (10.45)$$

For the energetically based computation of hydrostatic pressure (Section 10.1.1), the distance between tracer points is computed according to

$$\begin{aligned} dzwt(i, j, k = 0) &= \frac{dswt(i, j, k = 0)}{dst_dz t(i, j, k = 1)} \\ dzwt(i, j, k = 1, kmt - 1) &= \frac{2 * dswt(i, j, k)}{dst_dz t(i, j, k) + dst_dz t(i, j, k + 1)} \\ dzwt(i, j, k = kmt) &= \frac{dswt(i, j, k = kmt)}{dst_dz t(i, j, k = kmt)}. \end{aligned} \quad (10.46)$$

For the finite volume based computation of hydrostatic pressure (Section 10.1.2), the distance between tracer points is computed according to equations (10.23)-(10.25).

10.4.6 Steps to initialize pressure and p^* based models

We now summarize the steps required to initialize the pressure and p^* based models.

- Determine $dz t^0(i, j, k)$ as z-model, with $\sum_{k=1}^{kmt(i, j)} dz t^0(i, j, k) = ht(i, j)$
- Determine the density profile function $rho_o(k)$ according to equation (10.31), with default $rho_o(k) = \rho_o$.
- Set the pressure increments according to

$$dstlo(i, j, k) = -g rho_o(k) dz tlo^0(i, j, k) \quad (10.47)$$

$$dstup(i, j, k) = -g rho_o(k) dz t up^0(i, j, k) \quad (10.48)$$

$$dst(i, j, k) = dstlo(k) + dstup(k), \quad (10.49)$$

- Insert the initial temperature and salinity to the grid points (i, j, k) to then determine the initial density $\rho^{init}(i, j, k)$.

- Determine the modified thickness of the grid cells according to

$$dztl^1(i, j, k) = -\frac{dstlo(i, j, k)}{g \rho^{init}(i, j, k)} \quad (10.50)$$

$$dzgup^1(i, j, k) = -\frac{dstup(i, j, k)}{g \rho^{init}(i, j, k)} \quad (10.51)$$

$$dztl^1(i, j, k) = dztl(i, j, k) + dzgup(i, j, k). \quad (10.52)$$

- Determine if

$$\sum_{k=1}^{k=km(i,j)-1} dztl(i, j, k) \leq ht(i, j) - dztmin. \quad (10.53)$$

If so, then make no modifications to the bottom topography. If not, then deepen the bottom topography so that the following equality is satisfied

$$ht(i, j)^{mod} = \sum_{k=1}^{k=km(i,j)-1} dztl(i, j, k) + dztmin. \quad (10.54)$$

- Determine the bottom cell thickness according to

$$dztl(i, j, kmt) = ht(i, j) - \sum_{k=1}^{k=km(i,j)-1} dztl(i, j, k). \quad (10.55)$$

10.4.7 Terrain following $\sigma^{(p)}$ coordinate

For the terrain following $\sigma^{(p)}$ coordinate, we use the same dimensionless partition as for the $\sigma^{(z)}$ coordinate to initialize the grid arrangement. However, we have been unable to derive a self-consistent method to incorporate the *in situ* density into the algorithm, since to compute the bottom pressure we must know $dztl$, but to know $dztl$ requires the bottom pressure. Hence, we expect there to be a large and spurious deviation in surface height just after initialization for runs with $\sigma^{(p)}$ coordinate.

During the integration, we make use of the following grid increments

$$\begin{aligned} dztl_dst(i, j, k) &= -\left(\frac{pbot(i, j) - patm(i, j)}{g * rho(i, j, k)} \right) \\ dswt(i, j, k) &= -dzw(k)/zw(nk) \\ dst(i, j, k) &= -dztl(k)/zw(nk) \\ dztl(i, j, k) &= dztl_dst(i, j, k) * dst(i, j, k) \end{aligned} \quad (10.56)$$

For the energetically based computation of hydrostatic pressure (Section 10.1.1), the distance between tracer points is computed according to

$$\begin{aligned} dzwt(i, j, k=0) &= \frac{dswt(i, j, k=0)}{dst_dztl(i, j, k=1)} \\ dzwt(i, j, k=1, kmt-1) &= \frac{2 * dswt(i, j, k)}{dst_dztl(i, j, k) + dst_dztl(i, j, k+1)} \\ dzwt(i, j, k=kmt) &= \frac{dswt(i, j, k=kmt)}{dst_dztl(i, j, k=kmt)}. \end{aligned} \quad (10.57)$$

For the finite volume based computation of hydrostatic pressure (Section 10.1.2), the distance between tracer points is computed according to equations (10.23)-(10.25).

10.5 Surface height and bottom pressure diagnosed

The purpose of this section is to detail how the surface height and bottom pressure are diagnosed in non-Boussinesq and Boussinesq models, respectively.

10.5.1 Surface height diagnosed in pressure based models

For models using a pressure based vertical coordinate, the surface height η is diagnosed, whereas for depth based models it is computed prognostically (Section 10.6). To diagnose the surface height, we use the identity

$$\eta = -H + \sum dz \quad (10.58)$$

given the thickness dz of each cell determined via Table 10.1. This is the original calculation provided in MOM for η , with the associated diagnostic table entry being

$$\text{eta.t} = -H + \sum dz. \quad (10.59)$$

Another diagnostic method, identical in the continuum but differing numerically due to finite precision, uses the following identity valid for the three pressure-based vertical coordinates supported in MOM

$$H + \eta = \left(\frac{p_b - p_a}{\rho_o g} \right) - \int_{-H}^{\eta} \left(\frac{\rho - \rho_o}{\rho_o} \right) dz. \quad (10.60)$$

This alternative calculation separates the smaller density contribution arising from density anomaly $\rho' = \rho - \rho_o$, from the larger bottom pressure contribution. This separation aims to facilitate a more precise calculation by reducing numerical roundoff. However, in practice there is very little difference from the original calculation in equation (10.58). The diagnostic table entry for the modified diagnostic is given by

$$\text{eta.t}_{\text{mod}} = \frac{p_b - p_a - \rho_o g H}{\rho_o g} - \sum \left(\frac{\rho - \rho_o}{\rho_o} \right) dz. \quad (10.61)$$

10.5.1.1 Concerning nonzero areal average

It is useful to note a common occurrence with pressure based models. Namely, the surface height will generally have a nonzero areal average even in the absence of mass fluxes. Such should be expected since the pressure based models conserve mass, not volume. For example, surface height can actually decrease even when mass is added to a column, so long as the column density increases by a sufficient amount. Hence, we are unable to make a general statement regarding the sign of the surface height without knowledge of both the mass per area in the column (as determined by the bottom pressure) as well as the vertical sum of the inverse density. Relatedly, the steric effect will cause the surface height to rise in regions of heating/freshening and decrease in regions of cooling/evaporation.

10.5.1.2 Concerning small scale features

For a non-Boussinesq pressure-based simulation, the sea level is diagnosed through either equation (10.58) or equation (10.60). The cell thickness, dz , appearing in these equations is a function of density and mass in a cell. The density and mass change according to the flow, the temperature and salinity, and the cell pressure. There are opportunities for relatively small scale features to appear in the diagnosed sea level through the imprint of small scale features in the density and mass fields. Furthermore, adding the thickness from small cells (e.g., thin bottom partial cells) to those from large cells offers an opportunity for truncation errors, especially when later subtracting the summed depth of a resting ocean to compute the sea level. For those familiar with the smoother sea level fields arising from a Boussinesq simulation, the application of a smoothing operator to the diagnosed eta is a suitable approach to removing some of the small scale features.

In summary, the prognosed sea level from a Boussinesq simulation is generally smoother than the diagnosed sea level in a non-Boussinesq pressure-based simulation. The physical reason is that the non-Boussinesq sea level is impacted by the density and mass that sits within a fluid column, whereas the sea level in a Boussinesq fluid is impacted by density only through the impacts on the convergence of depth integrated currents.

10.5.2 Bottom pressure diagnosed in depth based models

For models using a depth based vertical coordinate, it is necessary to diagnose the bottom pressure p_b using the following identity

$$p_b = p_a + g \sum \rho dz. \quad (10.62)$$

Here, we use the *in situ* density ρ and the thickness dz of each cell.

10.6 Vertically integrated volume/mass budgets

The vertically integrated mass and volume budgets determine, respectively, the bottom pressure and the surface height. The purpose of this section is to derive these budgets for use with depth based and pressure based vertical coordinates.

10.6.1 Vertically integrated volume budget

The budget for the volume per unit horizontal area for a Boussinesq fluid integrated over the depth of a grid cell takes the following forms, depending on whether the cell is in the interior, the bottom, or the surface

$$\partial_t(dz) = -\nabla_s \cdot (\mathbf{u} dz) - (w^{(z)})_{s=s_{k-1}} + (w^{(z)})_{s=s_k} + \mathcal{S}^{(V)} dz \quad (10.63)$$

$$\partial_t(dz) = -\nabla_s \cdot (\mathbf{u} dz) - (w^{(z)})_{s=s_{kbot-1}} + \mathcal{S}^{(V)} dz \quad (10.64)$$

$$\partial_t(dz) = -\nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{k=1}} + Q_m/\rho_o + \mathcal{S}^{(V)} dz \quad (10.65)$$

We obtained these equations from the mass budgets (2.155), (2.163), and (2.173), with density set to the constant Boussinesq reference value ρ_o , and with $\mathcal{S}^{(V)}$ a volume source (with units of inverse time). The vertical sum of these budgets leads to

$$\partial_t(H + \eta) = -\nabla \cdot \mathbf{U} + Q_m/\rho_o + \sum_k \mathcal{S}^{(V)} dz, \quad (10.66)$$

where we used

$$\sum_k dz = H + \eta, \quad (10.67)$$

which is the total thickness of the water column, and we introduced the depth integrated horizontal velocity

$$\sum_k \mathbf{u} dz = \mathbf{U}. \quad (10.68)$$

Since H is the time independent ocean bottom, equation (10.66) provides a prognostic relation for the surface height

$$\partial_t \eta = -\nabla \cdot \mathbf{U} + Q_m/\rho_o + \sum_k \mathcal{S}^{(V)} dz. \quad (10.69)$$

This is the free surface equation used for depth based vertical coordinate Boussinesq models.

10.6.2 Vertically integrated mass budget

The budget of the mass per unit horizontal area for a non-Boussinesq fluid integrated over the depth of a grid cell takes the following forms, depending on whether the cell is in the interior, the bottom, or the surface

$$\partial_t(\rho dz) = -\nabla_s \cdot (\mathbf{u} \rho dz) - (\rho w^{(z)})_{s=s_{k-1}} + (\rho w^{(z)})_{s=s_k} + \mathcal{S}^{(M)} \rho dz \quad (10.70)$$

$$\partial_t(dz \rho) = -\nabla_s \cdot (\mathbf{u} \rho dz) - (\rho w^{(z)})_{s=s_{kbot-1}} + \mathcal{S}^{(M)} \rho dz \quad (10.71)$$

$$\partial_t(dz \rho) = -\nabla_s \cdot (\mathbf{u} \rho dz) + (\rho w^{(z)})_{s=s_k} + Q_m + \mathcal{S}^{(M)} \rho dz. \quad (10.72)$$

These are equations (2.155), (2.163), and (2.173). The vertical sum of these budgets lead to the vertically integrated balance of mass per area for a column of fluid

$$\partial_t \left(\sum_k \rho dz \right) = -\nabla \cdot \left(\sum_k \mathbf{u} \rho dz \right) + Q_m + \sum_k \mathcal{S}^{(M)} \rho dz. \quad (10.73)$$

The vertical integral $\sum_k \rho dz$ is the total mass per area in the fluid column. In a hydrostatic fluid, this mass per area is equal to the difference in pressure between the bottom and top of the column

$$\sum_k \rho dz = g^{-1} (p_b - p_a). \quad (10.74)$$

Consequently, the mass budget generally takes the form

$$\begin{aligned} \partial_t(p_b - p_a) &= -g \nabla \cdot \left(\sum_k \mathbf{u} \rho dz \right) + g Q_m + g \sum_k \mathcal{S}^{(M)} \rho dz \\ &= -g \nabla \cdot \mathbf{U}^\rho + g Q_m + g \sum_k \mathcal{S}^{(M)} \rho dz \end{aligned} \quad (10.75)$$

where

$$\mathbf{U}^\rho = \sum_k \mathbf{u} \rho dz \quad (10.76)$$

is the vertically integrated density weighted horizontal velocity. Equivalently, it is the vertically integrated horizontal momentum per horizontal area. The time tendency for the applied pressure could be provided by another component model. Without this information, it can be approximated by, for example,

$$\partial_t p_a \approx \frac{p_a(t) - p_a(t-1)}{\Delta t}. \quad (10.77)$$

For the vertical integral of the horizontal momentum per volume, $\rho \mathbf{u}$, note that $z_s \rho$ is depth independent for either choice of pressure based coordinates given in Table 5.2. In summary, for the pressure based coordinates in Table 5.2, the depth integrated mass balance (10.73) takes the form

$$\partial_t(p_b - p_a) = -g \nabla \cdot \mathbf{U}^\rho + g Q_m + g \sum_k \mathcal{S}^{(M)} \rho dz. \quad (10.78)$$

10.6.3 Summary of the vertically integrated volume/mass budgets

In summary, the vertically integrated volume and mass budgets take on the isomorphic form

$$\begin{aligned} \partial_t \eta &= -\nabla \cdot \mathbf{U} + Q_m / \rho_o + \sum_k \mathcal{S}^{(V)} dz \\ g^{-1} \partial_t(p_b - p_a) &= -\nabla \cdot \mathbf{U}^\rho + Q_m + \sum_k \mathcal{S}^{(M)} \rho dz. \end{aligned} \quad (10.79)$$

These budgets provide prognostic relations for the surface height η in the Boussinesq case, and the bottom pressure p_b in the non-Boussinesq case. The tendency for the applied pressure p_a must be determined by another component model, or approximated via equation (10.77).

10.7 Compatibility between tracer and mass

Although we do not time step the vertically integrated tracer budget in an ocean model, it is useful to write it down for diagnostic purposes. Furthermore, it allows us to introduce a compatibility condition between tracer and mass budgets. To do so, recall the tracer budgets for the interior, bottom, and surface grid cells, given by equations (36.122), (2.162), and (2.172)

$$\begin{aligned}
 \partial_t(C \rho dz) &= \mathcal{S}^{(C)} \rho dz - \nabla_s \cdot [\rho dz (\mathbf{u} C + \mathbf{F})] \\
 &\quad - [\rho (w^{(z)} C + F^{(s)})]_{s=s_{k-1}} \\
 &\quad + [\rho (w^{(z)} C + F^{(s)})]_{s=s_k} \\
 \partial_t(C \rho dz) &= \mathcal{S}^{(C)} \rho dz - \nabla_s \cdot [\rho dz (\mathbf{u} C + \mathbf{F})] \\
 &\quad - [\rho (w^{(z)} C + F^{(s)})]_{s=s_{kbot-1}} \\
 &\quad + Q_{(bot)}^{(C)} \\
 \partial_t(C \rho dz) &= \mathcal{S}^{(C)} \rho dz - \nabla_s \cdot [\rho dz (\mathbf{u} C + \mathbf{F})] \\
 &\quad + [\rho (w^{(z)} C + F^{(s)})]_{s=s_{k=1}} \\
 &\quad + Q_m C_m - Q_{(C)}^{(turb)} .
 \end{aligned}$$

Summing these budgets over a vertical column leads to

$$\begin{aligned}
 \partial_t \left(\sum_k C \rho dz \right) &= \sum_k \mathcal{S}^{(C)} \rho dz - \nabla_s \cdot \left(\sum_k \rho dz (\mathbf{u} C + \mathbf{F}) \right) \\
 &\quad + \left(Q_m C_m - Q_{(C)}^{(turb)} + Q_{(C)}^{(bott)} \right).
 \end{aligned} \tag{10.80}$$

As expected, the only contributions from vertical fluxes come from the top and bottom boundaries. Furthermore, by setting the tracer concentration to a uniform constant, all the turbulent flux terms vanish, in which case the budget reduces to the vertically integrated mass budget discussed in Section 10.6.2. This *compatibility* between tracer and mass budgets must be carefully maintained by the discrete model equations.¹

10.8 Diagnosing the dia-surface velocity component

The key distinction between Eulerian vertical coordinates and Lagrangian vertical coordinates is how they treat the dia-surface velocity component

$$w^{(z)} = \frac{\partial z}{\partial s} \frac{ds}{dt}. \tag{10.81}$$

The Lagrangian models prescribe it whereas Eulerian models diagnose it. The purpose of this section is develop Eulerian algorithms for diagnosing the dia-surface velocity component for the depth based and pressure based vertical coordinates of Chapter 5. As we will see, a crucial element for the utility of these algorithms is that the specific thickness z_s is depth independent using depth based coordinates in a Boussinesq fluid, and ρz_s is depth independent using pressure based coordinates in a non-Boussinesq fluid.

¹As discussed by Griffies et al. (2001), local conservation of an algorithm for tracer and volume/mass can readily be checked by running a model with uniform tracer concentration and blowing winds across the ocean surface. Surface height undulations will ensue, thus causing changes in volume for the grid cells. But the tracer concentration should remain uniform in the absence of surface fluxes. Changes in tracer concentration will not occur if the volume/mass and tracer budgets are compatible in the sense defined in this section.

10.8.1 Depth based vertical coordinates

Rearrange the grid cell volume budgets (10.63)-(10.65) to express the dia-surface velocity component for the top cell, interior cells, and bottom cell as

$$(w^{(z)})_{s=s_{k=1}} = \partial_t(dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) - Q_m/\rho_o \quad (10.82)$$

$$(w^{(z)})_{s=s_k} = \partial_t(dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{k-1}} \quad (10.83)$$

$$0 = \partial_t(dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{kbot-1}}. \quad (10.84)$$

These equations are written from the surface to the bottom, with this order familiar from the z -coordinate version of MOM4.0. Equation (10.84) indicates that there is no transport through the ocean bottom. In a numerical model, this equation provides a useful diagnostic to check that dia-surface velocity components in the cells above the bottom have been diagnosed correctly. A nonzero result at the bottom signals a code bug.

We now detail how the dia-surface velocity component is diagnosed for the depth based vertical coordinates discussed in Section 5.1. To do so, we determine diagnostic relations for the time tendency $\partial_t(dz)$ of the grid cell thickness as a function of vertical coordinate. Because z_s is independent of depth for these coordinates, we are able to express $\partial_t(dz)$ as a function of $\partial_t\eta$, which in turn can be diagnosed using the vertically integrated volume budget.

10.8.1.1 Depth coordinate

For $s = z$, the only grid cell that admits a non-zero $\partial_t(dz)$ is the surface cell, where $\partial_t(dz) = \partial_t\eta$. Also, in MOM4.0 we assumed that there are no volume sources for $k > 1$. But this assumption is not fundamental. Indeed, volume sources throughout the column are not a problem, so long as their affects on volume conservation for the cell are properly handled in the diagnosis of the vertical velocity component. These results lead to the following expressions for the dia-surface velocity component $w^{(z)} = dz/dt = w$

$$(w^{(z)})_{z=z_{k=1}} = \partial_t\eta - \mathcal{S}^{(V)} dz + \nabla_z \cdot (\mathbf{u} dz) - Q_m/\rho_o \quad (10.85)$$

$$(w^{(z)})_{z=z_k} = -\mathcal{S}^{(V)} dz + \nabla_z \cdot (\mathbf{u} dz) + (w^{(z)})_{z=z_{k-1}} \quad (10.86)$$

$$0 = -\mathcal{S}^{(V)} dz + \nabla_z \cdot (\mathbf{u} dz) + (w^{(z)})_{z=z_{kbot-1}}. \quad (10.87)$$

The right hand side of the surface height equation (10.69) can be used to eliminate $\partial_t\eta$ in equation (10.85), thus leading to a purely diagnostic set of equations

$$(w^{(z)})_{z=z_{k=1}} = -\mathcal{S}^{(V)} dz + \nabla_z \cdot (\mathbf{u} dz) + \sum_k \mathcal{S}^{(V)} dz - \nabla \cdot \mathbf{U} \quad (10.88)$$

$$(w^{(z)})_{z=z_k} = -\mathcal{S}^{(V)} dz + \nabla_z \cdot (\mathbf{u} dz) + (w^{(z)})_{z=z_{k-1}} \quad (10.89)$$

$$0 = -\mathcal{S}^{(V)} dz + \nabla_z \cdot (\mathbf{u} dz) + (w^{(z)})_{z=z_{kbot-1}}. \quad (10.90)$$

The algorithm starts at $k = 1$ given knowledge of the right hand side terms in equation (10.88). Movement down the vertical column leads to the diagnosis of w for the full column.

10.8.1.2 Depth deviation coordinate

For $s = z - \eta$, the only grid cell that admits a non-zero $\partial_t(dz)$ is the bottom cell where $\partial_t(dz) = \partial_t\eta$. The dia-surface velocity component $w^{(z)} = w - d\eta/dt$ thus is diagnosed via

$$(w^{(z)})_{s=s_{k=1}} = -\mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) - Q_m/\rho_o \quad (10.91)$$

$$(w^{(z)})_{s=s_k} = -\mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{k-1}} \quad (10.92)$$

$$0 = \partial_t\eta - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{kbot-1}}. \quad (10.93)$$

As with the depth coordinate $s = z$, we use the surface height equation (10.69) to eliminate $\partial_t \eta$ in equation (10.93) and so lead to a fully diagnostic set of equations

$$(w^{(z)})_{s=s_{k=1}} = -\mathcal{S}^{(V)} dz - Q_m/\rho_o + \nabla_z \cdot (\mathbf{u} dz) \quad (10.94)$$

$$(w^{(z)})_{z=z_k} = -\mathcal{S}^{(V)} dz + (w^{(z)})_{z=z_{k-1}} + \nabla_z \cdot (\mathbf{u} dz) \quad (10.95)$$

$$0 = -\mathcal{S}^{(V)} dz + (w^{(z)})_{z=z_{kbot-1}} + \nabla_z \cdot (\mathbf{u} dz) + \left(Q_m/\rho_o - \nabla \cdot \mathbf{U} + \sum_k \mathcal{S}^{(V)} dz \right). \quad (10.96)$$

10.8.1.3 Zstar coordinate

For $s = z^* = H(z - \eta)/(H + \eta)$, all grid cells have time independent ds since the range for z^* is time independent. However, the specific thickness $z_{,s} = 1 + \eta/H$ is time dependent. The dia-surface velocity component is thus diagnosed via the equations

$$(w^{(z)})_{s=s_{k=1}} = ds H^{-1} \partial_t \eta - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) - Q_m/\rho_o \quad (10.97)$$

$$(w^{(z)})_{s=s_k} = ds H^{-1} \partial_t \eta - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{k-1}} \quad (10.98)$$

$$0 = ds H^{-1} \partial_t \eta - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{kbot-1}}. \quad (10.99)$$

The surface height equation (10.69) is used to eliminate $\partial_t \eta$ from each of these equations. Note that in verifying the correctness of these results, recall that $\sum_k ds = H$ for $s = z^*$.

10.8.1.4 Depth-sigma coordinate

For $s = \sigma^{(z)} = (z - \eta)/(H + \eta)$, all grid cells have constant ds since the range for σ is time independent. However, it has a time dependent specific thickness $z_{,s} = H + \eta$. These results lead to the following expressions for the dia-surface velocity component

$$(w^{(z)})_{s=s_{k=1}} = ds \partial_t \eta - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) - Q_m/\rho_o \quad (10.100)$$

$$(w^{(z)})_{s=s_k} = ds \partial_t \eta - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{k-1}} \quad (10.101)$$

$$0 = ds \partial_t \eta - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{kbot-1}}. \quad (10.102)$$

The surface height equation (10.69) is used to eliminate $\partial_t \eta$ from each of these equations. In verifying the correctness of these results, recall that $\sum_k ds = 1$ for $s = \sigma^{(z)}$.

10.8.1.5 General expression for dia-surface velocity component

In summary, for depth based vertical coordinates, the dia-surface velocity component is diagnosed via

$$(w^{(z)})_{s=s_{k=1}} = \partial_t (dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) - Q_m/\rho_o$$

$$(w^{(z)})_{s=s_k} = \partial_t (dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{k-1}} \quad (10.103)$$

$$0 = \partial_t (dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(z)})_{s=s_{kbot-1}}$$

where the thickness of a grid cell evolves according to

$\partial_t (dz) = \delta_{k,1} \partial_t \eta$	$s = z$
$\partial_t (dz) = \delta_{k,kbot} \partial_t \eta$	$s = z - \eta$
$\partial_t (dz) = ds (\partial_t \eta / H)$	$s = H(z - \eta)/(H + \eta)$
$\partial_t (dz) = ds \partial_t \eta$	$s = (z - \eta)/(H + \eta)$.

(10.104)

The surface height evolution

$$\partial_t \eta = -\nabla \cdot \mathbf{U} + Q_m / \rho_o + \sum_k \mathcal{S}^{(v)} dz \quad (10.105)$$

embodies volume conservation for a Boussinesq fluid column. The right hand side of (10.105) is used in equations (10.104) to produce a purely diagnostic expression for the dia-surface velocity components.

10.8.2 Pressure based vertical coordinates

We now diagnose the dia-surface velocity component for pressure based vertical coordinates. For this purpose, rearrange the grid cell mass budgets (10.70)-(10.72) to express the vertical velocity component as

$$(\rho w^{(z)})_{s=s_{k=1}} = \partial_t (\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) - Q_m \quad (10.106)$$

$$(\rho w^{(z)})_{s=s_k} = \partial_t (\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) + (\rho w^{(z)})_{s=s_{k-1}} \quad (10.107)$$

$$0 = \partial_t (\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) + (\rho w^{(z)})_{s=s_{kbot-1}}. \quad (10.108)$$

As for the depth based vertical coordinates, we write these equations from the surface to the bottom. Equation (10.108) indicates that there is no transport through the ocean bottom. In a numerical model, this equation provides a useful diagnostic to check that velocity components in the cells above the bottom have been diagnosed correctly. A nonzero result at the bottom signals a code bug.

We proceed as for depth based vertical coordinates by determining diagnostic relations for $\partial_t (\rho dz)$ as a function of the pressure based vertical coordinates discussed in Section 5.2. Because $\rho z_{,s}$ is independent of depth for these coordinates, we are able to express $\partial_t (\rho dz)$ as a function of $\partial_t p_a$ and $\partial_t p_b$. The time tendency of the applied pressure is set according to other component models, or approximated as (10.77). The time tendency for the bottom pressure is set according to the vertically integrated mass budget (10.78). Finally, we note that it is the density weighted dia-surface velocity component $\rho w^{(z)}$ which is most naturally diagnosed in this approach. Conveniently, it is $\rho w^{(z)}$ that is required for the non-Boussinesq tracer and momentum budgets discussed in Sections 2.6 and 2.9.

10.8.2.1 Pressure coordinate

For $s = p$, the density weighted specific thickness is a constant for all grid cells

$$\rho z_{,s} = -g^{-1}, \quad (10.109)$$

but both the surface and bottom grid cells admit a non-zero $\partial_t (\rho dz)$. At the surface²,

$$\begin{aligned} \rho dz &= -g^{-1} dp \\ &= -g^{-1} (p_a - p_{\text{bottom of cell } k=1}) \end{aligned} \quad (10.110)$$

which then leads to

$$\partial_t (\rho dz) = -g^{-1} \partial_t p_a. \quad (10.111)$$

That is, the top cell mass per area decreases when the applied pressure increases. This result follows since the bottom face of the top cell has a fixed pressure, but the top face is at the applied pressure p_a . As noted in Section 5.2, if the applied pressure becomes greater than $p_{\text{bottom of cell } k=1}$, then the top cell vanishes. For the bottom cell,

$$\begin{aligned} \rho dz &= -g^{-1} dp \\ &= -g^{-1} (p_{\text{top of cell } k=kbot} - p_b), \end{aligned} \quad (10.112)$$

²Recall that our convention in equation (5.34) is that $ds < 0$ for pressure based vertical coordinates. At the surface with pressure coordinates $s = p$, the coordinate increment is $dp = p_a - p_{\text{bottom of cell } k=1}$. This increment is negative since the applied pressure is less than the pressure at the bottom interface to cell $k = 1$. For the bottom cell, $dp = p_{\text{top of cell } k=kbot} - p_b$, which is negative when the bottom pressure is greater than the pressure just above it.

and so

$$\partial_t (\rho dz) = g^{-1} \partial_t p_b. \quad (10.113)$$

Hence, the bottom cell thickness increases as the bottom pressure increases. If the bottom pressure decreases below $p_{\text{top of cell } k=k_{bot}}$, then the bottom cell vanishes. These results lead to the following expressions for the density weighted dia-surface velocity component

$$(\rho w^{(z)})_{s=s_{k=1}} = \partial_t p_a - ds \mathcal{S}^{(M)} + \nabla_s \cdot (ds \mathbf{u}) + g Q_m \quad (10.114)$$

$$(\rho w^{(z)})_{s=s_k} = -ds \mathcal{S}^{(M)} + \nabla_s \cdot (ds \mathbf{u}) + (\rho w^{(z)})_{s=s_{k-1}} \quad (10.115)$$

$$0 = -\partial_t p_b - ds \mathcal{S}^{(M)} + \nabla_s \cdot (dz \mathbf{u}) + (\rho w^{(z)})_{s=s_{k_{bot}-1}}. \quad (10.116)$$

As a check, a sum of these equations leads to the vertically integrated mass budget (10.78) written in pressure coordinates. These equations are converted to diagnostic expressions for the dia-surface velocity component by substituting the known time tendencies for the applied pressure $\partial_t p_a$ (e.g., equation (10.77)) and the bottom pressure $\partial_t p_b$ via the column integrated mass budget (10.78).

10.8.2.2 Pressure deviation coordinate

For $s = p - p_a$, the only grid cell that admits a non-zero $\partial_t (\rho dz)$ is the bottom cell. At this cell,

$$\begin{aligned} \rho dz &= -g^{-1} dp \\ &= -g^{-1} [p_{\text{top of cell } k=k_{bot}} - (p_b - p_a)], \end{aligned} \quad (10.117)$$

and so

$$\partial_t (\rho dz) = g^{-1} \partial_t (p_b - p_a). \quad (10.118)$$

The right hand side can be diagnosed via the column integrated mass budget (10.78). These results lead to the following expressions for the dia-surface velocity component

$$(\rho w^{(z)})_{s=s_{k=1}} = -ds \mathcal{S}^{(M)} + \nabla_s \cdot (ds \mathbf{u}) + g Q_m \quad (10.119)$$

$$(\rho w^{(z)})_{s=s_k} = -ds \mathcal{S}^{(M)} + \nabla_s \cdot (ds \mathbf{u}) + (\rho w^{(z)})_{s=s_{k-1}} \quad (10.120)$$

$$0 = -\partial_t (p_b - p_a) - dz \mathcal{S}^{(M)} + \nabla_s \cdot (ds \mathbf{u}) + (\rho w^{(z)})_{s=s_{k_{bot}-1}}. \quad (10.121)$$

As a check, the sum of these equations recovers the vertically integrated mass budget (10.78) written in pressure coordinates.

10.8.2.3 Pstar coordinate

For $s = p^*$ with

$$p^* = p_b^o (p - p_a) / (p_b - p_a), \quad (10.122)$$

all grid cells have time independent constant ds . We are then led to the following mass per horizontal volume of a grid cell

$$\begin{aligned} \rho dz &= \rho z_s ds \\ &= -(g p_b^o)^{-1} (p_b - p_a) ds. \end{aligned} \quad (10.123)$$

The time tendency

$$\partial_t (\rho dz) = -ds (g p_b^o)^{-1} \partial_t (p_b - p_a) \quad (10.124)$$

can be diagnosed via the column integrated mass budget (10.78). We then use these results in the general expressions (10.106)-(10.108) to generate the algorithm for diagnosing the vertical velocity components. As a check, the sum of these equations recovers the vertically integrated mass budget (10.78) written in

pressure coordinates. Note that in verifying this identity, it is important to note that $\sum_k ds = -p_b^o$ for the pstar coordinate, which results from the following identities

$$\begin{aligned} p_b - p_a &= g \sum_k \rho dz \\ &= g \sum_k \rho z_{,s} ds \\ &= -\left(\frac{p_b - p_a}{p_b^o}\right) \sum_k ds, \end{aligned} \quad (10.125)$$

where we used the hydrstatic balance (10.74) for the first equality.

10.8.2.4 Pressure sigma coordinate

For $s = \sigma^{(p)} = (p - p_a)/(p_b - p_a)$, all grid cells have time independent ds since the range for σ is time independent. However, this coordinate has a time dependent density weighted specific thickness, thus leading to

$$\begin{aligned} \rho dz &= \rho z_{,s} ds \\ &= -g^{-1} (p_b - p_a) ds. \end{aligned} \quad (10.126)$$

We use these results in the general expressions (10.106)-(10.108) to generate the algorithm for diagnosing the vertical velocity components. As a check, the sum of these equations recovers the vertically integrated mass budget (10.78) written in pressure coordinates. In verifying this identity, it is important to note that $\sum_k ds = -1$ for $s = \sigma^{(p)}$.

10.8.2.5 General expression for the dia-surface velocity component

In summary, for pressure based vertical coordinates, the dia-surface velocity component is diagnosed via

$$\begin{aligned} (\rho w^{(z)})_{s=s_{k=1}} &= \partial_t (\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) - Q_m \\ (\rho w^{(z)})_{s=s_k} &= \partial_t (\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) + (\rho w^{(z)})_{s=s_{k-1}} \\ 0 &= \partial_t (\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) + (\rho w^{(z)})_{s=s_{kbot-1}}. \end{aligned} \quad (10.127)$$

where the density weighted thickness of a grid cell evolves according to

$g \partial_t (\rho dz) = -\delta_{k,1} \partial_t p_a + \delta_{k,kbot} \partial_t p_b$	$s = p$
$g \partial_t (\rho dz) = \delta_{k,kbot} \partial_t (p_b - p_a)$	$s = p - p_a$
$g \partial_t (\rho dz) = -(ds/p_b^o) \partial_t (p_b - p_a)$	$s = p_b^o (p - p_a)/(p_b - p_a)$
$g \partial_t (\rho dz) = -ds \partial_t (p_b - p_a)$	$s = (p - p_a)/(p_b - p_a)$

(10.128)

and the bottom pressure evolution

$$\partial_t (p_b - p_a) = -g \nabla \cdot \mathbf{U}^p + g Q_m + g \sum_k \mathcal{S}^{(M)} \rho dz \quad (10.129)$$

embodies mass conservation for a non-Boussinesq fluid column.

10.8.3 Comments about diagnosing the dia-surface velocity component

We emphasize again that a critical element in the Eulerian algorithms for diagnosing the vertical velocity components is the ability to exploit the depth independence of the specific thickness $z_{,s}$ for the depth based coordinates for a Boussinesq fluid, and the density weighted specific thickness $\rho z_{,s}$ for the pressure

based coordinates for a non-Boussinesq fluid. These properties allow us to remove the time tendencies for surface height and pressure from the respective diagnostic relations by substituting the depth integrated budgets (10.69) for the depth based models, and (10.78) for the pressure based models. Absent the depth independence, one would be forced to consider another approach, such as the time extrapolation approach to approximate the time tendency proposed by Greatbatch et al. (2001) and McDougall et al. (2002) for implementing a non-Boussinesq algorithm within a Boussinesq model.

10.9 Vertically integrated horizontal momentum

We now outline the split between the fast vertically integrated dynamics from the slower depth dependent dynamics. This split forms the basis for the split-explicit method used in MOM to time step the momentum equation. For this purpose, we formulate the budget for the vertically integrated momentum budget.

10.9.1 Budget using contact pressures on cell boundaries

Before proceeding with a formulation directly relevant for MOM, we note the form of the vertically integrated budget arising when we consider pressure acting on a cell as arising from the accumulation of contact stresses. For this purpose, we vertically sum the momentum budgets given by equations (2.226), (2.229) and (2.234), which leads to

$$\begin{aligned}
 (\partial_t + f \hat{\mathbf{z}} \wedge) \sum (\mathbf{u} dz \rho) = & - \sum \left(\hat{\mathbf{z}} \wedge (\mathcal{M} \mathbf{u} \rho dz) + \nabla_s \cdot [\mathbf{u} (\mathbf{u} \rho dz)] \right) \\
 & + \sum \left(-\nabla_s (p dz) + \mathbf{F} \rho dz \right) \\
 & + [p_a \nabla \eta + \boldsymbol{\tau}^{\text{wind}} + \rho_w Q_m \mathbf{u}_m] \\
 & + [p_b \nabla H - \boldsymbol{\tau}^{\text{bottom}}].
 \end{aligned} \tag{10.130}$$

Contact pressures on the top and bottom of the grid cells cancel throughout the column, just as other vertical fluxes from momentum and friction. The remaining contact pressures are from the bottom and top of the ocean column and the vertically integrated contact pressures on the sides of the fluid column. Correspondingly, if we integrate over the horizontal extent of the ocean domain, we are left only with contact pressures acting on the solid boundaries and undulating free surface. Such is to be expected, since the full ocean domain experiences a pressure force only from its contact with other components of the earth climate system.

10.9.2 Budget using the pressure gradient body force

As discussed in Section 2.8.2, we prefer to formulate the contribution of pressure to the linear momentum balance as a body force, whereby we exploit the hydrostatic balance. Hence, to develop the vertically integrated horizontal momentum budget, we start from the form of the budget given by equations (4.19), (4.20), and (4.21), rewritten here for the interior, bottom, and surface grid cells

$$\begin{aligned}
 [\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge] (\mathbf{u} \rho dz) = & \rho dz \mathcal{S}^{(u)} - \nabla_s \cdot [\mathbf{u} (\mathbf{u} \rho dz)] \\
 & - dz (\nabla_s p + \rho \nabla_s \Phi) + \mathbf{F} \rho dz \\
 & - [\rho (w^{(z)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_{k-1}} \\
 & + [\rho (w^{(z)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_k}
 \end{aligned} \tag{10.131}$$

$$\begin{aligned}
 [\partial_t + (f + \mathcal{M}) \hat{\mathbf{z}} \wedge] (\mathbf{u} \rho dz) = & \mathcal{S}^{(u)} \rho dz - \nabla_s \cdot [\mathbf{u} (\mathbf{u} \rho dz)] \\
 & - dz (\nabla_s p + \rho \nabla_s \Phi) + \mathbf{F} \rho dz \\
 & - [\rho (w^{(z)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_{kbot-1}} \\
 & - \boldsymbol{\tau}^{\text{bottom}}
 \end{aligned} \tag{10.132}$$

$$\begin{aligned}
[\partial_t + (f + \mathcal{M})\hat{\mathbf{z}} \wedge](\mathbf{u} \rho \, dz) &= \mathcal{S}^{(\mathbf{u})} \rho \, dz - \nabla_s \cdot [\mathbf{u}(\mathbf{u} \rho \, dz)] \\
&\quad - dz(\nabla_s p + \rho \nabla_s \Phi) + \mathbf{F} \rho \, dz \\
&\quad + [\boldsymbol{\tau}^{\text{wind}} + Q_m \mathbf{u}_m] \\
&\quad + [\rho(w^{(z)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_{k=1}}.
\end{aligned} \tag{10.133}$$

A vertical sum of the momentum budgets leads to

$$\begin{aligned}
(\partial_t + f \hat{\mathbf{z}} \wedge) \sum (\mathbf{u} \rho \, dz) &= \sum \mathcal{S}^{(\mathbf{u})} \rho \, dz \\
&\quad - \sum \left(\hat{\mathbf{z}} \wedge (\mathcal{M} \mathbf{u} \rho \, dz) + \nabla_s \cdot [\mathbf{u}(\mathbf{u} \rho \, dz)] \right) \\
&\quad + \sum dz \left(-\nabla_s p - \rho \nabla_s \Phi + \rho \mathbf{F} \right) \\
&\quad + \boldsymbol{\tau}^{\text{wind}} - \boldsymbol{\tau}^{\text{bottom}} + Q_m \mathbf{u}_m.
\end{aligned} \tag{10.134}$$

Fluctuations in the surface height contribute both to fluctuations in the horizontal pressure gradient and the geopotential gradient. These fluctuations lead to fast barotropic or external gravity waves, and so they must be integrated with a small time step. In contrast, the slower baroclinic or internal motions can be integrated with a larger time step, upwards of 100 times longer depending on details of the motions. Hence, it is advantageous for ocean climate modeling to develop an algorithm that splits between the motions when time stepping the equations. Details of this split depend on whether we work with a depth based or pressure based vertical coordinate.

10.9.3 Depth based vertical coordinates

We follow the discussion in Section 3.2.1 where the pressure gradient is split according to either equation (3.18) for $s = z^*$ or $s = \sigma^{(z)}$, and equation (3.17) for $s = z$. For geopotential coordinates $s = z$ this split takes the form

$$\nabla_s p + \rho \nabla_s \Phi = \underbrace{\nabla(p_a + p_{\text{surf}})}_{\text{fast}} + \underbrace{\nabla_s p'_{\text{clinic}} + \rho' \nabla_s \Phi}_{\text{slow}} \tag{10.135}$$

where $p_{\text{surf}} = \rho(z=0)g\eta$, $\rho = \rho_o + \rho'$ and $p'_{\text{clinic}} = g \int_z^0 \rho' \, dz$. For z -star or sigma coordinates, this split takes the form

$$\nabla_s p + \rho \nabla_s \Phi = \underbrace{\nabla(p_a + \rho_o g \eta)}_{\text{fast}} + \underbrace{\nabla_s p' + \rho' \nabla_s \Phi}_{\text{slow}} \tag{10.136}$$

where $p' = g \int_z^\eta \rho' \, dz$ is the anomalous pressure field. The Boussinesq form of the vertically integrated momentum budget (10.134) thus takes the form

$$\rho_o (\partial_t + f \hat{\mathbf{z}} \wedge) \sum (\mathbf{u} \, dz) = \mathbf{G} - (H + \eta) \nabla(p_a + p_{\text{surf}}) \tag{10.137}$$

for $s = z$ coordinates, and similarly for $s = z^*$ and $s = \sigma^{(z)}$ coordinates. In either case, \mathbf{G} is the vertical integral of the depth dependent terms on the right hand side of equation (10.134). \mathbf{G} embodies all contributions that are generally evolving on the slower baroclinic time scale. This equation, along with the vertically integrated volume budget discussed in Section 10.6, form the barotropic system for the Boussinesq fluid in MOM. These equations are time stepped to resolve the fast waves using a predictor-corrector or leap-frog scheme discussed in Chapter 12 of Griffies (2004) (see also Section 11.2), where \mathbf{G} is held fixed over the course of the barotropic cycle. Note that the predictor-corrector is preferred due to its enhanced dissipation of small spatial scale features, which are of some concern on the B-grid due to the gravity wave null mode (Killworth et al., 1991; Griffies et al., 2001).

10.9.4 Pressure based vertical coordinates

We now follow the discussion in Section 3.2.3 where the pressure gradient is split according to equation (3.27) into a slow three dimensional term and fast two dimensional term

$$\nabla_s p + \rho \nabla_s \Phi = \underbrace{\rho \nabla_s \Phi' - (\rho'/\rho_o) \nabla_s p}_{\text{slow}} + \underbrace{(\rho/\rho_o) \nabla (p_b + \rho_o \Phi_b)}_{\text{fast}}. \quad (10.138)$$

where

$$\Phi' = -(g/\rho_o) \int_{-H}^z \rho' dz. \quad (10.139)$$

The vertically integrated pressure gradient can be written

$$\begin{aligned} \sum dz (\nabla_s p + \rho \nabla_s \Phi) &= \sum dz [\rho \nabla_s \Phi' - (\rho'/\rho_o) \nabla_s p] \\ &\quad + \nabla (p_b + \rho_o \Phi_b) \sum (\rho/\rho_o) dz \\ &= \sum dz [\rho \nabla_s \Phi' - (\rho'/\rho_o) \nabla_s p] \\ &\quad + (g \rho_o)^{-1} (p_b - p_a) \nabla (p_b + \rho_o \Phi_b), \end{aligned} \quad (10.140)$$

where we used the hydrostatic balance to write

$$g \sum \rho dz = p_b - p_a \quad (10.141)$$

The vertically integrated momentum budget (10.134) thus takes the form

$$(\partial_t + f \hat{\mathbf{z}} \wedge) \sum (\mathbf{u} \rho dz) = \mathbf{G} - (g \rho_o)^{-1} (p_b - p_a) \nabla (p_b + \rho_o \Phi_b), \quad (10.142)$$

where \mathbf{G} is the vertical integral of the depth dependent terms on the right hand side of equation (10.134), including the slow contribution to the pressure gradient force. The time stepping of equation (10.145) then proceeds as for the Boussinesq case discussed in Section 11.2. To help reduce errors in the calculation of the pressure gradient, it is useful to consider the following split of the bottom pressure

$$p_b = p'_b + \rho_o g H, \quad (10.143)$$

so that the vertically integrated mass and momentum budgets take the form

$$\partial_t (p'_b - p_a) = -g \nabla \cdot \mathbf{U}^p + g Q_m + g \sum_k \mathcal{S}^{(M)} \rho dz \quad (10.144)$$

$$(\partial_t + f \hat{\mathbf{z}} \wedge) \mathbf{U}^p = \mathbf{G} - (g \rho_o)^{-1} (p_b - p_a) \nabla p'_b. \quad (10.145)$$

The advantage of this formulation is that we remove the time independent bottom geopotential $\rho_o g H$ from the pressure gradient contribution to the vertically integrated velocity. As this contribution is huge, its removal enhances the numerical accuracy of the resulting pressure gradient.

TIME STEPPING SCHEMES

Contents

11.1	Split between fast and slow motions	160
11.2	Time stepping the model equations as in MOM4.0	160
11.2.1	The MOM4.0 scheme used in OM3.0	161
11.2.2	Problems related to tracer conservation	162
11.2.3	The time staggered scheme used in OM3.1	163
11.2.4	Sensitivity to the time stepping scheme	164
11.2.5	Dissipative aspects of the predictor-corrector	164
11.3	Introduction to time stepping in MOM	166
11.4	Basics of staggered time stepping in Boussinesq MOM	167
11.5	Predictor-corrector for the barotropic system	167
11.6	The Griffies (2004) scheme	169
11.7	Algorithms motivated from predictor-corrector	169
11.7.1	Barotropic time stepping and surface height on integer time steps	170
11.7.2	Surface height on half-integer time steps	170
11.7.3	Method A: $\mathbf{U}(\tau + \Delta\tau) = \bar{\mathbf{U}}$	172
11.7.3.1	Compatible tracer concentration	172
11.7.3.2	Why this scheme is not closed	173
11.7.4	Method B: $\mathbf{U}(\tau + \Delta\tau/2) = \bar{\mathbf{U}}$	173
11.7.4.1	Compatible tracer concentration	173
11.7.4.2	Why this scheme is not closed	174
11.8	Algorithms enforcing compatibility	174
11.8.1	Method I: Griffies (2004)	174
11.8.2	Method II: Algorithm based on barotropic predictor-corrector	175
11.8.3	Method III: Modified Griffies (2004)	175

The purpose of this chapter is to detail various issues of time stepping the discrete equations of MOM. It is written in two main parts, with the first part focusing on details of the scheme inherited from MOM4.0, and successfully used for climate modelling. The second part revisits the MOM4.0 scheme, and proposes some alternatives that are presently under investigation. The motivation for revisiting the MOM4.0 schemes is that they show problems when used with radiating open boundary conditions. Martin Schmidt led the studies into these alternative time stepping schemes, with some details shared with the more substantial methods studied by Shchepetkin and McWilliams (2005).

The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_barotropic.F90
ocean_core/ocean_velocity.F90
ocean_tracers/ocean_tracer.F90
```

11.1 Split between fast and slow motions

An algorithm of practical utility for climate modeling must split the fast and slow dynamics so that the slow dynamics can be updated with a much longer time step than the fast dynamics. These algorithms are known as split-explicit methods. Alternatives exist whereby the fast dynamics are time stepped implicitly and so may use the same time step as the slow dynamics. We prefer split-explicit methods since they are more efficient on parallel computers and arguably more straightforward (Griffies et al., 2001).

For a hydrostatic fluid, the fast motions can be approximated by the vertically integrated dynamics of Section 10.9 and the vertically integrated mass or volume budgets of Section 10.6. The remainder constitutes an approximation to the slower dynamics. Motions constituting the fast dynamics are embodied by the *barotropic* or *external* mode, and the slower motions are embodied by advection as well as the *baroclinic* or *internal* mode. Given the fundamental nature of the mass conserving non-Boussinesq flow, we formulate the split between the fast and slow modes using density weighting. For the Boussinesq flow, the density weighting reduces to an extra ρ_0 factor that trivially cancels.

Following the discussion in Section 12.3.5 of Griffies (2004), we consider the following split of the horizontal velocity field

$$\mathbf{u} = \underbrace{\left(\mathbf{u} - \frac{\sum_k \mathbf{u} \rho dz}{\sum_k \rho dz} \right)}_{\text{slow}} + \underbrace{\left(\frac{\sum_k \mathbf{u} \rho dz}{\sum_k \rho dz} \right)}_{\text{fast}} \quad (11.1)$$

$$\equiv \hat{\mathbf{u}} + \bar{\mathbf{u}}^z.$$

The fast barotropic velocity

$$\bar{\mathbf{u}}^z = \frac{\mathbf{U} \rho}{\sum_k \rho dz} \quad (11.2)$$

is updated according to the vertically integrated momentum equation of Section 10.9. The slow baroclinic velocity $\hat{\mathbf{u}}$ has zero density weighted vertical sum, and so its update is independent of any depth independent forcing, such as fast fluctuations in the surface height associated with external gravity waves. Therefore, we choose to update the slow dynamics using all pieces of the momentum equation forcing, except contributions from the rapid pressure and geopotential fluctuations. This update produces a velocity \mathbf{u}' that is related to the baroclinic velocity via

$$\hat{\mathbf{u}} = \mathbf{u}' - \frac{\sum_k \mathbf{u}' \rho dz}{\sum_k \rho dz}. \quad (11.3)$$

A similar relation was discussed in Section 12.4.2 of Griffies (2004). For global climate simulations, the time step available for the update of the slow dynamics is much larger (50 to 100 times larger) than the fast dynamics. It is this large time split, and the attendant improved model efficiency, that motivate the added complication arising from splitting the modes. Completing the updates of \mathbf{u}' and $\mathbf{U} \rho$ allows for an update of the full horizontal velocity via

$$\mathbf{u} = \left(\mathbf{u}' - \frac{\sum_k \mathbf{u}' \rho dz}{\sum_k \rho dz} \right) + \frac{\mathbf{U} \rho}{\sum_k \rho dz}. \quad (11.4)$$

11.2 Time stepping the model equations as in MOM4.0

We present here some details of the time stepping schemes available in MOM. Much of this section is taken from the paper Griffies et al. (2005) that documents two ocean climate models developed at GFDL; the

OM3.0 and OM3.1 models. Time stepping in OM3.0 is based on the standard MOM approach originating from the work of Bryan (1969), and detailed for an explicit free surface by Killworth et al. (1991) and Griffies et al. (2001). An alternative was developed for OM3.1.

The main motivation for developing an alternative was to address tracer non-conservation associated with time filtering used to suppress the leap frog computational mode appearing in the older method. The proposed *time staggered* method has much in common with that used by Hallberg (1997) for his isopycnal model, as well as by Marshall et al. (1997) and Campin et al. (2004) for their hydrostatic and non-hydrostatic z-coordinate models.

The purpose of this section is to detail features of the time stepping schemes employed in OM3.0 and OM3.1. Further details are provided in Chapter 12 of Griffies (2004). We also refer the reader to the pedagogical treatments of time stepping given by Mesinger and Arakawa (1976), Haltiner and Williams (1980), and Durran (1999). For simplicity, we focus here on the Boussinesq system assuming z-coordinates for the vertical. The more general case of arbitrary vertical coordinates with Boussinesq or non-Boussinesq equations follows trivially from the discussions here. Additionally, the original implementation of these ideas was based on the B-grid spatial discretization of MOM (Chapter 9). We include discussion here of modifications required for the C-grid available in MOM.

11.2.1 The MOM4.0 scheme used in OM3.0

We start by describing the standard approach used in MOM4.0 for time stepping tracers and baroclinic velocity. For the thickness weighted tracer equation, this update takes the form

$$\frac{(hT)^{\tau+1} - (\bar{h}\bar{T})^{\tau-1}}{2\Delta\tau_{\text{leap}}} = -\nabla_z \cdot [(h\mathbf{u})^\tau T^{\tau,\tau-1} + h^\tau \mathbf{F}^{\tau-1}] - \delta_k [w^\tau T^{\tau,\tau-1} + F_z^{\tau+1}]. \quad (11.5)$$

Here, h is the time dependent thickness of a tracer cell and T is the associated tracer concentration. Horizontal and vertical advection velocity components are written (\mathbf{u}, w) , and (\mathbf{F}, F_z) are the horizontal and vertical SGS flux components. The horizontal gradient operator is written ∇_z , and δ_k is the vertical finite difference operator acting across a discrete level k . Prognostic fields are updated in time increments of $\Delta\tau_{\text{leap}}$. The thickness of a tracer cell is updated analogously to the tracer, as required to maintain compatibility between volume and tracer evolution (see Section 10.7 as well as Griffies et al. (2001)).

The time tendency in equation (11.5) has been approximated with a centred in time discrete operator. Skipping the central time step τ introduces a spurious computational mode, where even and odd time steps decouple. We choose time filtering to suppress the associated instability, with \bar{h} and \bar{T} denoting the time filtered thickness and tracer concentration. Absent time filtering, the discrete time tendency has a second order global truncation error, whereas time filtering degrades the truncation error to first order (see Section 2.3.5 of Durran (1999)). We comment further on time filtering in the subsequent discussion, as it one of the two main reasons we consider alternative time stepping schemes to be preferable.

Global ocean models generally employ anisotropic grids, with significantly more refined vertical spacing than horizontal. When admitting realistically fast vertical mixing processes, parameterized by F_z , a time implicit method is used to overcome the stringent time step constraints of an explicit approach. Hence, F_z is evaluated at the future time $\tau + \Delta\tau_{\text{leap}}$. In contrast, coarser grid spacing in the horizontal generally allows for an explicit implementation of the horizontal SGS fluxes. Due to the dissipative nature of SGS fluxes, stability considerations require them to be evaluated at the lagged time $\tau - \Delta\tau_{\text{leap}}$, with evaluation at the central time τ numerically unstable. That is, the horizontal SGS fluxes are implemented with a forward time step of size $2\Delta\tau_{\text{leap}}$.

In contrast to dissipative terms, numerical stability dictates that tracer concentration in the advection operator be evaluated at the central time τ if using central spatial differencing. As reviewed by Griffies et al. (2000a), this approach was common in z-models for decades, particularly prior to around 2005. This form of the time stepping gives rise to the commonly referred name *leap frog* applied to the older time stepping method used in MOM4.0. However, it is important to note that leap frog in the tracer equation is used only for advection, and only for central spatial discretizations of advection. Dissipative terms are implemented with either a forward or an implicit time step as described above.

For purposes of ocean climate modeling with OM3.0, we found the dispersive errors from central differenced tracer advection to be unacceptable, due to the introduction of spurious tracer extrema and large

levels of spurious diapycnal mixing when convective adjustment acts on dispersion errors (Griffies et al., 2000b). To help remedy these problems, we chose a third order upwind biased scheme. As reviewed in Durran (1999), upwind biasing introduces a damping or dissipative element to numerical advection. Consequently, upwind biased fluxes *must* be evaluated at the lagged time $\tau - \Delta\tau_{\text{leap}}$ just like the dissipative horizontal SGS fluxes. A similar situation arises when implementing the Quicker advection scheme, in which one separates a dissipative portion evaluated at the lagged time step from a non-dissipative piece evaluated at τ (Holland et al., 1998; Pacanowski and Griffies, 1999). This is the origin of the two time labels placed on the tracer concentration for the advective flux in equation (11.5).

For the Sweby advection scheme used in OM3.0, the split into dissipative and non-dissipative terms is not possible. The full advective flux is thus evaluated at the lagged time step $\tau - \Delta\tau_{\text{leap}}$. This result may suggest increased levels of dissipation using Sweby relative to Quicker. Indeed, this is the case in regions where dissipation is welcomed, such as near river mouths where Quicker was found to introduce unacceptable tracer extrema. In other regions, we have seen negligible differences between the two advection schemes.

An update of the thickness weighted baroclinic velocity using the leap-frog scheme on a B-grid takes the form

$$\frac{h^{\tau+1} \mathbf{u}^{\tau+1} - \bar{h}^{\tau-1} \bar{\mathbf{u}}^{\tau-1}}{2\Delta\tau} = -\mathcal{M}^\tau \hat{\mathbf{z}} \times h^\tau \mathbf{u}^\tau + (w^\tau \mathbf{u}^\tau)_k - (w^\tau \mathbf{u}^\tau)_{k-1} - \nabla_z \cdot (h^\tau \mathbf{u}^\tau \mathbf{u}^\tau) - h^\tau (f \hat{\mathbf{z}} \times \mathbf{u})_{\text{trapezoidal}} - h^\tau \nabla_z (p^\tau / \rho_o) + h^\tau (\mathbf{F}^u)^{(\tau-1, \tau+1)}. \quad (11.6)$$

As for the tracer update, time filtering is applied to the lagged values of velocity and velocity cell thickness to suppress time splitting. Central differences are used to spatially discretize velocity self-advection, thus necessitating its evaluation at the central time step. Pressure is temporally evaluated likewise. The friction operator $(\mathbf{F}^u)^{(\tau-1, \tau+1)}$ arises from horizontal and vertical fluid deformations. Analogous to the treatment of tracer SGS fluxes, horizontal deformations are evaluated at $\tau - \Delta\tau_{\text{leap}}$ (forward time step) and vertical deformations at $\tau + \Delta\tau_{\text{leap}}$ (implicit time step).

Inertial energy is realistic in the coupled climate model CM2.0 since it includes a diurnal cycle of solar insolation, and the atmosphere and sea ice fields passed to the ocean (wind stress, fresh water, turbulent and radiative fluxes)¹ are updated every two hours. Inertial energy has important contributions to the mixing coefficients determined by the model's boundary layer scheme.

The model's baroclinic time step $\Delta\tau_{\text{leap}} = \text{one hour}$ is smaller than that needed to resolve inertial oscillations (e.g., Section 12.8.3 of Griffies (2004)). We nonetheless encountered an inertial-like instability in the climate model's Arctic sector when implementing the Coriolis force explicitly in time (see Chapter 12 for a discussion of a discrete implementation of the Coriolis Force). This instability is presumably related to the coupling between the ocean and sea ice, although the precise mechanism remains under investigation. The climate model remained stable, however, when implementing the ocean's Coriolis force with a trapezoidal or semi-implicit method (Section 12.2). Hence, the semi-implicit method is employed in both OM3.0 and OM3.1.²

11.2.2 Problems related to tracer conservation

Consider now the discrete time tracer equation in the abbreviated form

$$(hT)^{\tau+\Delta\tau_{\text{leap}}} = (\bar{h}\bar{T})^{\tau-\Delta\tau_{\text{leap}}} + 2\Delta\tau G, \quad (11.7)$$

where G symbolizes the advective and diffusive terms as well as boundary fluxes (we ignore source/sink terms for brevity). Thickness at the lagged time step results from a time average as described in Griffies et al. (2001), whereas time filtering of tracer concentration is taken in the form suggested by Robert (1966)

¹As recommended by Pacanowski (1987), wind stress applied to the ocean surface is computed using the relative velocity between the atmospheric winds and the ocean currents.

²Recall that both OM3.0 and OM3.1 used the B-grid, which allows for an implicit implementation of the Coriolis force. See Sections 12.2 and 12.3 for details.

and Asselin (1972) (see also Section 2.3.5 of Durran (1999)).³ Integrating equation (11.7) over the model domain leads to the balance of total tracer content in the model. Total tracer at time $\tau + \Delta\tau_{\text{leap}}$ is determined by the input of tracer through boundaries during the $2\Delta\tau_{\text{leap}}$ time step, plus the volume integrated product of the time filtered thickness and tracer concentration, $\bar{h}\bar{T}$, at the lagged time $\tau - \Delta\tau_{\text{leap}}$. Notably, because of time filtering, the model's total tracer changes even in the case of zero boundary fluxes.

The magnitude of tracer change associated with time filtering can be negligible for many purposes, as discussed in Griffies et al. (2001). However, we found the changes unacceptable when developing ecosystem models, where precise conservation is desired. Additionally, filtering contributed to a globally averaged heat non-conservation in the climate model on the order of $\pm 0.03 \text{ W m}^{-2}$. This non-conservative heat flux is a few percent of the surface insolation change expected from doubling greenhouse gas concentrations in the atmosphere. It is therefore of concern for our climate change simulations. Consequently, alternative approaches were investigated.⁴

11.2.3 The time staggered scheme used in OM3.1

The alternative scheme we employ in OM3.1 discretizes the time derivative with a forward time step. That is, it does not skip any time levels. Additionally, it staggers tracer and velocity fields by one-half time step in a manner analogous to spatial staggering on Arakawa grids. We therefore refer to this method as a *time staggered scheme*.

Forward time stepping does not admit time splitting, and so no time filters are needed. The alternative scheme therefore ensures tracer is conserved, which is our primary motivation for moving away from the OM3.0 method involving the leap frog. There are other consequences of changing the time tendency discretization, and the purpose of this section is to expose these issues.

A time staggered update of thickness weighted tracer is given by

$$\frac{(hT)^{\tau+1/2} - (hT)^{\tau-1/2}}{\Delta\tau_{\text{stag}}} = -\nabla_z \cdot [(h\mathbf{u})^\tau T^{\tau-1/2} + h^{\tau-1/2} \mathbf{F}^{\tau-1/2}] - \delta_k [w^\tau T^{\tau-1/2} + F_z^{\tau+1/2}]. \quad (11.8)$$

The two equations (11.5) and (11.8) become identical when the following holds:

- time steps are related by $\Delta\tau_{\text{stag}} = 2\Delta\tau_{\text{leap}}$,
- time filtering in the OM3.0 leap frog method is not used,
- tracer advection employs an upwind biased scheme.

In effect, the time staggered method stays on just one of the two leap frog branches. This is the fundamental reason that the two methods should be expected, for many purposes, to yield similar solutions.

We note the following points to keep in mind when transitioning to the staggered approach from the leap-frog.

- Centred spatial differencing of advection is unstable with a forward time step. Hence, for tracer advection we *must* employ an upwind biased advection scheme when using the staggered approach. For our purposes with global ocean climate modelling, such advection schemes are motivated to resolve problems with other schemes. Nonetheless, this consequence of changing the time stepping scheme may be unacceptable for certain applications. An alternative method is to retain the ability to discretize advection with centred spatial differences, but to alter the temporal evaluation of the advection operator according to Adams-Bashforth methods (Durran, 1999), or other schemes. In particular, we chose a temporally third order accurate Adams-Bashforth method for velocity self-advection, thus maintaining the traditional centred spatial differences of this operator. The third order Adams-Bashforth method requires the advection operator at time steps τ , $\tau - 1$, and $\tau - 2$, thus increasing memory requirements.

³We chose filtering for tracer over the alternative of periodically using a forward or backward time step, which was the method used by Cox (1984). The use of a periodic forward or backward time step introduces an unphysical periodicity to the simulation, and in particular was found by Marotzke (1991) to interact in unphysical ways with convective adjustment.

⁴Leclair and Madec (2009) propose a method to maintain conservation with the leap-frog scheme. We propose an alternative staggered scheme for MOM discussed in Section 11.2.3.

- When choosing a forward time step for the tendency, the Coriolis force on the B-grid *must* be computed using an implicit or semi-implicit approach, whereas on the C-grid we wrap the Coriolis force into the momentum advection using a 3rd order Adams-Bashforth scheme (see Chapter 12 for details of the Coriolis force). A time explicit approach is numerically unstable using a forward time step. In contrast, the OM3.0 approach with the leap frog on the B-grid allows for an explicit leap frog time stepping of the Coriolis force, as well as the semi-implicit or implicit.
- A leap frog discretization of the time tendency updates the ocean state by $\Delta\tau_{\text{leap}}$ through taking a $2\Delta\tau_{\text{leap}}$ step for the discrete time tendency. Consequently, gravity waves and dissipative operators (i.e., diffusion, friction, and upwind biased advection) are time step constrained based on $2\Delta\tau_{\text{leap}}$. In contrast, the staggered scheme updates the ocean state by $\Delta\tau_{\text{stag}}$ and it employs $\Delta\tau_{\text{stag}}$ to compute tendencies. It is therefore time step constrained based on a $\Delta\tau_{\text{stag}}$ time step. Hence, the staggered time step $\Delta\tau_{\text{stag}}$ can generally be twice that of the leap frog $\Delta\tau_{\text{leap}}$

$$\Delta\tau_{\text{stag}} = 2\Delta\tau_{\text{leap}}. \quad (11.9)$$

The computational cost of OM3.1 with the staggered scheme is therefore one-half that of OM3.0 using the older leap frog based scheme. There can be little argument that such an improvement in efficiency is of great use for ocean modelling.

11.2.4 Sensitivity to the time stepping scheme

During the bulk of our development, the ocean model employed the older leap frog based time stepping scheme for tracer, baroclinic, and barotropic equations. Upon developing the staggered time stepping scheme for the tracer and baroclinic equations, we became convinced that the modified scheme has utility for our climate modelling applications. The question arose whether switching time stepping schemes would require retuning of the physical parameterizations.

Tests were run with the ocean and ice models using an annually repeating atmospheric forcing with daily synoptic variability, again repeating annually. Runs using the staggered scheme had a two hour time step for both tracer and baroclinic momentum, and a predictor-corrector scheme (e.g., Killworth et al., 1991; Griffies, 2004) for the barotropic equations with a 90s time step.⁵ The comparison was made to the older time stepping scheme using one hour time steps for the tracer and baroclinic equations, and (3600/64)s for the leap frog barotropic equations.

Analysis of these solutions after 10 years revealed that regions with relatively high frequency temporal variability, such as the equatorial wave guide, exhibit the most differences instantaneously. Figure 11.1 illustrates the situation along the equator in the East Pacific. The older scheme exhibits substantial time splitting, even with a nontrivial level of time filtering from a Robert-Asselin time filter. Moving just five degrees north of the equator, however, reveals that the simulation has much less relative variability, and a correspondingly negligible amount of time splitting. Even though the simulation along the equator showed substantial time splitting, over longer periods of time, the large scale patterns and annual cycles showed negligible differences between time stepping schemes. Indeed, time averaging, even over just a day, seems sufficient to smooth over most of the instantaneous differences.

Tests were then run with the GFDL coupled climate models CM2.0 (using OM3.0 as the ocean component) and CM2.1 (using OM3.1). Instantaneous differences were much larger, as expected due to the non-trivial natural variability in the coupled system with a freely evolving atmospheric component. Nonetheless, differences for large scale patterns and seasonal or longer time averages were within levels expected from the model's natural variability.

11.2.5 Dissipative aspects of the predictor-corrector

The purpose of this section is to expose the dissipative aspects of the predictor-corrector scheme available for use in the barotropic equations in MOM. A similar treatment is given in Section 12.8.1 of Griffies (2004).

⁵We found the predictor-corrector to be suitable for the barotropic equations due to our ability to increase the barotropic time step beyond that of the leap frog. Additionally, it preferentially dissipates grid scale features, which are commonly found when discretizing gravity waves on a B-grid (Killworth et al., 1991; Griffies et al., 2001). We present an analysis of the dissipative aspects in Section 11.2.5.

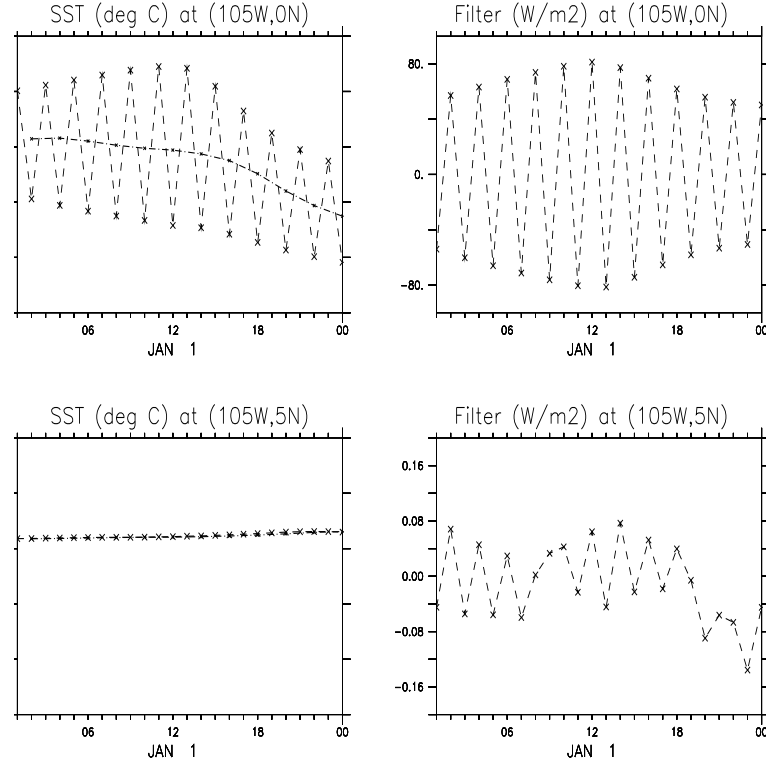


Figure 11.1: Upper left panel: Instantaneous sea surface temperature over January 1 at $(105^\circ\text{W}, 0^\circ\text{N})$ as realized in a simulation using the standard time stepping scheme with an hour tracer time step (noisy time series) and the staggered scheme with a two hour tracer time step (smooth time series). Upper right panel: Surface heating applied at $(105^\circ\text{W}, 0^\circ\text{N})$ from the Robert-Asselin time filter used to damp the leap frog splitting. Lower left panel: Instantaneous sea surface temperature over a single day at $(105^\circ\text{W}, 5^\circ\text{N})$ as realized in a simulation using the standard scheme with an hour tracer time step and the staggered scheme with a two hour tracer time step. Note the width of the temperature range is set the same as at the equator. In general, the agreement of the solution off the equator, where the leap frog splitting is minimal, is far greater than on the equator. Lower right panel: Surface heating applied at $(105^\circ\text{W}, 5^\circ\text{N})$ from the Robert-Asselin filter. Note the much smaller magnitude relative to the values on the equator.

In two space dimensions, the predictor-corrector equations for an update of the surface height and vertically integrated horizontal velocity in a Boussinesq model are

$$\frac{\eta^* - \eta^n}{\Delta t} = -\gamma \nabla \cdot \mathbf{U}^n \quad (11.10)$$

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = -c^2 \nabla \eta^* \quad (11.11)$$

$$\frac{\eta^{n+1} - \eta^n}{\Delta t} = -\nabla \cdot \mathbf{U}^{n+1}, \quad (11.12)$$

where n symbolizes the barotropic time step. For brevity we dropped the fresh water and source terms appearing in the free surface equation (10.69), and we assumed an unforced linear shallow water system with squared wave speed $c^2 = gH$. Setting the dimensionless dissipation parameter $\gamma \geq 0$ to zero recovers a *forward-backward* scheme discussed by Killworth et al. (1991). Keeping $\gamma > 0$ was useful in our simulations and was motivated by similar experiences in the Hallberg Isopycnal Model (Hallberg, 1997).

Eliminating the predicted surface height η^* leads to

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = -c^2 \nabla \eta^n + \gamma c^2 \Delta t \nabla [\nabla \cdot \mathbf{U}^n] \quad (11.13)$$

$$\frac{\eta^{n+1} - \eta^n}{\Delta t} = -\nabla \cdot \mathbf{U}^{n+1}. \quad (11.14)$$

To directly see how the surface height evolves, eliminate \mathbf{U} to find

$$\frac{\eta^{n+1} - 2\eta^n + \eta^{n-1}}{(\Delta t)^2} = (c \nabla)^2 \eta^n + \gamma (c \nabla)^2 (\eta^n - \eta^{n-1}). \quad (11.15)$$

Taking the limit $\Delta t \rightarrow 0$, yet with $\gamma \Delta t$ constant, leads to a dissipative wave equation

$$(\partial_{tt} - c^2 \nabla^2) \eta = (\gamma \Delta t) (c \nabla)^2 \partial_t \eta. \quad (11.16)$$

A single spatial Fourier mode with wavenumber amplitude κ thus satisfies

$$\left(d^2/dt^2 + \gamma \Delta t (c \kappa)^2 d/dt + (c \kappa)^2 \right) \eta = 0. \quad (11.17)$$

This is the equation for a damped harmonic oscillator with inverse e-folding time $(1/2) \gamma \Delta t (c \kappa)^2$. With $\gamma > 0$, external gravity waves are selectively dissipated in regions where the surface height is changing in time, and where the spatial scales are small. Faster waves are damped more readily than slower waves. These properties are useful when aiming to suppress the B-grid computational null mode discussed in [Killworth et al. \(1991\)](#) and [Griffies et al. \(2001\)](#).

11.3 Introduction to time stepping in MOM

For the remainder of this chapter, we step back from the OM3 simulations and revisit some of the basic algorithmic details of the time stepping schemes used in MOM. For this purpose, it is sufficient to focus on the Boussinesq version, where volume is conserved rather than mass. The exact same issues arise when using mass conserving non-Boussinesq vertical coordinates.

To start, we summarize advantages of the staggered time stepping scheme employed by MOM and introduced in Section 11.2.3 when discussing the OM3.1 model. For climate modelling, this scheme has proven to be a great improvement over the traditional leap-frog based methods found in earlier GFDL ocean codes, as well as many other ocean circulation models ([Griffies et al. \(2000a\)](#)). The improvements include the following.

- There is no need to employ explicit time filters (e.g., Robert-Asselin filter) with the staggered scheme, thus enhancing temporal accuracy over the time filtered leap-frog scheme.
- The time staggered scheme conserves seawater volume/mass and tracer mass to within numerical roundoff, whereas the leap-frog based methods, due to the use of explicit time filtering, fail to conserve.
- The time staggered scheme updates the state of the ocean one time step by employing tendencies based on that one time step. In contrast, leap-frog based schemes update the state over one time step by using tendencies based on two time steps. Hence, the leap-frog based schemes have a CFL stability constraint based on the two time step tendencies, and so can be run at only one-half the time step of the staggered scheme. Thus, the staggered scheme is generally one half the computational cost of the leap-frog based schemes.

The purpose of the following sections of this chapter is to expose salient points regarding the time stepping algorithm that have been raised when developing the radiating open boundary condition.

11.4 Basics of staggered time stepping in Boussinesq MOM

Fundamental to the time staggered method is the need to provide a surface height η at integer time steps $\eta(\tau)$ as well as half integer time steps $\eta(\tau + \Delta\tau/2)$. The surface height at integer time steps is needed to couple to velocity variables, which are placed on integer time steps, whereas half integer time steps provide a surface height for tracers. In addition, due to the split between barotropic and baroclinic modes, the surface height is updated over the small barotropic time steps $\eta^{(b)}(\tau, t_n)$, where a raised (b) denotes a field evaluated on the small barotropic time step. The first time label τ designates which baroclinic branch the cycle started, and

$$t_n = \tau + n\Delta t \quad (11.18)$$

is the barotropic time step. The relation between the large time step $\Delta\tau$ and small time step Δt is given by

$$2\Delta\tau = N\Delta t. \quad (11.19)$$

The barotropic time stepping proceeds from the initial barotropic time $t_0 = \tau$ to the final time $t_N = \tau + 2\Delta\tau$. The integer N is a function of the split between barotropic and baroclinic gravity waves, which can be on the order of 100 in a global model. Deducing the connection between $\eta(\tau)$, $\eta(\tau + \Delta\tau/2)$, and $\eta^{(b)}(\tau, t_n)$ is a focus of these notes. Correspondingly, we require a connection between the barotropic time cycled vertically integrated velocity $\mathbf{U}^{(b)}$ and $\mathbf{U}(\tau)$.

The barotropic cycle integrates over time $2\Delta\tau$ for every $\Delta\tau$ update of the baroclinic system. Why the doubling of the time integration? This method is based on experience with split-explicit time stepping schemes, where we have found it important to provide sufficient time averaging to damp instabilities arising from the incomplete split between the fast and slow motions available with a vertical integration. Longer time averaging is possible, though less convenient algorithmically, less accurate, and more expensive.

A fundamental constraint of any time stepping scheme is that the tracer and volume/mass equations must remain compatible. Compatibility means that the tracer concentration equation reduces to the volume or mass conservation equation when setting the tracer concentration to a constant. Without compatibility, tracer and volume/mass conservation are lost, and the algorithm is of limited use for ocean climate modelling.

After completing the barotropic cycle, which extends from $t_0 = \tau$ to $t_N = \tau + 2\Delta\tau$, we aim to have a prescription for updating the vertically integrated velocity $\mathbf{U}(\tau + \Delta\tau)$, the free surface height $\eta(\tau + \Delta\tau/2)$, as well as $\eta(\tau + \Delta\tau)$.

11.5 Predictor-corrector for the barotropic system

The preferred barotropic time stepping algorithm is the predictor-corrector scheme. The first step in the algorithm “predicts” the surface height (again, we are focusing on the Boussinesq version of MOM) via

$$\frac{\eta^{(*)}(\tau, t_{n+1}) - \eta^{(b)}(\tau, t_n)}{\gamma\Delta t} = -\nabla \cdot \mathbf{U}^{(b)}(\tau, t_n) + \mathcal{E}, \quad (11.20)$$

where \mathcal{E} is the fresh water forcing or volume source, both of which are held constant over the course of the barotropic cycle. We expose the time labels on these fields in later discussions. The raised (*) denotes an intermediate value of the surface height. This is the “predicted” value, to be later “corrected.” The nondimensional parameter $0 \leq \gamma$ acts to dissipate the small scales of motion (see Section 12.8 of Griffies (2004)). Setting $\gamma = 0$ recovers a second order accurate forward-backward scheme, in which case the predictor step (11.20) is eliminated. Larger values of γ reduce the order of accuracy, yet provide effective damping. However, as shown in Section 12.8 of Griffies (2004), values of γ larger than 1/4 can compromise the scheme’s stability. The value $\gamma = 1/5$ has been found useful for many purposes.

The predicted surface height $\eta^{(*)}(\tau, t_{n+1})$ is used to compute the surface pressure via

$$\rho_o \tilde{p}_s^{(*)}(\tau, t_{n+1}) = g \eta^{(*)}(\tau, t_{n+1}) \rho_{k=1}^{(\tau+1/2)} \quad (11.21)$$

where the applied pressure p_a has been dropped for brevity but can be trivially added. The surface pressure is used to update the vertically integrated velocity

$$\frac{\mathbf{U}^{(b)}(\tau, t_{n+1}) - \mathbf{U}^{(b)}(\tau, t_n)}{\Delta t} = \left[-f \hat{\mathbf{z}} \wedge \frac{\mathbf{U}^{(b)}(\tau, t_n) + \mathbf{U}^{(b)}(\tau, t_{n+1})}{2} - D(\tau) \nabla_z \tilde{p}_s^{(*)}(\tau, t_{n+1}) + \mathbf{G}(\tau) \right]. \quad (11.22)$$

For the vertically integrated transport, the Coriolis force on the B-grid version of MOM is evaluated using the Crank-Nicholson semi-implicit time scheme in equation (11.22). Inverting the B-grid semi-implicit approach provides an explicit update of the vertically integrated transport

$$U^{(b)}(\tau, t_{n+1}) = [1 + (f \Delta t/2)^2]^{-1} [U^{(\#)}(\tau, t_{n+1}) + (f \Delta t/2) V^{(\#)}(\tau, t_{n+1})] \quad (11.23)$$

$$V^{(b)}(\tau, t_{n+1}) = [1 + (f \Delta t/2)^2]^{-1} [V^{(\#)}(\tau, t_{n+1}) - (f \Delta t/2) U^{(\#)}(\tau, t_{n+1})] \quad (11.24)$$

where $\mathbf{U}^{(\#)}(\tau, t_{n+1})$ is updated just with the time-explicit tendencies

$$\frac{U^{(\#)}(\tau, t_{n+1}) - U^{(b)}(\tau, t_n)}{\Delta t} = (f/2) V^{(b)}(\tau, t_n) - D(\tau) \partial_x \tilde{p}_s^{(*)}(\tau, t_{n+1}) + G^x(\tau) \quad (11.25)$$

$$\frac{V^{(\#)}(\tau, t_{n+1}) - V^{(b)}(\tau, t_n)}{\Delta t} = -(f/2) U^{(b)}(\tau, t_n) - D(\tau) \partial_y \tilde{p}_s^{(*)}(\tau, t_{n+1}) + G^y(\tau). \quad (11.26)$$

For the C-grid version of MOM, the Coriolis force is evaluated using an Adams-Bashforth scheme (see Section 12.3).

The “corrector” part of the scheme steps the surface height using the updated transport

$$\frac{\eta^{(b)}(\tau, t_{n+1}) - \eta^{(b)}(\tau, t_n)}{\Delta t} = -\nabla \cdot \mathbf{U}^{(b)}(\tau, t_{n+1}) + \mathcal{E}. \quad (11.27)$$

Note that $\eta^{(b)}(\tau, t_{n+1})$ is used rather than the predicted height $\eta^{(*)}(\tau, t_{n+1})$, since $\eta^{(*)}(\tau, t_{n+1})$ is computed with the altered time step $\gamma \Delta t$. Temporal dissipation is localized to the predictor portion of the time stepping, with the corrector part hidden from this dissipation. Because of the predictor step, the convergence of the vertically integrated transport is computed twice in the predictor-corrector scheme, thus increasing the cost relative to a forward-backward approach where $\gamma = 0$. The payoff is an extra parameter that can be used to tune the level of dissipation. Additionally, there is added stability towards representing gravity waves so that Δt can be longer than when using the leap-frog method.

Let us detail how the barotropic steps accumulate over the course of a particular barotropic cycle. For this purpose, write out the first and second corrector steps (11.27) for the surface height

$$\frac{\eta^{(b)}(\tau, t_{n+1}) - \eta^{(b)}(\tau, t_{n=0})}{\Delta t} = F(t_{n=1}) \quad (11.28)$$

$$\frac{\eta^{(b)}(\tau, t_{n+2}) - \eta^{(b)}(\tau, t_{n=1})}{\Delta t} = F(t_{n=2}), \quad (11.29)$$

where the right-hand side of equation (11.27) is abbreviated as F . Adding these two equations leads to

$$\frac{\eta^{(b)}(\tau, t_{n+2}) - \eta^{(b)}(\tau, t_{n=0})}{\Delta t} = F(t_{n=1}) + F(t_{n=2}), \quad (11.30)$$

where the intermediate value $\eta^{(b)}(\tau, t_{n=1})$ has identically cancelled. This result easily generalizes, so that

$$\frac{\eta^{(b)}(\tau, t_{n=N}) - \eta^{(b)}(\tau, t_{n=0})}{N \Delta t} = \frac{1}{N} \sum_{n=1}^N F(t_n). \quad (11.31)$$

This result does not hold for a leap-frog algorithm, since the intermediate values of the surface height do not generally cancel completely, as they do here for the predictor-corrector.

11.6 The Griffies (2004) scheme

The only piece of the forcing $F(t_n)$ that changes during the barotropic cycle is the convergence of the vertically integrated velocity. The result (11.31) then suggests that the time averaged vertically integrated velocity should be given back to the baroclinic part of the model upon completion of the barotropic cycle. To have this velocity centered on the integer time step $\tau + \Delta\tau$, it is necessary to run the barotropic cycle to $\tau + 2\Delta\tau$. Hence, upon reaching the last barotropic time step

$$t_{n=N} = \tau + 2\Delta\tau, \quad (11.32)$$

the vertically integrated velocity is time averaged,

$$\tilde{\mathbf{U}} = \frac{1}{N} \sum_{n=1}^N \mathbf{U}^{(b)}(\tau, t_n). \quad (11.33)$$

To produce the updated vertically integrated velocity at baroclinic time $\tau + \Delta\tau$, the vertically integrated velocity $\mathbf{U}(\tau + \Delta\tau)$ is identified with this time averaged value,

$$\mathbf{U}(\tau + \Delta\tau) \equiv \tilde{\mathbf{U}}. \quad (11.34)$$

The surface height is needed at the integer time steps in order to specify the thickness of the velocity cells. There are two options for updating the surface height to time step $\tau + \Delta\tau$. First, we could take the instantaneous value from the barotropic portion of the cycle

$$\eta(\tau + \Delta\tau) \equiv \eta^{(b)}(\tau, t_{n=N/2}). \quad (11.35)$$

This approach has not been tried, since it likely leads to a meta-stable algorithm due to the absence of time averaging, depending on the predictor-corrector dissipation parameter γ . In contrast, extensive experience indicates that added stability is realized by using the time averaged surface height

$$\eta(\tau + \Delta\tau) = \frac{1}{N+1} \sum_{n=0}^N \eta^{(b)}(\tau, t_n). \quad (11.36)$$

Notably, tracer and volume conservation is not compromised by this specification since it is only used to define the surface height carried by the velocity cells. However, the surface height at half integer timesteps needed for the tracer equation is diagnosed using equation (11.33),

$$\frac{\eta(\tau + \Delta\tau/2) - \eta(\tau - \Delta\tau/2)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau) + q_w(\tau) + \mathcal{S}^{(\eta)}(\tau). \quad (11.37)$$

As described in Section 11.7, this approach may cause divergence of sea level at integer and half integer time steps.

11.7 Algorithms motivated from predictor-corrector

The previous algorithm makes a distinction between how the integer and half-integer surface heights are updated. This distinction exposes the algorithm to possible time splitting between these surface heights. The splitting has been found to be unacceptable for models with radiating open boundary conditions, whereas other boundary conditions have shown no problem. Given the interest in radiating boundary conditions, we consider here an alternative approach which is motivated from details of the barotropic predictor-corrector method. It will turn out that the schemes developed here are not algorithmically closed. However, approximations are considered in 11.8.2 to close the algorithms.

11.7.1 Barotropic time stepping and surface height on integer time steps

The predictor step in the predictor-corrector algorithm updates the surface height according to

$$\eta^{(b)}(\tau, t_{n+1}) - \eta^{(b)}(\tau, t_n) = \Delta t F(t_{n+1}). \quad (11.38)$$

We now expose the time labels on all terms appearing in the forcing, whereby we set volume sources

$$F(t_{n+1}) = -\nabla \cdot \mathbf{U}^{(b)}(\tau, t_{n+1}) + q_w(\tau + \Delta\tau/2) + \mathcal{S}^{(\eta)}(\tau + \Delta\tau/2). \quad (11.39)$$

Only the convergence of the vertically integrated velocity $\mathbf{U}^{(b)}(\tau, t_{n+1})$ changes on each barotropic time step, whereas the water forcing $q_w(\tau + \Delta\tau/2)$ and source $\mathcal{S}^{(\eta)}(\tau + \Delta\tau/2)$ are held fixed.

To begin the barotropic integration of the surface height, it is necessary to prescribe an initial condition. We choose to set

$$\eta^{(b)}(\tau, t_0) = \eta(\tau) \quad (11.40)$$

for the surface height, and

$$\mathbf{U}^{(b)}(\tau, t_0) = \mathbf{U}(\tau) \quad (11.41)$$

for the vertically integrated velocity. This choice of the starting point is essential for η , but different approximations are possible for the vertically integrated velocity. Here, $\eta(\tau)$ and $\mathbf{U}(\tau)$ are centred at an integer baroclinic time step, which again is the time step where the velocity field is centred using the MOM time staggered method. These two prescriptions lead us to ask how to compute the updated surface height $\eta(\tau + \Delta\tau)$ and vertically integrated velocity $\mathbf{U}(\tau + \Delta\tau)$. Experience with various versions of the split-explicit time stepping in MOM motivates us to take for the surface height a time average over the full suite of barotropic surface heights according to

$$\eta(\tau + \Delta\tau) = \frac{1}{N+1} \sum_{n=0}^N \eta^{(b)}(\tau, t_n). \quad (11.42)$$

We choose this simple form of time filtering, in which all terms within the barotropic time stepping contribute equally. Although more sophisticated time filters are available, this one has been found suitable for our purposes. Without time filtering, the algorithm can be very unstable and thus is unsuitable for large scale modelling. As shown below, this time averaging for the surface height motivates a different form for the time averaging of the vertically integrated velocity field.

11.7.2 Surface height on half-integer time steps

The fundamental prescription (11.42) for the integer time step surface height is readily extended to the half-integer surface height by setting

$$\eta(\tau + \Delta\tau/2) \equiv \frac{\eta(\tau) + \eta(\tau + \Delta\tau)}{2}. \quad (11.43)$$

This prescription couples the integer and half-integer time steps, and ensures that both are determined by time averages over the barotropic cycle. The question then arises how to make this prescription compatible with the time stepping for the tracer concentration. Compatibility is required for conservation of volume and tracer, and so is of fundamental importance. Compatibility is also needed with the baroclinic velocity scheme, but keeping in mind the uncertainties of wind stress parameterisation, minor approximation should be possible. Addressing these issues forms the remainder of this section.

To proceed, we first deduce the time stepping algorithm for the integer time steps which is implied from the barotropic time stepping (11.38) and the time average (11.42). For this purpose, start by using the initial condition (11.40) in the time average (11.42) to find

$$\eta(\tau + \Delta\tau) = \frac{1}{N+1} \sum_{n=0}^N \eta^{(b)}(\tau, t_n) \quad (11.44)$$

$$= \frac{\eta(\tau)}{N+1} + \frac{1}{N+1} \sum_{n=1}^N \eta^{(b)}(\tau, t_n). \quad (11.45)$$

Iterating the barotropic time stepping equation (11.38) and using the initial condition (11.40) renders

$$\eta^{(b)}(\tau, t_n) = \eta(\tau) + \Delta t \sum_{i=1}^n F(t_i). \quad (11.46)$$

Substitution of this result into equation (11.45) then leads to

$$\eta(\tau + \Delta\tau) - \eta(\tau) = \frac{\Delta t}{N+1} \sum_{n=1}^N \sum_{i=1}^n F(t_i). \quad (11.47)$$

The double sum on the right hand side arises from the need to ensure that over the course of the barotropic cycle, changes in volume correspond to changes in forcing; in particular, with changes in the convergence of the depth integrated transport. To facilitate computing the double sum within the barotropic time stepping scheme, we employ the following identity to reduce the double sum to a single sum

$$\sum_{n=1}^N \sum_{i=1}^n F(t_i) = \sum_{n=1}^N (N - n + 1) F(t_n), \quad (11.48)$$

which can be readily verified by induction.

The sum (11.48) does not represent a straightforward time average. It does, nonetheless, motivate defining a “modified average” forcing that is implied by the barotropic cycle running from $t_{n=0} = \tau$ to $t_N = \tau + 2\Delta\tau$. In particular, the relation $2\Delta\tau = N\Delta t$ between baroclinic and barotropic time steps motivates the following definition for the averaged forcing

$$\begin{aligned} \bar{F} &\equiv \frac{2}{N(N+1)} \sum_{n=1}^N \sum_{i=1}^n F(t_i), \\ &\equiv \frac{2}{N(N+1)} \sum_{n=1}^N (N - n + 1) F(t_n), \end{aligned} \quad (11.49)$$

which renders

$$\frac{\eta(\tau + \Delta\tau) - \eta(\tau)}{\Delta\tau} = \bar{F}. \quad (11.50)$$

Note that the average operator (11.49) reduces to the trivial result $\bar{F} = F$ in the special case when each of the barotropic time steps see a constant forcing $F(t_i) \equiv F$. That is,

$$\sum_{n=1}^N \sum_{i=1}^n F(t_i) = F \sum_{n=1}^N n \quad (11.51)$$

$$= F(N/2)(N+1), \quad (11.52)$$

where the last step used a common summation identity. This special case supports our definition of the averaging operator, and furthermore checks the integrity of the manipulations. In particular, since the fresh water and volume source are assumed to be constant over the barotropic time steps, we have

$$\bar{F} = \frac{2}{N(N+1)} \sum_{n=1}^N \sum_{i=1}^n F(t_i) \quad (11.53)$$

$$= -\frac{2}{N(N+1)} \sum_{n=1}^N \sum_{i=1}^n \nabla \cdot \mathbf{U}^{(b)}(\tau, t_i) + q_w(\tau + \Delta\tau/2) + \mathcal{S}^{(\eta)}(\tau + \Delta\tau/2) \quad (11.54)$$

which leads to

$$\frac{\eta(\tau + \Delta\tau) - \eta(\tau)}{\Delta\tau} = -\nabla \cdot \bar{\mathbf{U}} + q_w(\tau + \Delta\tau/2) + \mathcal{S}^{(\eta)}(\tau + \Delta\tau/2). \quad (11.55)$$

So far, we have avoided placing a time label on the modified average operator. In particular, the question arises how the averaged vertically integrated velocity

$$\bar{\mathbf{U}} = \frac{2}{N(N+1)} \sum_{n=1}^N \sum_{i=1}^n \mathbf{U}^{(b)}(\tau, t_i) \quad (11.56)$$

is related to the updated velocity $\mathbf{U}(\tau + \Delta\tau)$ or $\mathbf{U}(\tau + \Delta\tau/2)$. Absent the second summation, the resulting average would be closely centred on the time step $\tau + \Delta\tau$, but the barycenter of the double sum is at $\tau + \Delta\tau/2$. We now discuss algorithms based on both approximations.

11.7.3 Method A: $\mathbf{U}(\tau + \Delta\tau) = \bar{\mathbf{U}}$

In this method, we consider $\mathbf{U}(\tau + \Delta\tau) = \bar{\mathbf{U}}$, so that

$$\mathbf{U}(\tau + \Delta\tau) = \frac{2}{N(N+1)} \sum_{n=1}^N \sum_{i=1}^n \mathbf{U}^{(b)}(\tau, t_i). \quad (11.57)$$

Given this assumed time labelling of $\bar{\mathbf{U}}$, we are able to update the three dimensional velocity to the new time step $\tau + \Delta\tau$ after the baroclinic velocity is updated. The prescription (11.57) implies that the integer time step surface height, which is computed as the time average in equation (11.42), also satisfies the following time tendency equation

$$\frac{\eta(\tau + \Delta\tau) - \eta(\tau)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau + \Delta\tau) + q_w(\tau + \Delta\tau/2) + \mathcal{S}^{(\eta)}(\tau + \Delta\tau/2). \quad (11.58)$$

The definition (11.43) of the half-integer time step surface height then implies that it satisfies the tendency equation

$$\frac{\eta(\tau + \Delta\tau/2) - \eta(\tau - \Delta\tau/2)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau + \Delta\tau/2) + q_w(\tau) + \mathcal{S}^{(\eta)}(\tau), \quad (11.59)$$

where

$$\mathbf{U}(\tau + \Delta\tau/2) = \frac{\mathbf{U}(\tau) + \mathbf{U}(\tau + \Delta\tau)}{2} \quad (11.60)$$

$$q_w(\tau) = \frac{q_w(\tau + \Delta\tau/2) + q_w(\tau - \Delta\tau/2)}{2} \quad (11.61)$$

$$\mathcal{S}^{(\eta)}(\tau) = \frac{\mathcal{S}^{(\eta)}(\tau + \Delta\tau/2) + \mathcal{S}^{(\eta)}(\tau - \Delta\tau/2)}{2}. \quad (11.62)$$

11.7.3.1 Compatible tracer concentration

For the surface height on half integer time steps, we must maintain compatibility with tracer concentration fields, which are also centered on half-integer time steps. Compatibility means that time stepping the surface height must take the identical form to time stepping tracer concentration, so that the two equations agree in the special case of a constant tracer concentration. Without such compatibility, tracer and volume are not conserved by the discrete model. This point was emphasized by Griffies et al. (2001) in the context of the leap-frog based algorithm exclusively used in earlier versions of MOM.

Compatibility implies that the tracer concentration must be forced with the water source (11.61), the volume source (11.62), and, because of equation (11.59), with the half-integer advection velocity. Given these considerations, a compatible staggered time discretization of thickness weighted tracer, absent sub-grid scale processes, takes the form (note the shorthand used for the time labels)

$$\frac{h^{\tau+1/2} C^{\tau+1/2} - h^{\tau-1/2} C^{\tau-1/2}}{\Delta\tau} = -\nabla_s \cdot [(h\mathbf{u})^{\tau+1/2} C^{\tau-1/2}] \quad (11.63)$$

$$+ [w^{\tau+1/2} C^{\tau-1/2}]_k - [w^{\tau+1/2} C^{\tau-1/2}]_{k-1}, \quad (11.64)$$

where the thickness weighted advection velocity on half-integer time steps is given by

$$(h\mathbf{u})^{\tau+1/2} = \frac{(h\mathbf{u})^{\tau+1} + (h\mathbf{u})^{\tau}}{2}. \quad (11.65)$$

11.7.3.2 Why this scheme is not closed

This algorithm is not closed, and so is not practical. The reason is that the compatible tracer equation (11.64) requires the thickness weighted advection velocity $(h\mathbf{u})^{\tau+1/2}$. However, this velocity requires the updated thickness weighted velocity $(h\mathbf{u})^{\tau+1}$, but the velocity at time $\mathbf{u}^{\tau+1}$ is not known until the momentum is updated.

11.7.4 Method B: $\mathbf{U}(\tau + \Delta\tau/2) = \bar{\mathbf{U}}$

Since the barycenter of the double sum (11.48) is $\tau + \Delta\tau/2$, it is reasonable to prescribe $\mathbf{U}(\tau + \Delta\tau/2) = \bar{\mathbf{U}}$, so that

$$\mathbf{U}(\tau + \Delta\tau/2) = \frac{2}{N(N+1)} \sum_{n=1}^N \sum_{i=1}^n \mathbf{U}^{(b)}(\tau, t_i). \quad (11.66)$$

The prescription (11.57) implies that the integer time step surface height, which is computed as the time average in equation (11.42), also satisfies the following time tendency equation

$$\frac{\eta(\tau + \Delta\tau) - \eta(\tau)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau + \Delta\tau/2) + q_w(\tau + \Delta\tau/2) + \mathcal{S}^{(\eta)}(\tau + \Delta\tau/2). \quad (11.67)$$

The definition (11.43) of the half-integer time step surface height then implies

$$\frac{\eta(\tau + \Delta\tau/2) - \eta(\tau - \Delta\tau/2)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau) + q_w(\tau) + \mathcal{S}^{(\eta)}(\tau), \quad (11.68)$$

where

$$\mathbf{U}(\tau) = \frac{\mathbf{U}(\tau + \Delta\tau/2) + \mathbf{U}(\tau - \Delta\tau/2)}{2} \quad (11.69)$$

$$q_w(\tau) = \frac{q_w(\tau + \Delta\tau/2) + q_w(\tau - \Delta\tau/2)}{2} \quad (11.70)$$

$$\mathcal{S}^{(\eta)}(\tau) = \frac{\mathcal{S}^{(\eta)}(\tau + \Delta\tau/2) + \mathcal{S}^{(\eta)}(\tau - \Delta\tau/2)}{2}. \quad (11.71)$$

11.7.4.1 Compatible tracer concentration

Compatibility implies that in contrast to Section 11.7.3.1, the tracer concentration must be forced with the water source (11.70), the volume source (11.71), and, because of Equation (11.68) with the integer advection velocity. The compatible staggered time discretization of thickness weighted tracer, absent subgrid scale processes, takes the form

$$\frac{h^{\tau+1/2} C^{\tau+1/2} - h^{\tau-1/2} C^{\tau-1/2}}{\Delta\tau} = -\nabla_s \cdot [(h\mathbf{u})^{\tau} C^{\tau-1/2}] \quad (11.72)$$

$$+ [w^{\tau} C^{\tau-1/2}]_k - [w^{\tau} C^{\tau-1/2}]_{k-1}, \quad (11.73)$$

where the thickness weighted advection velocity on integer time steps is given by

$$(h\mathbf{u})^{\tau} = \frac{(h\mathbf{u})^{\tau+1/2} + (h\mathbf{u})^{\tau-1/2}}{2}. \quad (11.74)$$

11.7.4.2 Why this scheme is not closed

This scheme is not closed algorithmically. That is, the velocity scheme cannot be completed after the barotropic sequence is ready, because $\mathbf{U}(\tau + \Delta\tau)$ is still unknown. The compatible tracer equation (11.73) requires also the thickness weighted advection velocity $(h\mathbf{u})^{\tau+1/2}$, which itself requires the updated thickness weighted velocity $(h\mathbf{u})^{\tau+1}$. The velocity at time $\tau+1$, however, is not known until the momentum is updated which in turn requires the barotropic scheme to be completed. Yet the momentum is updated only after the tracer is updated. The repeated mapping between integer to half integer steps would reduce accuracy. A significant rearrangement of the baroclinic and tracer equation may facilitate the use of this algorithm. However, many attempts have failed.

11.8 Algorithms enforcing compatibility

We present three methods for time stepping the equations in MOM. Method I is that one discussed in Section 11.6 based on Griffies (2004) and Griffies et al. (2005). Method III is a modification to Method I, and Method II is a closed algorithm based on the barotropic predictor-corrector from Section 11.7. Methods II and III each aim to provide a closed and compatible scheme that maintains stability with the radiating open boundary condition. Methods I and II are implemented in MOM, with Method III remaining untested.

11.8.1 Method I: Griffies (2004)

We first summarize the method of Griffies (2004) and Griffies et al. (2005), as described in Section 11.6. To produce an algorithm that maintains compatibility with tracer concentration, and is algorithmically closed, we take the philosophy here that the fundamental fields are those which live on the baroclinic time steps (including baroclinic velocity and tracer fields). The barotropic fields are coupled to the baroclinic and tracer fields, but details of the barotropic algorithm do not dictate details of the baroclinic and tracer algorithm. In particular, details of whether we use a barotropic leap-frog or predictor-corrector are unimportant, nor are details of the initial values used for the surface height and vertically integrated velocity (so long as the initial values are reasonable). This philosophy is in contrast to that taken in Section 11.7, and further described in Method III below, where the barotropic predictor-corrector motivated details of the baroclinic and tracer updates.

The main steps of this scheme prescribe an updated vertically integrated velocity and updated surface height, both as time averages over the barotropic time steps

$$\mathbf{U}(\tau + \Delta\tau) = \frac{1}{N} \sum_{n=1}^N \mathbf{U}^{(b)}(\tau, t_n) \quad (11.75)$$

$$\eta(\tau + \Delta\tau) = \frac{1}{N+1} \sum_{n=0}^N \eta^{(b)}(\tau, t_n). \quad (11.76)$$

The half-integer time step surface height is driven by the convergence of the time averaged vertically integrated velocity, as well as surface boundary fluxes and interior volume sources

$$\frac{\eta(\tau + \Delta\tau/2) - \eta(\tau - \Delta\tau/2)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau) + q_w(\tau) + \mathcal{S}^{(\eta)}(\tau). \quad (11.77)$$

The compatible evolution equation for the tracer concentration follows from the update to the half-integer surface height

$$\frac{h^{\tau+1/2} C^{\tau+1/2} - h^{\tau-1/2} C^{\tau-1/2}}{\Delta\tau} = -\nabla_s \cdot [(h\mathbf{u})^\tau C^{\tau-1/2}] \quad (11.78)$$

$$+ [w^\tau C^{\tau-1/2}]_k - [w^\tau C^{\tau-1/2}]_{k-1}. \quad (11.79)$$

There is a distinction in this method between $\eta(\tau + \Delta\tau)$, which is based on a time average, and $\eta(\tau + \Delta\tau/2)$, which is based on a baroclinic forward time step. This dichotomy has been found to allow splitting between the surface heights when using radiating open boundary conditions.

11.8.2 Method II: Algorithm based on barotropic predictor-corrector

We were led to the non-closed algorithms in Section 11.7 by aiming to treat the barotropic system in a systematic manner, and by prescribing the use of a particular form of time averaging for the surface height. Alternative methods can be found by altering the form of the time average, or by jettisoning time averaged operators altogether. However, we are not in favour of jettisoning the time average, as past explorations of this approach have led to unacceptably unstable schemes. Instead, we consider approximations described in the following that close the algorithm.

To start, we follow the scheme proposed in Section 11.7.4, in which the integer time step surface height is updated via a time average as in equation (11.76)

$$\eta(\tau + \Delta\tau) = \frac{1}{N+1} \sum_{n=0}^N \eta^{(b)}(\tau, t_n), \quad (11.80)$$

and the half-integer time step surface height is also a time average via

$$\eta(\tau + \Delta\tau/2) \equiv \frac{\eta(\tau) + \eta(\tau + \Delta\tau)}{2}. \quad (11.81)$$

Following the details of the barotropic predictor-corrector, we are led to the updated vertically integrated velocity via the sum in equation (11.66) and using the barycenter for the time step placement as in Section 11.7.4

$$\begin{aligned} \mathbf{U}(\tau + \Delta\tau/2) &= \frac{2}{N(N+1)} \sum_{n=1}^N \sum_{i=1}^n \mathbf{U}^{(b)}(\tau, t_i) \\ &= \frac{2}{N(N+1)} \sum_{n=1}^N (N-n+1) \mathbf{U}^{(b)}(\tau, t_i). \end{aligned} \quad (11.82)$$

As described in Section 11.7.2, this sum arises from the need to maintain consistency with volume fluxes passing across the radiating open boundaries, and thus for providing a stable scheme with radiating open boundaries.

As noted in Section 11.7.4, the prescription (11.82) does not lead to a closed algorithm, since we need to know the updated velocity $\mathbf{U}(\tau + \Delta\tau)$ at the end of the barotropic cycle in order to update the three dimensional velocity field $\mathbf{u}(\tau + \Delta\tau)$. The following approximation which closes the algorithm has also been found to lead to a stable scheme with radiating open boundaries

$$\mathbf{U}(\tau + \Delta\tau) \approx \mathbf{U}(\tau + \Delta\tau/2). \quad (11.83)$$

The half-integer time step surface height, which is defined by the time average (11.81), also satisfies the time tendency equation

$$\frac{\eta(\tau + \Delta\tau/2) - \eta(\tau - \Delta\tau/2)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau) + q_w(\tau) + S^{(\eta)}(\tau). \quad (11.84)$$

It follows that the compatible tracer equation is given by

$$\frac{h^{\tau+1/2} C^{\tau+1/2} - h^{\tau-1/2} C^{\tau-1/2}}{\Delta\tau} = -\nabla_s \cdot [(h\mathbf{u})^\tau C^{\tau-1/2}] \quad (11.85)$$

$$+ [w^\tau C^{\tau-1/2}]_k - [w^\tau C^{\tau-1/2}]_{k-1}. \quad (11.86)$$

The discrete tracer equation thus takes the same form as in Methods I and III.

11.8.3 Method III: Modified Griffies (2004)

To possibly resolve the problem of splitting between the integer and half-integer time steps encountered with Method I in radiating open boundary problems, we consider here an alternative approach, whereby

the integer time step surface height is prescribed as the time average of the half-integer time step surface height

$$2\eta(\tau + \Delta\tau) = \eta(\tau + \Delta\tau/2) + \eta(\tau + 3\Delta\tau/2). \quad (11.87)$$

That is, the integer time step surface height is no longer based on a time average, but instead evolves according to

$$\frac{\eta(\tau + \Delta\tau) - \eta(\tau)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau + \Delta\tau/2) + q_w(\tau + \Delta\tau/2) + S^{(\eta)}(\tau + \Delta\tau/2), \quad (11.88)$$

where

$$2\mathbf{U}(\tau + \Delta\tau/2) = \mathbf{U}(\tau + \Delta\tau) + \mathbf{U}(\tau) \quad (11.89)$$

$$2q_w(\tau + \Delta\tau/2) = q_w(\tau + \Delta\tau) + q_w(\tau) \quad (11.90)$$

$$2S^{(\eta)}(\tau + \Delta\tau/2) = S^{(\eta)}(\tau + \Delta\tau) + S^{(\eta)}(\tau). \quad (11.91)$$

The problem with this prescription is that it is not closed, since the surface boundary condition module only provides information about the surface forcing at the present time step. Likewise, we do not know the updated volume source. Hence, to close the algorithm we make the following approximation

$$q_w(\tau + \Delta\tau/2) \approx q_w(\tau) \quad (11.92)$$

$$S^{(\eta)}(\tau + \Delta\tau/2) \approx S^{(\eta)}(\tau), \quad (11.93)$$

which amounts to saying that the boundary forcing and volume source term remain constant over the course of a baroclinic time step; i.e., we cannot obtain information at higher frequency for these fields. Hence, we are led to the following update for the integer time step surface height

$$\frac{\eta(\tau + \Delta\tau) - \eta(\tau)}{\Delta\tau} = -\nabla \cdot \mathbf{U}(\tau + \Delta\tau/2) + q_w(\tau) + S^{(\eta)}(\tau). \quad (11.94)$$

Although of some interest, this scheme has not yet been coded in MOM. It thus remains untested.

Chapter 12

DISCRETE SPACE-TIME CORIOLIS FORCE

Contents

12.1	The Coriolis force and inertial oscillations	177
12.1.1	B-grid considerations	178
12.1.2	C-grid considerations	178
12.2	Time stepping for the B-grid version of MOM	179
12.2.1	Explicit temporal discretization with a leap frog	179
12.2.2	Semi-implicit time discretization with a leap frog	180
12.2.3	Semi-implicit time discretization with a forward time step	180
12.2.4	Discretization for the B-grid MOM	181
12.2.4.1	Algorithm in the code	181
12.2.4.2	Namelist parameters	182
12.2.5	Energy analysis	182
12.3	Time stepping for the C-grid version of MOM	182

The purpose of this chapter is to present the methods used in MOM for discretizing the Coriolis force in space and time. We pay particular attention to the distinct needs of a B-grid and C-grid implementation (Section 9.1), as well as considering differences between forward time stepping (Section 11.4) and the older leap frog (Section 11.2). Some of this material was presented in the MOM4 Guide of [Griffies et al. \(2004\)](#), with new considerations here to handle the density and thickness weighting used in MOM.

The following MOM modules are directly connected to the material in this chapter:

ocean_core/ocean_coriolis.F90
ocean_core/ocean_velocity.F90

12.1 The Coriolis force and inertial oscillations

The inviscid horizontal momentum equation in the absence of pressure gradient forces is given by

$$\left(\frac{d}{dt} + f \hat{\mathbf{z}} \wedge \right) \mathbf{u} = 0, \quad (12.1)$$

which is equivalent to the second order free oscillator equation

$$\left(\frac{d^2}{dt^2} + f^2 \right) \mathbf{u} = 0. \quad (12.2)$$

Here, d/dt is the material time derivative relevant for Lagrangian observers. Motions which satisfy this equation are termed *inertial oscillations* and they have period given by

$$T_{\text{inertial}} = \frac{2\pi}{f} = \frac{11.97}{\sin \phi} \text{ hour} \quad (12.3)$$

where $\Omega = 7.292 \times 10^{-5} \text{ s}^{-1}$ is the earth's angular speed. The period of inertial oscillations is smallest at the North pole where $\phi = \pi/2$ and $T_{\text{smallest}} \approx 12 \text{ hour}$. An explicit temporal discretization of the inertial oscillation equation (12.1) will be unstable if the time step is longer than some fraction of the inertial period, where the fraction depends on details of the time stepping.

12.1.1 B-grid considerations

Coarse resolution models (models with resolutions on the order of 3 degrees or coarser) generally have weak advection velocities. Hence, these models can have their baroclinic momentum equation partially time step limited by inertial oscillations. To get around this limitation, a semi-implicit temporal treatment has been traditionally considered, as in Bryan (1969). Temporally implicit treatment is available only for the B-grid, where the two horizontal velocity components sit at the same grid point (Figure 9.1).

Additional issues with coupling to sea ice may warrant an implicit treatment even for ocean models run with a momentum time step that well resolves the inertial period. In these cases, temporal details of ocean-ice coupling have been found to cause enhanced energy at the inertial period. Semi-implicit time stepping of the Coriolis force may assist in damping this energy.

It is for these reasons that MOM provides an option for implementing the Coriolis force on the B-grid either explicitly in time or semi-implicitly for the baroclinic portion of the model. The namelist parameter `acor` sets the level of implicitness, as described in Section 12.2.4.2. For the barotropic time stepping on the B-grid, MOM generally uses a semi-implicit approach (Section 11.5).

12.1.2 C-grid considerations

Horizontal velocity components sit at different faces of the tracer cell (Figure 9.2). Hence, a spatial averaging must be applied to bring the Coriolis force onto the proper position. Consider the Coriolis force acting on the zonal velocity component $u_{i,j}$ sitting at the east face of the tracer cell $T(i, j)$. There are various ways to construct the averaging. We follow that used in GOLD for the energy conserving approach of Sadourny (1975), in which the zonal Coriolis force per area acting to accelerate the zonal velocity $u_{i,j}$ is given by

$$\begin{aligned} (f v \rho dz)_{\text{zonal Coriolis force}} &\approx (1/4) \left[f_{i,j} (v \rho dz)_{i,j} + f_{i,j} (v \rho dz)_{i+1,j} + f_{i,j-1} (v \rho dz)_{i,j-1} + f_{i,j-1} (v \rho dz)_{i+1,j-1} \right] \\ &= \overline{\left[f \left((v \rho dz)_{i-1}^x \right) \right]_{j-1}^y}, \end{aligned} \quad (12.4)$$

where we introduced northeast grid averaging operators

$$\overline{A}_i^x = \frac{A_i + A_{i+1}}{2} \quad (12.5)$$

$$\overline{A}_j^y = \frac{A_j + A_{j+1}}{2}. \quad (12.6)$$

The discretization (12.4) computes the Coriolis parameter f at the vorticity point (northeast corner of the tracer cell), which accords with the energy conserving method of Sadourny (1975) and GOLD. The normalization by 1/4 holds regardless the land-sea mask, as such is required to maintain global energy conservation from the Coriolis force. The meridional Coriolis force per area acting to accelerate the meridional velocity $v_{i,j}$ follows similarly to the zonal Coriolis force

$$\begin{aligned} -(f u \rho dz)_{\text{merid Coriolis force}} &\approx -(1/4) \left[f_{i-1,j} (u \rho dz)_{i-1,j} + f_{i-1,j} (u \rho dz)_{i-1,j+1} + f_{i,j} (u \rho dz)_{i,j} + f_{i,j} (u \rho dz)_{i,j+1} \right] \\ &= -\overline{\left[f \left((u \rho dz)_{j-1}^y \right) \right]_{i-1}^x}. \end{aligned} \quad (12.7)$$

The spatial averaging used on the C-grid makes it impractical to compute the Coriolis force implicitly in time.¹ Furthermore, the C-grid in MOM has been implemented solely for the staggered two-level time scheme. To maintain temporal stability, we thus follow the approach used for the advection of momentum (Section 11.2.3), in which a third order Adams-Bashforth method is used (Durrant, 1999).

12.2 Time stepping for the B-grid version of MOM

We now consider examples relevant to MOM of how the Coriolis force can be discretized in time for the B-grid version of MOM.

12.2.1 Explicit temporal discretization with a leap frog

Consider now just the linear part of the inertial oscillation equation, where advection is dropped

$$(\partial_t + f \hat{\mathbf{z}} \wedge) \mathbf{u} = 0. \quad (12.8)$$

Following the time integration discussions in O'Brien (1986) and Bryan (1989) (see also Section 2.3 of Durrant (1999)), introduce the complex velocity

$$w = u + i v \quad (12.9)$$

where $i = \sqrt{-1}$ and w should not be confused with the vertical velocity component. In terms of w , equation (12.8) takes the form

$$\partial_t w = -i f w \quad (12.10)$$

which has an oscillatory solution

$$w = w_o e^{i f t} \quad (12.11)$$

with period

$$T_{\text{inertial}} = 2\pi/f. \quad (12.12)$$

Time discretizing equation (12.10) with a centered leap-frog scheme leads to

$$w(\tau + \Delta\tau) = w(\tau - \Delta\tau) - i \lambda w(\tau) \quad (12.13)$$

with

$$\lambda = 2 f \Delta\tau \quad (12.14)$$

a dimensionless number. We can write the finite difference solution in terms of an amplification factor

$$w(\tau + \Delta\tau) = G w(\tau). \quad (12.15)$$

Substituting this *ansatz* into equation (12.13) leads to the quadratic equation

$$G^2 + i \lambda G - 1 = 0 \quad (12.16)$$

whose solution is

$$G = \frac{-i \lambda \pm \sqrt{-\lambda^2 + 4}}{2}. \quad (12.17)$$

If

$$\lambda/2 = f \Delta\tau < 1, \quad (12.18)$$

then $|G| = 1$, which means the two finite difference solutions are neutral and stable. One root is an unphysical mode, known as the leap-frog *computational mode*, and the other corresponds to the physical solution. If $\lambda > 2$ then $|G| > 1$ which means both roots are unstable. Hence, stability requires a time step satisfying

$$\Delta\tau < f^{-1}. \quad (12.19)$$

¹It is practical to compute the Coriolis Force implicitly in time when time stepping the single-layer shallow water equations as in (Adcroft et al., 1999).

That is,

$$\frac{T_{\text{inertial}}}{\Delta\tau} = \frac{2\pi}{f\Delta\tau} > 2\pi, \quad (12.20)$$

meaning the leap-frog scheme remains stable if there are at least 2π time steps per inertial period. At the North Pole, this constraint means

$$\Delta\tau < 1.9 \text{ hours}. \quad (12.21)$$

For the baroclinic part of the model algorithm, $\Delta\tau < 1.9 \text{ hours}$ can be the limiting time step for coarse resolution global models, thus motivating an alternative approach discussed in Section 12.2.2.

12.2.2 Semi-implicit time discretization with a leap frog

To overcome the time-step constraint (12.21) on the baroclinic time step, we now consider a semi-implicit time stepping scheme within the leap-frog portion of the baroclinic algorithm. As with any implicit approach, stability can be enhanced relative to explicit schemes. The price to pay is dissipation of the inertial motions. See section 2.3 of [Durrán \(1999\)](#) for discussion.

A semi-implicit treatment of the Coriolis force, within a leap-frog scheme, leads to

$$w(\tau + \Delta\tau) - w(\tau - \Delta\tau) = -i\lambda[(1 - \gamma)w(\tau - \Delta\tau) + \gamma w(\tau + \Delta\tau)] \quad (12.22)$$

where

$$0 \leq \gamma \leq 1 \quad (12.23)$$

is a dimensionless number whose value is set according to stability considerations. We can write $w(\tau + \Delta\tau) = Gw(\tau - \Delta\tau)$, with the semi-implicit scheme yielding the amplification factor

$$G = \frac{1 - i\lambda(1 - \gamma)}{1 + i\lambda\gamma}. \quad (12.24)$$

The squared modulus $|G|^2$ is used to determine conditions for stability

$$|G|^2 = \frac{[1 - \gamma\lambda^2(1 - \gamma)]^2 + \lambda^2}{[1 + (\gamma\lambda)^2]^2}. \quad (12.25)$$

For $\gamma = 0$, $|G| > 1$ which leads to an unstable scheme. For $\gamma = 1/2$, $|G| = 1$ and so the scheme is neutral. With $1/2 < \gamma \leq 1$, $|G| < 1$, and so the scheme is unconditionally stable. Hence, we arrive at the stability range for the semi-implicit parameter

$$1/2 \leq \gamma \leq 1, \quad (12.26)$$

with $\gamma = 1$ yielding the most stable scheme. Section 2.3.2 of [Durrán \(1999\)](#) details the impact on the phase and amplitude of inertial waves depending on the value of γ . That analysis shows that $\gamma = 1/2$ is the most accurate, with zero amplitude error and favorable phase errors relative to other methods.

12.2.3 Semi-implicit time discretization with a forward time step

As discussed in Section 12.8.3 of [Griffies \(2004\)](#) (see also page 51 of [Durrán \(1999\)](#)), the Coriolis force with a forward time step is unstable, and so an alternative must be considered. We apply here the semi-implicit approach from Section 12.2.2 with a forward time step rather than the leap frog. Here, we consider

$$w(\tau + \Delta\tau) - w(\tau) = -i\Lambda[(1 - \gamma)w(\tau) + \gamma w(\tau + \Delta\tau)] \quad (12.27)$$

where again $0 \leq \gamma \leq 1$ is a dimensionless number whose value is set according to stability considerations. The dimensionless parameter Λ is given by

$$\Lambda = f\Delta\tau. \quad (12.28)$$

Note the factor of 2 needed for the leap frog scheme (equation (12.14)) is now absent for the forward scheme. All of the analysis in Section 12.2.2 follows through, with the factor of 2 the only distinction.

12.2.4 Discretization for the B-grid MOM

We now detail the treatment in MOM when employing the B-grid. Both an explicit and semi-implicit treatment of the Coriolis force in the baroclinic equations are available when using leap frog tendencies. However, the semi-implicit treatment is required when using the forward tendencies. For both cases, the semi-implicit piece is handled at the end of a baroclinic time step, even after the implicit treatment of vertical mixing. The logic used in the code can be a bit confusing, so it is useful to expose some details here.

12.2.4.1 Algorithm in the code

Let us separate that portion of the Coriolis force proportional to the dimensionless parameter γ (see equation (12.23)) from the portion independent of γ in order to ease coding for the case with a fully explicit Coriolis force. We also expose the thickness and density weighting used in MOM. Since velocity is updated first as the density and thickness weighted velocity, it is useful to introduce a shorthand

$$\tilde{\mathbf{u}} \equiv (\rho \, dz) \mathbf{u}. \quad (12.29)$$

We consider now three cases for handling the Coriolis force.

- An explicit treatment of the Coriolis force with the leap frog takes the form

$$-f \hat{\mathbf{z}} \wedge \tilde{\mathbf{u}} \rightarrow -f \hat{\mathbf{z}} \wedge \tilde{\mathbf{u}}(\tau), \quad (12.30)$$

- whereas a semi-implicit Coriolis force with the leap frog is

$$\begin{aligned} -f \hat{\mathbf{z}} \wedge \tilde{\mathbf{u}} &\rightarrow -f \hat{\mathbf{z}} \wedge [(1 - \gamma) \tilde{\mathbf{u}}(\tau - \Delta\tau) + \gamma \tilde{\mathbf{u}}(\tau + \Delta\tau)] \\ &= -f \hat{\mathbf{z}} \wedge \tilde{\mathbf{u}}(\tau - \Delta\tau) - f \gamma \hat{\mathbf{z}} \wedge [\tilde{\mathbf{u}}(\tau + \Delta\tau) - \tilde{\mathbf{u}}(\tau - \Delta\tau)], \end{aligned} \quad (12.31)$$

- and a semi-implicit Coriolis force with a forward time step is

$$\begin{aligned} -f \hat{\mathbf{z}} \wedge \tilde{\mathbf{u}} &\rightarrow -f \hat{\mathbf{z}} \wedge [(1 - \gamma) \tilde{\mathbf{u}}(\tau) + \gamma \tilde{\mathbf{u}}(\tau + \Delta\tau)] \\ &= -f \hat{\mathbf{z}} \wedge \tilde{\mathbf{u}}(\tau) - f \gamma \hat{\mathbf{z}} \wedge [\tilde{\mathbf{u}}(\tau + \Delta\tau) - \tilde{\mathbf{u}}(\tau)]. \end{aligned} \quad (12.32)$$

We now consider the remaining terms in the equations of motion. As stated earlier, when treating the Coriolis force with an implicit piece (i.e., with $\gamma > 0$), this is handled last. We write those accelerations independent of γ in the form

$$\delta_\tau \tilde{\mathbf{u}}^* = \mathbf{F} \quad (12.33)$$

where \mathbf{F} includes the thickness weighted and density weighted accelerations from velocity self-advection, the horizontal pressure gradient force, friction force (both explicit and implicit), as well as that piece of the Coriolis force independent of γ . If the Coriolis force is computed explicitly, then \mathbf{F} is the full time tendency for the baroclinic velocity. For the semi-implicit treatment, we require those contributions proportional to γ . For the leap frog, this leads to

$$\tilde{\mathbf{u}}(\tau + \Delta\tau) = \tilde{\mathbf{u}}(\tau - \Delta\tau) + 2\Delta\tau \delta_\tau \tilde{\mathbf{u}}^* - \lambda \gamma \hat{\mathbf{z}} \wedge [\tilde{\mathbf{u}}(\tau + \Delta\tau) - \tilde{\mathbf{u}}(\tau - \Delta\tau)] \quad (12.34)$$

where again $\lambda = 2f \Delta\tau$. Writing out the components leads to

$$\tilde{u}(\tau + \Delta\tau) = \tilde{u}(\tau - \Delta\tau) + 2\Delta\tau \delta_\tau \tilde{u}^* + \lambda \gamma [\tilde{v}(\tau + \Delta\tau) - \tilde{v}(\tau - \Delta\tau)] \quad (12.35)$$

$$\tilde{v}(\tau + \Delta\tau) = \tilde{v}(\tau - \Delta\tau) + 2\Delta\tau \delta_\tau \tilde{v}^* - \lambda \gamma [\tilde{u}(\tau + \Delta\tau) - \tilde{u}(\tau - \Delta\tau)], \quad (12.36)$$

and solving for $\tilde{\mathbf{u}}(\tau + \Delta\tau)$ renders

$$\tilde{u}(\tau + \Delta\tau) = \tilde{u}(\tau - \Delta\tau) + 2\Delta\tau \left(\frac{\delta_\tau \tilde{u}^* + \lambda \gamma \delta_\tau \tilde{v}^*}{1 + (\lambda \gamma)^2} \right) \quad (12.37)$$

$$\tilde{v}(\tau + \Delta\tau) = \tilde{v}(\tau - \Delta\tau) + 2\Delta\tau \left(\frac{\delta_\tau \tilde{v}^* - \lambda \gamma \delta_\tau \tilde{u}^*}{1 + (\lambda \gamma)^2} \right). \quad (12.38)$$

The forward time stepping scheme is handled analogously, which leads to the update for the two components

$$\tilde{u}(\tau + \Delta\tau) = \tilde{u}(\tau) + \Delta\tau \left(\frac{\delta_\tau \tilde{u}^* + \Lambda \gamma \delta_\tau \tilde{v}^*}{1 + (\Lambda \gamma)^2} \right) \quad (12.39)$$

$$\tilde{v}(\tau + \Delta\tau) = \tilde{v}(\tau) + \Delta\tau \left(\frac{\delta_\tau \tilde{u}^* - \Lambda \gamma \delta_\tau \tilde{u}^*}{1 + (\Lambda \gamma)^2} \right), \quad (12.40)$$

where again $\Lambda = f \Delta\tau$.

12.2.4.2 Namelist parameters

In the code,

$$\Delta\tau = dtuv \quad (12.41)$$

is the baroclinic time step, and

$$\gamma = acor \quad (12.42)$$

is a namelist parameter setting the level of implicit treatment for the Coriolis force. The method for discretizing the Coriolis force in the baroclinic part of the model is set according to the value of *acor*, with

$$acor = 0 \Rightarrow \text{explicit Coriolis: only stable for leap frog} \quad (12.43)$$

$$1/2 \leq acor \leq 1 \Rightarrow \text{semi-implicit Coriolis: required if using forward step.} \quad (12.44)$$

The vertically integrated part of the model algorithm typically uses a time step much smaller than f^{-1} . Hence, it is not necessary to discretize the Coriolis force semi-implicitly when time stepping the vertically integrated equations with a leap frog algorithm. However, when using the predictor-corrector described in Section 12.7 of [Griffies \(2004\)](#), $1/2 \leq \gamma \leq 1$ is required for stability, and we choose $\gamma = 1/2$.

12.2.5 Energy analysis

In the continuum, the Coriolis force does no work on a fluid parcel since it is always directed orthogonal to the flow direction

$$\mathbf{v} \cdot f \hat{\mathbf{z}} \wedge \mathbf{u} = 0. \quad (12.45)$$

This property is respected on the B-grid when we discretize the Coriolis force explicitly in time

$$\mathbf{v}(\tau) \cdot f \hat{\mathbf{z}} \wedge \mathbf{u}(\tau) = 0. \quad (12.46)$$

However, the semi-implicit treatment does not respect this property since in general the product

$$\mathbf{v}(\tau) \cdot f \hat{\mathbf{z}} \wedge [(1 - \gamma)\mathbf{u}(\tau - \Delta\tau) + \gamma\mathbf{u}(\tau + \Delta\tau)] \quad (12.47)$$

does not vanish unless the flow is in time independent steady state.

12.3 Time stepping for the C-grid version of MOM

As stated in Section 12.1.2, temporal stability is maintained using a forward time step on the C-grid if we discretize the Coriolis force following the approach used for the advection of momentum (Section 11.2.3), in which a third order Adams-Bashforth method is used ([Durrant, 1999](#)). The third order Adams-Bashforth method requires the Coriolis force at time steps τ , $\tau - 1$, and $\tau - 2$, thus increasing memory requirements. Spatial discretization is detailed in Section 12.1.2.

TIME-IMPLICIT TREATMENT OF VERTICAL MIXING AND BOTTOM DRAG

Contents

13.1	General form of discrete vertical diffusion	184
13.2	Discretization of vertical fluxes	184
13.3	A generic form: Part A	185
13.3.1	Surface cells	185
13.3.2	Interior cells	185
13.3.3	Bottom cells	185
13.3.4	Form appropriate for Numerical Recipes	186
13.4	A generic form with implicit bottom drag	186
13.4.1	Surface cells	187
13.4.2	Interior cells	187
13.4.3	Bottom cells	187
13.4.4	Form appropriate for Numerical Recipes	187

The purpose of this chapter is to detail the method used to time step vertical subgrid scale processes, including bottom drag, using an implicit time stepping method. The material here is based on Section 9.5 of the MOM4.0 Guide (Griffies et al., 2004). There are some novel features discussed here arising from the possibility of including bottom drag implicitly in MOM, which is useful when employing large bottom drag coefficients.

When the MOM namelist `aidif` is set to unity, vertical mixing of momentum and tracers is time stepped implicitly. When `aidif = 0.0`, vertical mixing is time stepped explicitly. Intermediate values give a semi-implicit treatment, although at present MOM does not support semi-implicit treatments. An implicit treatment of vertical mixing allows unrestrained values of the vertical mixing coefficients. An explicit treatment, especially with fine vertical grid resolution, places an unreasonable limitation on the size of the time step. The use of fine vertical resolution with sophisticated mixed layer and/or neutral physics schemes has prompted the near universal implicit treatment of vertical mixing in ocean climate models.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_param/mixing/ocean_vert_mix.F90
ocean_core/ocean_util.F90
ocean_core/ocean_bbc.F90
```

13.1 General form of discrete vertical diffusion

We can write the vertical diffusion equation in the discrete form

$$\partial_t (\phi \rho dz)_k = -(J_{k-1}^z - J_k^z), \quad (13.1)$$

where J_k^z is the vertical SGS flux entering cell k through the bottom face of the cell, and J_{k-1}^z is the vertical SGS flux leaving cell k through its top face. The field ϕ can be either a tracer concentration or a horizontal velocity component. For an implicit treatment of either vertical diffusion (for tracers) or vertical friction (for velocity), we have time stepped ϕ using all time explicit pieces, and thus produced a field $\phi^*(\tau + 1)$, which is the updated field sans the time implicit contributions. So for the purpose of formulating the implicit time stepping portion of the vertical physics, we write the time discrete vertical diffusion equation

$$(\phi \rho dz)_k(\tau + 1) = (\rho dz \phi)_k^*(\tau + 1) - \Delta\tau \text{aif} (J_{k-1}^z - J_k^z), \quad (13.2)$$

where we assumed the preferred MOM forward time stepping scheme¹, and exposed the dimensionless time-implicit factor aif . Again, for implicit time stepping, $\text{aif} = 1.0$, which is the general case for a simulation with nontrivial vertical physics. The mass per area of a grid cell is updated prior to the tracer concentration or velocity components, thus allowing us to divide equation (13.2) by ρdz at time $\tau + 1$, yielding

$$\phi_k(\tau + 1) = \phi_k^*(\tau + 1) - \Gamma_k (J_{k-1}^z - J_k^z), \quad (13.3)$$

with

$$\Gamma_k = \frac{\text{aif} \Delta\tau}{(\rho dz)(\tau + 1)}. \quad (13.4)$$

13.2 Discretization of vertical fluxes

The vertical flux J_k^z is located at the bottom of the k^{th} tracer or velocity cell. A positive value for J_k^z leads to an increase in $\phi_k(\tau + 1)$. Away from surface and bottom boundaries, we assume that this flux takes the downgradient form

$$J_k^z = -\rho_o \kappa_k \left(\frac{\phi_k(\tau + 1) - \phi_{k+1}(\tau + 1)}{dzwt_k} \right). \quad (13.5)$$

The factors of ϕ are evaluated at time $\tau + 1$ because of the implicit treatment. The vertical mixing coefficient κ_k has a general space-time dependence set by a vertical mixing scheme. As for the flux itself, the diffusivity κ_k is situated at the bottom of the tracer or velocity cell, depending on whether ϕ is a tracer field or velocity component. The factor of ρ_o is needed for dimensional consistency, and by our assumption that κ is a kinematic viscosity or diffusivity. The array $dzwt_k$ represents the vertical distance between tracer points at time τ . For vertical mixing of velocity, $dzwt$ becomes the distance between velocity points $dzwu$.

At the ocean surface, the vertical flux is given by the surface boundary condition $sflux$ placed on the velocity or tracer. For a tracer,

$$J_{k=0}^z = -stf, \quad (13.6)$$

with stf MOM's surface tracer flux array with units of velocity times density times tracer concentration. The minus sign arises from the MOM convention that associates a positive stf with an increase in tracer within the $k = 1$ cell. In contrast, the present discussion assumes a convention for the flux J^z whereby a positive $J_{k=0}^z$ is associated with a decrease in tracer within the $k = 1$ cell. For velocity,

$$J_{k=0}^z = -smf, \quad (13.7)$$

with smf the surface momentum flux with units of density times squared velocity. At the ocean bottom, a similar condition leads to

$$J_{k=kmt}^z = -btf \quad (13.8)$$

¹For the leap-frog scheme, the $\Delta\tau$ factor goes to $2\Delta\tau$.

for bottom tracer fluxes, and

$$J_{k=\text{kmt}}^z = -\text{bmf} \quad (13.9)$$

for bottom momentum fluxes. The minus signs again represent a difference in convention between MOM and the present discussion. In MOM, a negative btf represents the passage of tracer from solid rock into the ocean domain, as in geothermal heating. For velocity, a positive bmf represents a drag on the ocean momentum field due to SGS interactions with the solid earth.

13.3 A generic form: Part A

To develop the solution algorithm, it is necessary to put the vertical diffusion equation into a generic form. For this purpose, let us consider in sequence the equation for surface cells $k = 1$, interior cells with $k > 1$, and bottom cells with $k = \text{kmt}$.

13.3.1 Surface cells

For surface cells with $k = 1$ we have

$$\begin{aligned} \phi_k^*(\tau+1) &= \phi_k(\tau+1) + \Gamma_k(\tau)(J_{k-1}^z - J_k^z) \\ &= \phi_k(\tau+1) - \Gamma_k(\tau)(\text{stf} + J_k^z) \\ &= \phi_k(\tau+1) - \Gamma_k(\tau)\text{stf} + \Gamma_k(\tau)\rho_o\kappa_k \left(\frac{\phi_k(\tau+1) - \phi_{k+1}(\tau+1)}{\text{dzwt}_k} \right), \end{aligned} \quad (13.10)$$

which leads to

$$\phi_k^*(\tau+1) + \Gamma_k(\tau)\text{stf} = \phi_k(\tau+1) \left(1 + \frac{\Gamma_k(\tau)\rho_o\kappa_k}{\text{dzwt}_k} \right) - \phi_{k+1}(\tau+1) \left(\frac{\Gamma_k(\tau)\rho_o\kappa_k}{\text{dzwt}_k} \right). \quad (13.11)$$

For velocity mixing, stf becomes smf, and dzwt becomes dzwu.

13.3.2 Interior cells

For interior cells,

$$\begin{aligned} \phi_k^*(\tau+1) &= \phi_k(\tau+1) + \Gamma_k(\tau)(J_{k-1}^z - J_k^z) \\ &= \phi_k(\tau+1) - \Gamma_k(\tau)\rho_o\kappa_{k-1} \left(\frac{\phi_{k-1}(\tau+1) - \phi_k(\tau+1)}{\text{dzwt}_{k-1}} \right) \\ &\quad + \Gamma_k(\tau)\rho_o\kappa_k \left(\frac{\phi_k(\tau+1) - \phi_{k+1}(\tau+1)}{\text{dzwt}_k} \right) \end{aligned} \quad (13.12)$$

which leads to

$$\begin{aligned} \phi_k^*(\tau+1) &= \phi_k(\tau+1) \left(1 + \frac{\Gamma_k(\tau)\rho_o\kappa_{k-1}}{\text{dzwt}_{k-1}} + \frac{\Gamma_k(\tau)\rho_o\kappa_k}{\text{dzwt}_k} \right) \\ &\quad - \phi_{k-1}(\tau+1) \left(\frac{\Gamma_k(\tau)\rho_o\kappa_{k-1}}{\text{dzwt}_{k-1}} \right) - \phi_{k+1}(\tau+1) \left(\frac{\Gamma_k(\tau)\rho_o\kappa_k}{\text{dzwt}_k} \right). \end{aligned} \quad (13.13)$$

13.3.3 Bottom cells

Bottom cells with $k = \text{kmt}(i, j)$ have

$$\begin{aligned} \phi_k^*(\tau+1) &= \phi_k(\tau+1) + \Gamma_k(\tau)(J_{k-1}^z - J_k^z) \\ &= \phi_k(\tau+1) + \Gamma_k(\tau)(J_{k-1}^z + \text{btf}) \\ &= \phi_k(\tau+1) + \Gamma_k(\tau)\text{btf} - \Gamma_k(\tau)\rho_o\kappa_{k-1} \left(\frac{\phi_{k-1}(\tau+1) - \phi_k(\tau+1)}{\text{dzwt}_{k-1}} \right), \end{aligned} \quad (13.14)$$

which leads to

$$\phi_k^*(\tau+1) - \Gamma_k(\tau) \text{btf} = \phi_k(\tau+1) \left(1 + \frac{\Gamma_k(\tau) \rho_o \kappa_{k-1}}{\text{dzwt}_{k-1}} \right) - \phi_{k-1}(\tau+1) \left(\frac{\Gamma_k(\tau) \rho_o \kappa_{k-1}}{\text{dzwt}_{k-1}} \right). \quad (13.15)$$

13.3.4 Form appropriate for Numerical Recipes

Introducing the notation

$$A_k = \begin{cases} -\Gamma_k(\tau) \rho_o \kappa_{k-1} / \text{dzwt}_{k-1} & \text{if } k > 1 \\ 0 & \text{if } k = 1 \end{cases} \quad (13.16)$$

$$C_k = \begin{cases} -\Gamma_k(\tau) \rho_o \kappa_k / \text{dzwt}_k & \text{if } k < \text{km} \\ 0 & \text{if } k = \text{km} \end{cases} \quad (13.17)$$

$$B_k = 1 - A_k - C_k \quad (13.18)$$

$$\Phi_k^* = \begin{cases} \phi_k^*(\tau+1) + \Gamma_k(\tau) \text{stf} & \text{if } k = 1 \\ \phi_k^*(\tau+1) & \text{if } 1 < k < \text{km} \\ \phi_k^*(\tau+1) - \Gamma_k(\tau) \text{btf} & \text{if } k = \text{km} \end{cases} \quad (13.19)$$

renders

$$\Phi_k^* = A_k \phi_{k-1}(\tau+1) + B_k \phi_k(\tau+1) + C_k \phi_{k+1}(\tau+1). \quad (13.20)$$

The solution is arrived at by performing a decomposition and forward substitution. The details are taken from pages 42 and 43 of [Press et al. \(1992\)](#).

13.4 A generic form with implicit bottom drag

We deviate from the previous approach to present here the formulation assuming the bottom boundary fluxes are computed implicitly. Such is important for the case of a bottom drag

$$\mathbf{J}_{k=\text{km}}^z = -\rho_o C_d \mathbf{u} \sqrt{\mathbf{u}_{res}^2 + \mathbf{u}^2}, \quad (13.21)$$

where a large bottom drag coefficient C_d , or large residual velocity \mathbf{u}_{res} require a time implicit solution method. For the global one-degree class of models typically run at GFDL, $C_d > 0.002$ generally requires an implicit treatment of bottom drag. Implicit bottom drag is enabled in MOM by setting the appropriate namelist logical inside *ocean_bbc.nml*.

To time step bottom drag implicitly requires a nonlinear solver. Rather than take that route, we take the simpler approximate approach, also employed when the diffusivity or viscosity is a nonlinear function of the flow. That is, we time discretize the bottom drag as

$$\mathbf{J}_{k=\text{km}}^z = -\rho_o C_d \mathbf{u}(\tau+1) \sqrt{\mathbf{u}_{res}^2 + \mathbf{u}^2(\tau)}. \quad (13.22)$$

Hence, for the purpose of formulating the time implicit algorithm, we write the bottom drag

$$\mathbf{J}_{k=\text{km}}^z = -\gamma \mathbf{u}(\tau+1), \quad (13.23)$$

where

$$\gamma = \rho_o C_d \sqrt{\mathbf{u}_{res}^2 + \mathbf{u}^2(\tau)} \quad (13.24)$$

is a nonlinear function of velocity at time τ . We can now modify the steps detailed in Section 13.3, using the nonlinear bottom drag (13.23). As this situation arises in practice for the momentum equation, we employ velocity cell labels where appropriate.

13.4.1 Surface cells

For surface cells with $k = 1$ we have

$$\begin{aligned}\phi_k^*(\tau + 1) &= \phi_k(\tau + 1) + \Gamma_k(\tau)(J_{k-1}^z - J_k^z) \\ &= \phi_k(\tau + 1) - \Gamma_k(\tau)(\text{smf} + J_k^z) \\ &= \phi_k(\tau + 1) - \Gamma_k(\tau)\text{smf} + \Gamma_k(\tau)\rho_o \kappa_k \left(\frac{\phi_k(\tau + 1) - \phi_{k+1}(\tau + 1)}{\text{dzwu}_k} \right),\end{aligned}\quad (13.25)$$

which leads to

$$\phi_k^*(\tau + 1) + \Gamma_k(\tau)\text{smf} = \phi_k(\tau + 1) \left(1 + \frac{\Gamma_k(\tau)\rho_o \kappa_k}{\text{dzwu}_k} \right) - \phi_{k+1}(\tau + 1) \left(\frac{\Gamma_k(\tau)\rho_o \kappa_k}{\text{dzwu}_k} \right). \quad (13.26)$$

13.4.2 Interior cells

For interior cells,

$$\begin{aligned}\phi_k^*(\tau + 1) &= \phi_k(\tau + 1) + \Gamma_k(\tau)(J_{k-1}^z - J_k^z) \\ &= \phi_k(\tau + 1) - \Gamma_k(\tau)\rho_o \kappa_{k-1} \left(\frac{\phi_{k-1}(\tau + 1) - \phi_k(\tau + 1)}{\text{dzwu}_{k-1}} \right) \\ &\quad + \Gamma_k(\tau)\rho_o \kappa_k \left(\frac{\phi_k(\tau + 1) - \phi_{k+1}(\tau + 1)}{\text{dzwu}_k} \right)\end{aligned}\quad (13.27)$$

which leads to

$$\begin{aligned}\phi_k^*(\tau + 1) &= \phi_k(\tau + 1) \left(1 + \frac{\Gamma_k(\tau)\rho_o \kappa_{k-1}}{\text{dzwu}_{k-1}} + \frac{\Gamma_k(\tau)\rho_o \kappa_k}{\text{dzwu}_k} \right) \\ &\quad - \phi_{k-1}(\tau + 1) \left(\frac{\Gamma_k(\tau)\rho_o \kappa_{k-1}}{\text{dzwu}_{k-1}} \right) - \phi_{k+1}(\tau + 1) \left(\frac{\Gamma_k(\tau)\rho_o \kappa_k}{\text{dzwu}_k} \right).\end{aligned}\quad (13.28)$$

13.4.3 Bottom cells

Bottom cells with $k = \text{kmu}(i, j)$ have

$$\begin{aligned}\phi_k^*(\tau + 1) &= \phi_k(\tau + 1) + \Gamma_k(\tau)(J_{k-1}^z - J_k^z) \\ &= \phi_k(\tau + 1) + \Gamma_k(\tau)(J_{k-1}^z + \text{bmf}) \\ &= \phi_k(\tau + 1) + \Gamma_k(\tau)\gamma \phi_k(\tau + 1) - \Gamma_k(\tau)\rho_o \kappa_{k-1} \left(\frac{\phi_{k-1}(\tau + 1) - \phi_k(\tau + 1)}{\text{dzwu}_{k-1}} \right),\end{aligned}\quad (13.29)$$

which leads to

$$\phi_k^*(\tau + 1) = \phi_k(\tau + 1) \left(1 + \gamma \Gamma_k(\tau) + \frac{\Gamma_k(\tau)\rho_o \kappa_{k-1}}{\text{dzwu}_{k-1}} \right) - \phi_{k-1}(\tau + 1) \left(\frac{\Gamma_k(\tau)\rho_o \kappa_{k-1}}{\text{dzwt}_{k-1}} \right). \quad (13.30)$$

13.4.4 Form appropriate for Numerical Recipes

Introducing the notation

$$A_k = \begin{cases} -\Gamma_k(\tau)\rho_o \kappa_{k-1}/\text{dzwu}_{k-1} & \text{if } k > 1 \\ 0 & \text{if } k = 1 \end{cases} \quad (13.31)$$

$$C_k = \begin{cases} -\Gamma_k(\tau)\rho_o \kappa_k/\text{dzwu}_k & \text{if } k < \text{kmu} \\ 0 & \text{if } k = \text{kmu} \end{cases} \quad (13.32)$$

$$B_k = \begin{cases} 1 - A_k - C_k & \text{if } k < \text{kmu} \\ 1 + \gamma \Gamma_k(\tau) - A_k - C_k & \text{if } k = \text{kmu} \end{cases} \quad (13.33)$$

$$\Phi_k^* = \begin{cases} \phi_k^*(\tau + 1) + \Gamma_k(\tau)\text{smf} & \text{if } k = 1 \\ \phi_k^*(\tau + 1) & \text{if } 1 < k \leq \text{kmu} \end{cases} \quad (13.34)$$

renders

$$\Phi_k^* = A_k \phi_{k-1}(\tau + 1) + B_k \phi_k(\tau + 1) + C_k \phi_{k+1}(\tau + 1). \quad (13.35)$$

The solution is arrived at by performing a decomposition and forward substitution. The details are taken from pages 42 and 43 of [Press et al. \(1992\)](#).

MECHANICAL ENERGY CONVERSIONS AND ADVECTIVE MASS TRANSPORT

Contents

14.1	Basic considerations	190
14.2	Energetic conversions in the continuum	191
14.2.1	Pressure work conversions in Boussinesq fluids	191
14.2.2	Pressure work conversions in non-Boussinesq fluids	192
14.2.3	Boussinesq kinetic energy advection conversion	192
14.2.4	Non-Boussinesq kinetic energy advection conversion	193
14.3	How we make use of energetic conversions	193
14.3.1	Conservation versus accuracy	193
14.3.2	Energy conservation, consistency, and conversion	194
14.3.3	A caveat regarding the tripolar grid in MOM	194
14.4	Thickness weighted volume and mass budgets	194
14.5	Thickness and mass per area for the momentum	195
14.5.1	B-grid momentum	196
14.5.2	C-grid momentum	196
14.6	B-grid Boussinesq pressure work conversions	196
14.6.1	The vertically integrated term \mathcal{P}_1	198
14.6.2	Advection velocity components for tracers	198
14.6.3	Divergence operator for surface height evolution	199
14.6.4	Completing the manipulations for \mathcal{P}_2	200
14.6.4.1	Energetic approach	200
14.6.4.2	Finite volume approach	201
14.6.5	The geopotential gradient term \mathcal{P}_3	202
14.6.6	Summary for the Boussinesq pressure conversion	202
14.7	C-grid Boussinesq pressure work conversions	203
14.7.1	Force from the horizontal pressure gradient	203
14.7.2	The vertically integrated term \mathcal{P}_1	204
14.7.3	Advection velocity components for tracers	205
14.7.4	Divergence operator for surface height evolution	205
14.7.5	The geopotential gradient term \mathcal{P}_3	205
14.7.6	Summary for the Boussinesq pressure conversion	206
14.8	B-grid non-Boussinesq pressure work conversions	206

14.8.1	The vertically integrated term \mathcal{P}_1	207
14.8.2	Defining the advective mass transport	207
14.8.3	Completing the manipulations for \mathcal{P}_2	208
14.8.3.1	Energetic approach	209
14.8.3.2	Finite volume approach	209
14.8.4	The pressure gradient term \mathcal{P}_3	209
14.8.5	Summary for the non-Boussinesq pressure conversion	210
14.9	C-grid non-Boussinesq pressure work conversions	210
14.9.1	The vertically integrated term \mathcal{P}_1	211
14.9.2	Defining the advective mass transport	211
14.9.3	The pressure gradient term \mathcal{P}_3	212
14.9.4	Summary for the non-Boussinesq pressure conversion	212
14.10	Effective Coriolis force and mechanical energy	212
14.10.1	B-grid	213
14.10.2	C-grid	213
14.10.3	Comments	214
14.11	B-grid kinetic energy advection	214
14.11.1	B-grid momentum equation contribution from advection	215
14.11.2	Horizontal convergence	215
14.11.3	Diagnosing the vertical transport for U-cells	215
14.11.4	Discrete integration by parts on horizontal convergence	216
14.11.5	Discrete integration by parts on the vertical convergence	217
14.11.6	Final result for the Boussinesq case	217
14.11.7	Non-Boussinesq kinetic energy advection	217
14.12	C-grid kinetic energy advection	218
14.12.1	C-grid momentum equation contribution from advection	218
14.12.2	Energetic manipulations not generally useful	219

The purpose of this chapter is to discuss continuum and discrete mechanical energy balances. Maintenance of such balances on the discrete grid have implications for spatial discretization of advective mass transport. Both the B-grid and C-grid discretizations are considered.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_velocity.F90
ocean_core/ocean_advection_velocity.F90
ocean_core/ocean_velocity_advect.F90
ocean_diag/ocean_velocity_diag.F90
```

14.1 Basic considerations

The following are the assumptions made for manipulations of this chapter.

- Choosing to maintain the integrity of certain energetic balances on the B-grid lattice prescribes the form of the discrete advection velocity components located on the sides of tracer cells.
- Second order finite differenced advective fluxes are used for momentum. Tracer fluxes can remain arbitrarily discretized.
- We choose a finite difference computation of the pressure gradient force, as described in Sections 3.1, 3.2, and 3.3. The finite volume method for computing the pressure force, as described in Section 2.8.1, does not lend itself to the results of this chapter.

- Details of the time stepping scheme play a role in determining the form of the energy diagnostics.
- Energy balance diagnostics are important for checking the integrity of certain finite difference algorithms. Consequently, it is useful to provide a careful suite of energy diagnostics for algorithm development purposes.

14.2 Energetic conversions in the continuum

In the continuum, the horizontal momentum equation for a shallow ocean fluid is given by (see chapter 4 of [Griffies, 2004](#))

$$(\rho \mathbf{u})_{,t} + \nabla \cdot (\rho \mathbf{v} \mathbf{u}) + (f + \mathcal{M})(\hat{\mathbf{z}} \wedge \rho \mathbf{u}) = -\nabla_z p + \rho \mathbf{F} \quad (14.1)$$

for the non-Boussinesq case, and

$$(\mathbf{u})_{,t} + \nabla \cdot (\mathbf{v} \mathbf{u}) + (f + \mathcal{M})(\hat{\mathbf{z}} \wedge \mathbf{u}) = -\nabla_z (p/\rho_o) + \mathbf{F} \quad (14.2)$$

for the Boussinesq case. The evolution of horizontal kinetic energy can be found by taking the scalar product of horizontal velocity \mathbf{u} with the momentum equation. When globally integrating the kinetic energy evolution, the forcing terms can be transformed into terms that highlight physically interesting processes. These manipulations identify *conversions* between one form of energy and another. The form of these conversions can be deduced from the momentum equations, boundary conditions, mass or volume conservation, and integration by parts. Maintaining an analog of these energetic conversions on the discrete lattice has been found to be very useful in the development of ocean model algorithms. The reason is that these conversions provide the modeler with a powerful set of diagnostics to test the integrity of the numerics.

There are three forms of energy conversion of interest in MOM. The first involves the pressure gradient term, the second involves the advection term, and the third involves friction. We address only the inviscid terms in this chapter. Part 5 of [Griffies \(2004\)](#) describes how friction dissipates kinetic energy in both the continuous case and for a particular friction algorithm available in MOM.

14.2.1 Pressure work conversions in Boussinesq fluids

Let us first examine how pressure work is converted to other processes in Boussinesq fluids. For this purpose, consider the following identities found using z for the vertical coordinate

$$\begin{aligned} \int dV \mathbf{u} \cdot \nabla p &= \int dV (\mathbf{v} \cdot \nabla p - w p_{,z}) \\ &= \int dV [\nabla \cdot (\mathbf{v} p) - w p_{,z}] \\ &= \int dA_{(\hat{\mathbf{n}})} p (\hat{\mathbf{n}} \cdot \mathbf{v}) + g \int dV w \rho \\ &= \int dA_{(\hat{\mathbf{n}})} p (\hat{\mathbf{n}} \cdot \mathbf{v}) + \int dV \rho d\Phi/dt \end{aligned} \quad (14.3)$$

where $dV = dx dy dz$ is the volume element and $d\Phi/dt$ is the material time derivative of the geopotential $\Phi = gz$. To reach these results required volume conservation for a parcel in the form of the constraint $\nabla \cdot \mathbf{v} = 0$, the hydrostatic relation $p_{,z} = -\rho g$, and the definitions

$$\begin{aligned} g w &= g dz/dt \\ &= d\Phi/dt. \end{aligned} \quad (14.4)$$

Assuming no-normal flow at the solid boundaries leaves only the surface boundary at $z = \eta$ for the surface integral. The surface kinematic boundary condition, and volume conservation, lead to¹

$$dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{v} = -dx dy \nabla \cdot \mathbf{U}, \quad (14.5)$$

¹See Section 3.4 of [Griffies \(2004\)](#) for derivation.

and so

$$\int dV \mathbf{u} \cdot \nabla p = - \int_{z=\eta} dx dy p_a \nabla \cdot \mathbf{U} + \int dV \rho d\Phi/dt. \quad (14.6)$$

In a rigid lid model, the first term vanishes. For the free surface model it represents the work done by atmospheric pressure on the depth integrated flow. The second term is the volume integrated work done by vertical currents against the buoyancy force. In generalized vertical coordinates, the buoyancy term takes the form

$$\begin{aligned} d\Phi/dt &= (\partial_t + \mathbf{u} \cdot \nabla_s + w^{(s)} \partial_z) \Phi \\ &= (\partial_t + \mathbf{u} \cdot \nabla_s) \Phi + g w^{(s)}, \end{aligned} \quad (14.7)$$

where equation (6.72) of [Griffies \(2004\)](#) was used to express the material time derivative in general vertical coordinates. Hence, the pressure conversion becomes

$$\int dV \mathbf{u} \cdot \nabla p = - \int_{z=\eta} dx dy p \nabla \cdot \mathbf{U} + \int dV \rho [(\partial_t + \mathbf{u} \cdot \nabla_s) \Phi + g w^{(s)}]. \quad (14.8)$$

Buoyancy contributions now comprise three terms instead of the one found with z -coordinates. This result reflects the non-orthogonal nature of generalized vertical coordinates.

14.2.2 Pressure work conversions in non-Boussinesq fluids

For non-Boussinesq flows, pressure conversion takes the form

$$\int dV \mathbf{u} \cdot \nabla p = \int_{z=\eta} p \hat{\mathbf{n}} \cdot \mathbf{v} + \int dV (\rho d\Phi/dt - p \nabla \cdot \mathbf{v}). \quad (14.9)$$

The $p \nabla \cdot \mathbf{v}$ term represents pressure work on the changing volume of fluid parcels found in the compressible non-Boussinesq fluid. The boundary condition

$$dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{v} = dx dy (\eta_{,t} - Q_m/\rho) \quad (14.10)$$

is discussed in Section 3.4.3 of [Griffies \(2004\)](#). The generalized vertical coordinate form of equation (14.9) follows similarly to the Boussinesq case, where extra terms arise from expanding the material time derivative.

14.2.3 Boussinesq kinetic energy advection conversion

Just as for the pressure gradient term, the scalar product of the horizontal velocity and the advection of momentum can be converted into alternative forms. To see this conversion in the continuum, write the advection of horizontal velocity in the Boussinesq fluid as

$$\mathbf{A} \equiv -\nabla \cdot (\mathbf{v} \mathbf{u}) - \mathcal{M} \hat{\mathbf{z}} \wedge \mathbf{v}. \quad (14.11)$$

The scalar product of \mathbf{A} with the horizontal currents leads to

$$\begin{aligned} \mathbf{u} \cdot \mathbf{A} &= -\mathbf{u} \cdot \nabla \cdot (\mathbf{v} \mathbf{u}) \\ &= -\nabla \cdot (\mathbf{v} \mathcal{K}), \end{aligned} \quad (14.12)$$

where

$$\mathcal{K} = \frac{\mathbf{u} \cdot \mathbf{u}}{2} \quad (14.13)$$

is the horizontal kinetic energy per mass. Integrating over the volume of the domain, and using the surface and solid wall boundary conditions, leads to

$$\begin{aligned}\mathcal{A} &\equiv \int dV \mathbf{u} \cdot \mathbf{A} \\ &= \int_{z=\eta} dx dy \mathcal{K} \nabla \cdot \mathbf{U}.\end{aligned}\tag{14.14}$$

Consequently, the global integral of kinetic energy advection reduces to a boundary term, which vanishes in the rigid lid model but remains nontrivial in a free surface model.

14.2.4 Non-Boussinesq kinetic energy advection conversion

For the non-Boussinesq fluid, we consider

$$\mathbf{A} \equiv -\nabla \cdot (\rho \mathbf{v} \mathbf{u}) - \mathcal{M} \hat{\mathbf{z}} \wedge \rho \mathbf{v}.\tag{14.15}$$

The scalar product of \mathbf{A} with the horizontal currents leads to

$$\begin{aligned}\mathbf{u} \cdot \mathbf{A} &= -\mathbf{u} \cdot \nabla \cdot (\rho \mathbf{v} \mathbf{u}) \\ &= -\nabla \cdot (\mathbf{v} \mathcal{K}) - \mathcal{K} \nabla \cdot (\rho \mathbf{v}),\end{aligned}\tag{14.16}$$

and integrating over the volume of the domain yields

$$\begin{aligned}\mathcal{A} &\equiv \int dV \mathbf{u} \cdot \mathbf{A} \\ &= - \int_{z=\eta} dA_{(\hat{\mathbf{n}})} \mathcal{K} \rho \hat{\mathbf{n}} \cdot \mathbf{v} - \int dV \mathcal{K} \nabla \cdot (\rho \mathbf{v}) \\ &= - \int_{z=\eta} dx dy \mathcal{K} (\rho \eta_{,t} - Q_m) - \int dV \mathcal{K} \nabla \cdot (\rho \mathbf{v}),\end{aligned}\tag{14.17}$$

where we used the surface boundary condition (14.10) for the last step.

14.3 How we make use of energetic conversions

We offer here some general comments regarding the utility of the energetic methods for deriving numerical discretizations.

14.3.1 Conservation versus accuracy

Accuracy is often a primary consideration for numerical methods. Additionally, ease of analysis and interpretation are also important. The presence of discrete analogs to continuous conservation properties assists in the interpretation of the numerical simulation. Unfortunately, conservation and accuracy are often incompatible.

Traditionally, climate modelers have chosen conservation properties over accuracy. For example, conservation of scalar properties are essential to ensure that mass/volume, heat and salt are conserved over the course of a long climate integration. Another property that certain models claim is conservation of mechanical energy. This claim, however, is unfounded for the space-time discrete equations in all models discussed in Griffies et al. (2000a). All ocean climate models break kinetic energy conservation when discretizing in time. This point is explained below in the discussion of equation (14.18).

14.3.2 Energy conservation, consistency, and conversion

What is often meant by energy conservation statements is the more qualified property whereby certain spatially discrete terms are discretized so they do not alter global kinetic energy in the absence of boundary forcing. Deriving energetically consistent numerical schemes requires some care. In particular, ensuring that pressure work transfers into vertical buoyancy work in the Boussinesq model necessitates a particular form for the discrete advection velocity. We discuss this point in Sections 14.6 and 14.7.

When considering energetic issues using alternative time stepping schemes, one often encounters the situation where certain terms, such as advection, the Coriolis force, and pressure gradients, are evaluated at staggered time steps. Indeed, the preferred method discussed in Chapter 12 of Griffies (2004) and in Chapter 11 in this document staggers the velocity and tracer one-half time step relative to one another, and generally uses non-centred in-time methods for the advection and Coriolis force. Hence, pressure gradients, whose temporal placement is set by density, is off-set in time from momentum advection, the Coriolis force, and friction. These details are important when interpreting energetic balances of a space-time discrete model. Often the more sophisticated the time stepping scheme (e.g., the three-time level Adams-Bashforth method discussed in Chapter 12 of Griffies (2004)), the more difficult it is to maintain energetic consistency and balances.

Energetic consistency is necessary but not sufficient for ensuring the discrete system conserves mechanical energy in the unforced inviscid limit. For example, time stepping according to the leap-frog method, which possesses useful energy consistency properties, precludes mechanical energy conservation. The Robert-Asselin time filter breaks energy conservation in a manner analogous to its corruption of global tracer conservation (see Section 11.2.2 as well as Section 12.5.4 of Griffies (2004)). Furthermore, even without time filtering, the continuum identity

$$2\mathbf{u} \cdot \partial_t \mathbf{u} = \partial_t (\mathbf{u} \cdot \mathbf{u}) \quad (14.18)$$

is generally not satisfied by discrete time stepping schemes. As noted on page 158 of Durran (1999), trapezoidal time differencing allows for this property. Other schemes commonly used do not. As trapezoidal time differencing is semi-implicit and not readily implemented for the primitive equations, it is not considered in the following.

14.3.3 A caveat regarding the tripolar grid in MOM

The tripolar grid (Section 9.3) is routinely used in global simulations with MOM in order to remove the spherical coordinate singularity from the liquid ocean domain. Unfortunately, due to some limitations of the energetic diagnostics, the energy conversion diagnostics are not precisely maintained when using the tripolar grid. The issue is not related to a problem with the prognostic equations that time step the model fields, but rather related to limitations with the diagnostic code. To resolve the diagnostic code requires adding processor updates that have not been deemed important enough to warrant the potential model slowdown.

14.4 Thickness weighted volume and mass budgets

We make use of the thickness weighted volume budgets for the Boussinesq fluid when deriving the discrete energetic balances. The volume budgets are given by equations (10.82), (10.83), and (10.84). We expose them here for completeness

$$(w^{(s)})_{s=s_{k+1}} = \partial_t (dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) - Q_m / \rho_o \quad (14.19)$$

$$(w^{(s)})_{s=s_k} = \partial_t (dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(s)})_{s=s_{k-1}} \quad (14.20)$$

$$0 = \partial_t (dz) - \mathcal{S}^{(V)} dz + \nabla_s \cdot (\mathbf{u} dz) + (w^{(s)})_{s=s_{kbot-1}}. \quad (14.21)$$

We also make use of the thickness weighted mass budgets for the non-Boussinesq case, given by equations (10.106), (10.107), and (10.108). We expose them here for completeness

$$(\rho w^{(s)})_{s=s_{k+1}} = \partial_t(\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) - Q_m \quad (14.22)$$

$$(\rho w^{(s)})_{s=s_k} = \partial_t(\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) + (\rho w^{(s)})_{s=s_{k-1}} \quad (14.23)$$

$$0 = \partial_t(\rho dz) - \mathcal{S}^{(M)} \rho dz + \nabla_s \cdot (\mathbf{u} \rho dz) + (\rho w^{(s)})_{s=s_{kbot-1}}. \quad (14.24)$$

As described in Section 10.8.1, depth based vertical coordinates used in MOM (Section 5.1) allow for the time derivative $\partial_t(dz)$ to be diagnosed from the vertically integrated volume budget. Likewise, the pressure based vertical coordinates (Section 5.2) allow for the time derivative $\partial_t(\rho dz)$ to be diagnosed from the vertically integrated mass budget. These two properties are important to ensure the utility of the Eulerian algorithms employed by MOM.

14.5 Thickness and mass per area for the momentum

We consider here the specification of various thickness or mass per horizontal area required for the B-grid and C-grid.

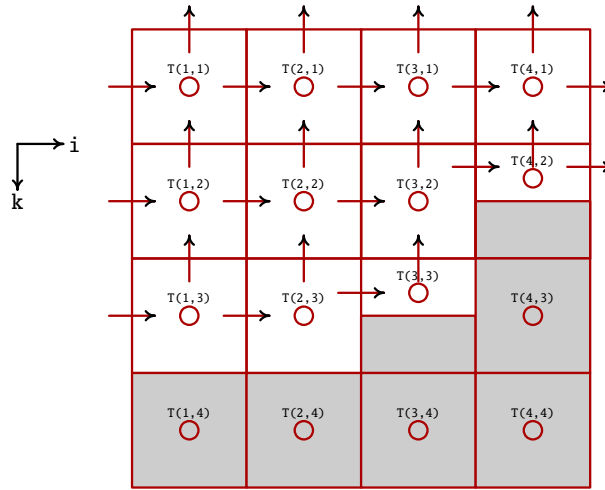


Figure 14.1: Shown here is a 4x4 region of a zonal-vertical domain of tracer cells $T(i, k)$, with ocean cells (unshaded) and land cells (shaded). Note the partial bottom cells in cells $T(3, 3)$, $T(4, 3)$ and $T(4, 2)$. For both the B-grid and C-grids, the advective transport through the zonal tracer cell face is depicted by horizontal arrows, and vertical advective transport is depicted by vertical arrows. On the C-grid, the thickness appearing in the discrete expression for the C-grid zonal momentum per area $(u \rho_o dz)_{i,j}$ is taken as the minimum thickness $\rho_o (dz)_{i,j} = \rho_o \min(dzt_{i,j}, dzt_{i+1,j})$ between the adjacent tracer cells. Likewise, for a non-Boussinesq fluid, the mass per unit area appearing in the discrete expression for $(u \rho dz)_{i,j}$ is given by $(\rho dz)_{i,j} = \min(\rho dzt_{i,j}, \rho dzt_{i+1,j})$. Similar expressions hold for the meridional cell face. The minimum function ensures mass conservation when moving across cell faces where the adjacent cells have distinct thicknesses, as when there is partial cell bottom topography or generalized level coordinates. That is, the minimum function precludes too much mass entering or leaving the thinnest of the adjacent cells. When considering a B-grid, where horizontal velocity components are co-located, it is the least massive of the four surrounding tracer cells that provides the velocity cell mass per horizontal area $(\rho dz)_{i,j} = \min[(\rho dzt)_{i,j}, (\rho dzt)_{i+1,j}, (\rho dzt)_{i,j+1}, (\rho dzt)_{i+1,j+1}]$.

14.5.1 B-grid momentum

The B-grid identifies a distinct control volume for the velocity cell and tracer cell (Figure 9.1). The mass per unit area of the velocity cell is determined by the minimum of the surrounding four tracer cells

$$(\text{rho_dzu})_{i,j} = \min[(\text{rho_dzt})_{i,j}, (\text{rho_dzt})_{i+1,j}, (\text{rho_dzt})_{i,j+1}, (\text{rho_dzt})_{i+1,j+1}], \quad (14.25)$$

with the density factors set to the constant ρ_o when making the Boussinesq approximation. Once the momentum per horizontal area, $\mathbf{u} \text{rho_dzu}$, is updated to a new time step, we divide by $(\text{rho_dzu})_{i,j}$ to diagnose the updated B-grid velocity \mathbf{u} .

The definition (14.25) follows the partial bottom cell considerations of Pacanowski and Gnanadesikan (1998). Griffies et al. (2001) then applied this definition to the surface ocean in an explicit free surface model with a geopotential vertical coordinate, where the top grid cells have a time dependent thickness. Starting from MOM4p1, MOM follows that applies the definition (14.25) throughout the fluid column for all generalized level coordinates. Further discussion is given in the caption to Figure 14.1.

14.5.2 C-grid momentum

On the C-grid, there is no velocity control volume. Instead, there is a separate thickness or mass per unit area associated with each momentum component, which are determined according to

$$(\text{rho_dzte})_{i,j} = \min[(\text{rho_dzt})_{i,j}, (\text{rho_dzt})_{i+1,j}] \quad (14.26)$$

$$(\text{rho_dztn})_{i,j} = \min[(\text{rho_dzt})_{i,j}, (\text{rho_dzt})_{i,j+1}]. \quad (14.27)$$

Motivation for these definitions is provided in Figure 14.1.

14.6 B-grid Boussinesq pressure work conversions

We now consider manipulations of the globally integrated discrete B-grid representation of $\mathbf{u} \cdot \nabla_z p$ for the Boussinesq case. In this section, discrete grid labels are exposed when needed, with many labels suppressed to reduce clutter. Also, the horizontal velocity components are co-located on the northeast corners of the tracer grid, as per the B-grid convention (Figure 9.1). The material in this Section is based on a similar z -coordinate discussion given in Griffies et al. (2004), but it has been generalized to the arbitrary level coordinates available in MOM.

Consider the domain integrated scalar product of

$$\begin{aligned} \mathbf{u} \cdot \nabla_z p &= \mathbf{u} \cdot (\nabla_s p + \rho \nabla_s \Phi) \\ &= \mathbf{u} \cdot \nabla (p_a + p_{\text{surf}}) + \mathbf{u} \cdot (\nabla_s p' + \rho' \nabla_s \Phi). \end{aligned} \quad (14.28)$$

To reach this result we use equation (3.18) with $s = z^*$ or $s = \sigma^{(z)}$, in which case

$$p_{\text{surf}} = g \rho_o \eta \quad \text{when } s = z^* \text{ or } s = \sigma^{(z)} \quad (14.29)$$

is the rapidly fluctuating surface pressure term, and

$$p' = g \int_z^\eta \rho' dz \quad \text{when } s = z^* \text{ or } s = \sigma^{(z)} \quad (14.30)$$

is the slower fluctuating pressure anomaly where $\rho' = \rho - \rho_o$. When $s = z$ is the vertical coordinate, equation (3.17) is used, in which case

$$p_{\text{surf}} = g \rho_{\text{surf}} \eta \quad \text{when } s = z \quad (14.31)$$

and

$$p' = g \int_z^0 \rho' dz \quad \text{when } s = z. \quad (14.32)$$

To determine the proper discrete form of the pressure conversion, recall from Section 3.3.1 the momentum equations for a Boussinesq fluid on a B-grid with just the impacts from pressure acting

$$\partial_t (u \rho_o dzu)^{\text{pressure}} = -dzu \left(\text{FDX_NT}(\text{FAY}(p_a + p_{\text{surf}} + p')) + \text{FAY}[\delta_i \Phi \text{FAX}(\rho')]/dxu_{i,j} \right) \quad (14.33)$$

$$\partial_t (v \rho_o dzu)^{\text{pressure}} = -dzu \left(\text{FDY_ET}(\text{FAX}(p_a + p_{\text{surf}} + p')) + \text{FAX}[\delta_j \Phi \text{FAY}(\rho')]/dyu_{i,j} \right). \quad (14.34)$$

The volume integrated pressure conversion

$$\mathcal{P} = - \int \mathbf{u} \cdot \nabla_z p dV, \quad (14.35)$$

thus takes on the following discrete form on the B-grid

$$\begin{aligned} \mathcal{P} \equiv & - \sum_{i,j,k} dV^{(v)} [u \text{FDX_NT}(\text{FAY}(p_a + p_{\text{surf}})) + v \text{FDY_ET}(\text{FAX}(p_a + p_{\text{surf}}))] \\ & - \sum_{i,j,k} dV^{(v)} [u \text{FDX_NT}(\text{FAY}(p')) + v \text{FDY_ET}(\text{FAX}(p'))] \\ & - \sum_{i,j,k} dV^{(v)} [u \text{FAY}(\text{FAX}(\rho') \delta_i \Phi)/dxu + v \text{FAX}(\text{FAY}(\rho') \delta_j \Phi)/dyu] \end{aligned} \quad (14.36)$$

with

$$dV^{(v)} = dau dzu \quad (14.37)$$

the U-cell volume. The discrete expressions for the pressure gradient are based on the discussion in Section 3.3.1, where the horizontal pressure gradient body force is written for the B-grid. MOM employs the following discrete forward derivative operators

$$\text{FDX_NT}(A) = \frac{A_{i+1} - A_i}{dxu_{i,j}} \quad (14.38)$$

$$\text{FDY_ET}(A) = \frac{A_{j+1} - A_j}{dyu_{i,j}}, \quad (14.39)$$

where the arguments of the derivatives live on the north and east faces, respectively, of a tracer cell. The operators $\delta_i A$ and $\delta_j A$ compute the forward difference

$$\delta_i A = A_{i+1} - A_i \quad (14.40)$$

$$\delta_j A = A_{j+1} - A_j \quad (14.41)$$

of a discrete field. MOM also employs the following forward averaging operators

$$\text{FAX}(A) = \frac{A_{i+1} + A_i}{2} \quad (14.42)$$

$$\text{FAY}(A) = \frac{A_{j+1} + A_j}{2}. \quad (14.43)$$

The first group of terms in equation (14.36) arises from applied pressure and surface geopotential acting on the vertically integrated velocity. The second group represents the constant s lateral pressure gradient taken between cells living on the same discrete k -level. The third group arises from the use of generalized vertical coordinates, where the depth of a k -level is generally a function of horizontal position.

The goal of the remainder of this section is to rearrange the discrete terms appearing in the pressure conversion equation (14.36) to reveal an alternative, and physically sensible, form. In effect, we are performing a discrete integration by parts. The MOM energy conversion diagnostic computes the unmanipulated form of the pressure conversion and the manipulated form, and compares the two results: *left hand side = right hand side?* Except for the caveat noted for the tripolar grid (Section 14.3.3), differences between the two calculations can reveal basic algorithm mistakes.

14.6.1 The vertically integrated term \mathcal{P}_1

The first term in equation (14.36) can be vertically integrated to yield

$$\begin{aligned}\mathcal{P}_1 &\equiv - \sum_{i,j,k} dV^{(v)} [u \text{ FDX_NT}(\text{FAY}(p_a + p_{\text{surf}})) + v \text{ FDY_ET}(\text{FAX}(p_a + p_{\text{surf}}))] \\ &= - \sum_{i,j} \text{dau} [U \text{ FDX_NT}(\text{FAY}(p_a + p_{\text{surf}})) + V \text{ FDY_ET}(\text{FAX}(p_a + p_{\text{surf}}))] \end{aligned} \quad (14.44)$$

where

$$(U, V) = \sum_k dz u(u, v) \quad (14.45)$$

is the vertically integrated horizontal velocity field. The \mathcal{P}_1 term represents the work of applied pressure and geopotential moving vertical columns of fluid.

14.6.2 Advection velocity components for tracers

Focus on the zonal piece of the baroclinic pressure term appearing in equation (14.36), in which

$$\begin{aligned}\mathcal{P}_{2x} &\equiv - \sum_{i,j,k} \text{dau} \, dz u \, u \, \text{FDX_NT}(\text{FAY}(p')) \\ &= - \frac{1}{2} \sum \text{dyu} \, dz u \, u \, \delta_i(p'_j + p'_{j+1}) \\ &= - \sum \text{BAY}(\text{dyu} \, dz u) \, \delta_i p'_j. \end{aligned} \quad (14.46)$$

The boundary terms were dropped since they vanish for either periodic or solid wall conditions. We also introduced the backward meridional average operator

$$\text{BAY}(A) = \frac{A_j + A_{j-1}}{2}. \quad (14.47)$$

Now *define* the zonal thickness weighted advective transport velocity on the eastern face of a tracer cell as

$$\text{uh_et}_{i,j,k} = \frac{\text{BAY}(\text{dyu} \, dz u)}{\text{dyte}_{i,j}}, \quad (14.48)$$

where $\text{dyte}_{i,j}$ is the meridional width of the tracer cell's east side (see Figure 14.3 for definitions of grid distances). Doing so leads to

$$\begin{aligned}\mathcal{P}_{2x} &= - \sum \delta_i p' (\text{dyte} \, \text{uh_et}) \\ &= \sum p' \delta_i (\text{dyte} \, \text{uh_et}) \\ &= \sum p' \text{dat} \, \text{BDX_ET}(\text{uh_et}), \end{aligned} \quad (14.49)$$

where boundary terms vanish, and

$$\text{BDX_ET}(A) = \frac{A_{i,j} \text{dyte}_{i,j} - A_{i-1,j} \text{dyte}_{i-1,j}}{\text{dat}_{i,j}} \quad (14.50)$$

is a backwards finite difference operator for fields defined on the east face of tracer cells. Similar manipulations with the meridional term $v \partial_y p'$ leads to

$$\mathcal{P}_2 = \sum_{i,j,k} p' \text{dat} (\text{BDX_ET}(\text{uh_et}) + \text{BDY_NT}(\text{vh_nt})), \quad (14.51)$$

with

$$\text{vh_nt}_{i,j,k} = \frac{\text{BAX}(\text{dxu dzu } v)}{\text{dx tn}_{i,j}} \quad (14.52)$$

the meridional tracer advective velocity on the north face of the tracer cell, and

$$\text{BAX}(A) = \frac{A_i + A_{i-1}}{2} \quad (14.53)$$

defining the backward averaging operator. Finally,

$$\text{BDY_NT}(A) = \frac{A_{i,j} \text{dx tn}_{i,j} - A_{i,j-1} \text{dx tn}_{i,j-1}}{\text{dat}_{i,j}} \quad (14.54)$$

is a backwards finite difference operator for fields defined on the north face of tracer cells.

The horizontal thickness weighted advective velocity components uh_et and vh_nt are defined at the sides of the tracer cells, just like the C-grid velocity components (Figure 9.2 and 9.3)). They are the discrete representation of the thickness weighted advective velocity components that transport tracer and volume through the east and north cell faces. When fluid volume converges horizontally to a tracer cell, there is a corresponding dia-surface velocity component and a generally nonzero time tendency for the cell thickness. The thickness weighted volume budgets given by equations (14.19), (14.20), and (14.21) describe these effects.

Given that the advective velocity components uh_et and vh_nt are defined at the sides of the tracer cells, we are led to define a dia-surface velocity component w_bt_k at the bottom of the cell. It can generally be written by the discrete form of equation (14.20)

$$\text{w_bt}_k = \text{w_bt}_{k-1} + \text{BDX_ET}(\text{uh_et}_k) + \text{BDY_NT}(\text{vh_nt}_k) + \partial_t (\text{dz t}_k) - \mathcal{S}^{(V)} \text{dz t}_k. \quad (14.55)$$

Again, the time tendency on tracer cell thickness dz t_k is known in MOM from information about the vertically integrated volume budget (Section 10.8.1.5). So equation (14.55) is indeed a diagnostic expression for w_bt_k , evaluated from the surface down to the bottom. At the ocean surface, the dia-surface velocity component is determined by the input of water to the system

$$\text{w_bt}_{k=0} = -Q_m / \rho_o. \quad (14.56)$$

The minus sign is a convention, where positive $w > 0$ represents upward transport whereas positive $Q_m > 0$ represents downward transport of fresh water through the ocean surface into the ocean domain. Note that in general, water can enter the ocean domain at any depth through the source term $\mathcal{S}^{(V)}$. At the ocean bottom, we are ensured of a proper discretization so long as

$$\text{w_bt}_{k=k_{bot}} = 0 \quad (14.57)$$

is diagnosed to within numerical truncation. This statement is valid on either the B-grid or C-grid, since the ocean bottom on tracer cells is flat. It is a useful diagnostic for verifying the integrity of volume conservation discretization throughout a vertical column.

14.6.3 Divergence operator for surface height evolution

Integrating the continuity equation (14.55) vertically over an ocean column leads to

$$\sum_{k=1}^{nk} (\text{w_bt}_k - \text{w_bt}_{k-1}) = \sum_{k=1}^{nk} (\text{BDX_ET}(\text{uh_et}_k) + \text{BDY_NT}(\text{vh_nt}_k) + \partial_t (\text{dz t}_k) - \mathcal{S}^{(V)} \text{dz t}_k). \quad (14.58)$$

The time tendency for the thickness of an ocean column equals to that of the sea surface height, so that

$$\partial_t \eta = \sum_{k=1}^{nk} \partial_t (\text{dz t}_k). \quad (14.59)$$

Use of the surface and bottom boundary conditions (14.56) and (14.57) thus lead to

$$\partial_t \eta = Q_m / \rho_o - \sum_{k=1}^{nk} [\text{BDX_ET}(\text{uh_et}_k) + \text{BDY_NT}(\text{vh_nt}_k)] + \sum_{k=1}^{nk} \mathcal{S}^{(V)} dz t_k. \quad (14.60)$$

We are thus led to introduce a finite difference operator for the divergence of the vertically integrated transport

$$\text{DIV_UD}(U, V) = \sum_{k=1}^{nk} [\text{BDX_ET}(\text{uh_et}_k) + \text{BDY_NT}(\text{vh_nt}_k)]. \quad (14.61)$$

Use of the operator definitions (14.47), (14.50), (14.53), and (14.54), as well as the advection velocity components (14.48) and (14.52) leads to the form relevant for the B-grid

$$\boxed{\text{DIV_UD}(U, V) = \left(\frac{[\text{BAY}(\text{dyu } U)]_{i,j} - [\text{BAY}(\text{dyu } U)]_{i-1,j}}{\text{dat}_{i,j}} \right) + \left(\frac{[\text{BAX}(\text{dxu } V)]_{i,j} - [\text{BAX}(\text{dxu } V)]_{i,j-1}}{\text{dat}_{i,j}} \right)}, \quad (14.62)$$

where (U, V) is the vertically integrated horizontal velocity field defined by equation (14.45). Note that this is the same divergence operator that is used for the bottom pressure evolution when implementing the non-Boussinesq mass conserving form of MOM on a B-grid. The only difference is that the arguments become the density-weighted horizontal velocity (see Section 14.8.2).

14.6.4 Completing the manipulations for \mathcal{P}_2

Substituting expression (14.55) for the vertical advective velocity component into equation (14.51) leads to

$$\mathcal{P}_2 = \sum_{i,j,k} p'_k \text{dat} \left(\text{w_bt}_k - \text{w_bt}_{k-1} - \partial_t (dz t_k) + \mathcal{S}^{(V)} dz t_k \right). \quad (14.63)$$

Now move the vertical difference operator from the dia-surface velocity to the hydrostatic pressure via the following identity

$$\sum_{k=1}^{kbot} p'_k \text{dat} (\text{w_bt}_k - \text{w_bt}_{k-1}) = -p'_{k=1} \text{w_bt}_{k=0} \text{dat} - \sum_{k=1}^{kbot} \text{dat} \text{w_bt}_k (p'_{k+1} - p'_k), \quad (14.64)$$

where we used the lower boundary condition $p'_{kbot+1} \text{w_bt}_{kbot} = 0$ to reach this result. The next step requires us to specify how the hydrostatic pressure is computed. There are two ways, described in Sections 10.1.1 and 10.1.2.

14.6.4.1 Energetic approach

Section 10.1.1 noted that the older energetically based method specifies the hydrostatic pressure at the tracer point depth (Figure 14.2) according to

$$p'_{k=1} = g dz \text{tup}_{k=1} \rho'_{k=1} \quad k = 1 \quad (14.65)$$

$$p'_{k+1} = p'_k + g dz \text{wt}_k \overline{\rho'_k}^z \quad k > 1, \quad (14.66)$$

where

$$\overline{\rho'_k}^z = \frac{\rho'_k + \rho'_{k+1}}{2} \quad (14.67)$$

is the algebraically averaged density over the region between two tracer points T_k and T_{k+1} , and $dz \text{wt}_k$ is the vertical distance between the tracer points (Section 10.3). Substituting this result into equation (14.64) renders

$$\sum_{k=1}^{kbot} p'_k \text{dat} (\text{w_bt}_k - \text{w_bt}_{k-1}) = -p'_{k=1} \text{w_bt}_{k=0} \text{dat} - g \sum_{k=1}^{kbot} \text{dat} dz \text{wt}_k \text{w_bt}_k \overline{\rho'_k}^z. \quad (14.68)$$

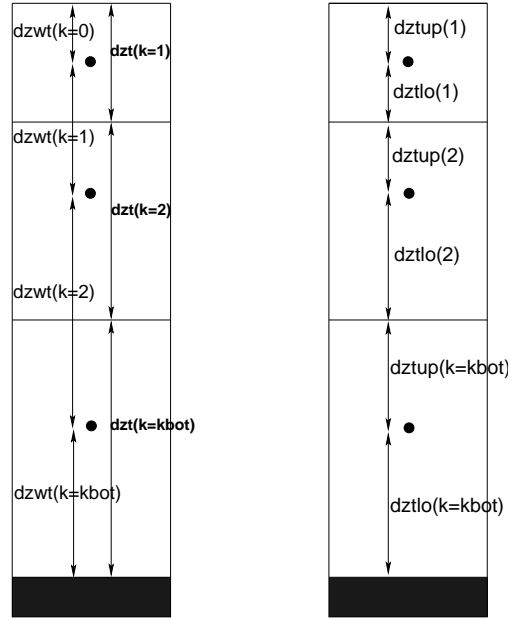


Figure 14.2: Left panel: schematic of the vertical grid cell arrangement used for computing the hydrostatic pressure at a depth $k + 1$ in terms of the pressure at depth k using equations (14.65) and (14.66). The vertical average of density is meant to account for the part of density within each of the two adjacent cells. The factor of $1/2$ used in the average operator yields an approximate average when vertical cells are non-uniform. Yet the $1/2$ factor is used for all vertical grid spacing since it renders a simple conversion of discrete pressure work to discrete gravity work. Right panel: grid cell thicknesses used for computing the finite volume hydrostatic pressure at depth $k + 1$ (see equations (14.70) and (14.71)). Note that this figure was also presented in Section 10.1 (see Figure 10.1).

This result then leads to

$$\mathcal{P}_2 = -g \sum_{i,j,k} \text{dat} \, dzwt_k \, w_bt_k \, \overline{\rho'_k}^z - \sum_{i,j} \text{dat} \, p'_{k=1} \, w_bt_{k=0} - \sum_{i,j,k} \text{dat} \, p'_k \left[\partial_t (dz t_k) - \mathcal{S}^{(V)} dz t_k \right]. \quad (14.69)$$

14.6.4.2 Finite volume approach

Section 10.1.2 noted that a finite volume based method specifies the hydrostatic pressure at the tracer point depth according to (see Figure 14.2)

$$p'_{k=1} = g \, dz t_{up k=1} \, \rho'_{k=1} \quad (14.70)$$

$$p'_{k+1} = p'_k + g \, dz t_{lo k} \, \rho'_k + g \, dz t_{up k+1} \, \rho'_{k+1}. \quad (14.71)$$

Substituting this result into equation (14.64) renders

$$\sum_{k=1}^{kbot} p'_k \, \text{dat} \, (w_bt_k - w_bt_{k-1}) = -p'_{k=1} \, w_bt_{k=0} \, \text{dat} - g \sum_{k=1}^{kbot} \text{dat} \, w_bt_k \, (dz t_{lo k} \, \rho'_k + g \, dz t_{up k+1} \, \rho'_{k+1}). \quad (14.72)$$

This result then leads to

$$\mathcal{P}_2 = -g \sum_{i,j,k} \text{dat} \, w_bt_k \, (dz t_{lo k} \, \rho'_k + g \, dz t_{up k+1} \, \rho'_{k+1}) - \sum_{i,j} \text{dat} \, p'_{k=1} \, w_bt_{k=0} - \sum_{i,j,k} \text{dat} \, p'_k \left[\partial_t (dz t_k) - \mathcal{S}^{(V)} dz t_k \right]. \quad (14.73)$$

14.6.5 The geopotential gradient term \mathcal{P}_3

Now consider the zonal piece of the geopotential gradient from equation (14.36)

$$\mathcal{P}_{3x} = - \sum_{i,j,k} dxu dyu dzu u \text{FAY}(\text{FAX}(\rho') \delta_i \Phi) / dxu. \quad (14.74)$$

Transferring the forward average FAY to a backward average BAY leads to

$$\mathcal{P}_{3x} = - \sum \text{BAY}(dyu dzu u) \text{FAX}(\rho') \delta_i \Phi, \quad (14.75)$$

where boundary terms vanish. Introducing the zonal thickness weighted advective transport velocity (14.48) yields

$$\mathcal{P}_{3x} = - \sum dyte uh_{et} \text{FAX}(\rho') \delta_i \Phi. \quad (14.76)$$

Moving the difference operator $\delta_i \Phi = \Phi_{i+1} - \Phi_i$ from the geopotential to the remaining terms gives

$$\begin{aligned} \mathcal{P}_{3x} &= \sum \Phi \delta_i (dyte \text{FAX}(\rho') uh_{et}) \\ &= \sum \Phi \text{dat BDX_ET}(\text{FAX}(\rho') uh_{et}), \end{aligned} \quad (14.77)$$

where boundary terms vanish. Similar manipulations with the meridional piece of \mathcal{P}_3 lead to

$$\mathcal{P}_3 = \sum \Phi \text{dat} [\text{BDX_ET}(\text{FAX}(\rho') uh_{et}) + \text{BDY_NT}(\text{FAY}(\rho') vh_{nt})]. \quad (14.78)$$

14.6.6 Summary for the Boussinesq pressure conversion

In summary, for the energetically based method for computing hydrostatic pressure, the projection of the horizontal velocity onto the downgradient pressure field is given by

$$\begin{aligned} \mathcal{P} &= - \sum_{i,j} \text{dau} [U \text{FDX_NT}(\text{FAY}(p_a + p_{\text{surf}})) + V \text{FDY_ET}(\text{FAX}(p_a + p_{\text{surf}}))] \\ &\quad - \sum_{i,j} \text{dat } p'_{k=1} w_{bt_{k=0}} \\ &\quad - g \sum_{i,j,k} \text{dat } dzwt_k w_{bt_k} \overline{\rho_k^z} \\ &\quad - \sum_{i,j,k} \text{dat } p'_k [\partial_t (dz t_k) - \mathcal{S}^{(V)} dz t_k] \\ &\quad + \sum_{i,j,k} \Phi \text{dat} [\text{BDX_ET}(\text{FAX}(\rho') uh_{et}) + \text{BDY_NT}(\text{FAY}(\rho') vh_{nt})]. \end{aligned} \quad (14.79)$$

Within the MOM energy analysis diagnostic, the code computes the left hand side of equation (14.79) and compares to the right hand side. Differences are due to coding errors. This diagnostic is very effective because it involves advective velocities on the tracer cells, both tracer and velocity cell distances, the calculation of pressure, and details of a partial step representation of the ocean bottom. Each requires precise discretization to ensure an energy conversion error at the numerical roundoff level. In a similar manner, for the finite volume approach to computing hydrostatic pressure, we have the projection of the horizontal

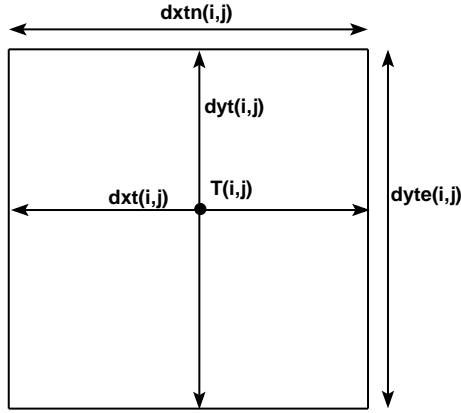


Figure 14.3: Time independent horizontal grid distances (meters) used for the tracer cell $T_{i,j}$ in MOM. $dxt_{i,j}$ and $dym_{i,j}$ are the grid distances of the tracer cell in the generalized zonal and meridional directions, and $dat_{i,j} = dxt_{i,j} dym_{i,j}$ is the area of the cell. The grid distance $dxtn_{i,j}$ is the zonal width of the north face of a tracer cell, and $dyme_{i,j}$ is the meridional width of the east face. Note that the tracer point $T_{i,j}$ is not generally at the center of the tracer cell. Distances are functions of both i and j due to the use of generalized orthogonal coordinates.

velocity onto the downgradient pressure field is given by

$$\begin{aligned}
 \mathcal{P} = & - \sum_{i,j} dau [U \text{ FDX_NT}(\text{FAY}(p_a + p_{\text{surf}})) + V \text{ FDY_ET}(\text{FAX}(p_a + p_{\text{surf}}))] \\
 & - \sum_{i,j} dat p'_{k=1} w_bt_{k=0} \\
 & - g \sum_{i,j,k} dat w_bt_k (dz t lo_k \rho'_k + g dz t up_{k+1} \rho'_{k+1}) \\
 & - \sum_{i,j,k} dat p'_k \left[\partial_t (dz t_k) - \mathcal{S}^{(v)} dz t_k \right] \\
 & + \sum_{i,j,k} \Phi dat [\text{BDX_ET}(\text{FAX}(\rho') uh_et) + \text{BDY_NT}(\text{FAY}(\rho') vh_nt)].
 \end{aligned} \tag{14.80}$$

14.7 C-grid Boussinesq pressure work conversions

We now consider pressure work conversions for the C-grid version of MOM. In this section, the horizontal velocity components (u, v) are located on the zonal and meridional faces of the tracer cell, as per the C-grid convention (Figure 9.2).

As for the B-grid in Section 14.6, the manipulations here lead to a consistent definition of the advective transport crossing tracer cell faces. Again, the definition is specified by requiring a sensible discrete energy conversion that reflects that found in the continuum. Once we define the advective transport uh_et and vh_nt for the C-grid, all subsequent results for the pressure work conversion manipulations are identical to those already considered for the B-grid.

14.7.1 Force from the horizontal pressure gradient

To start, consider the expressions given in Section 3.4.1 for the horizontal pressure gradient force acting on the C-grid momentum. The zonal momentum per horizontal area, $u \rho_o dz$, sits at the zonal face of the

tracer cell, and the meridional momentum per horizontal area, $v \rho_o dz$, sits at the meridional face (Figure 9.2). Horizontal pressure gradients impart a force to the momentum according to

$$\partial_t (u \rho_o dz te)^{\text{pressure}} = -dz te [FDX_T(p_a + p_{\text{surf}} + p') + FAX(\rho') FDX_T(\Phi')] \quad (14.81)$$

$$\partial_t (v \rho_o dz tn)^{\text{pressure}} = -dz tn [FDY_T(p_a + p_{\text{surf}} + p') + FAY(\rho') FDY_T(\Phi')]. \quad (14.82)$$

In these equations, derivative operators are defined by

$$\begin{aligned} FDX_T(A) &= \frac{A_{i+1} - A_i}{dx te_{i,j}} \\ FDY_T(A) &= \frac{A_{j+1} - A_j}{dy tn_{i,j}}, \end{aligned} \quad (14.83)$$

with the grid distances given in Figure 9.7. These operators are used for fields that live at the tracer point.

To compute the pressure work conversion

$$\mathcal{P} = - \int \mathbf{u} \cdot \nabla_z p dV \quad (14.84)$$

on the discrete C-grid, we multiply the zonal momentum equation (14.129) by the thickness weighted zonal velocity, $u dz te$, and the meridional momentum equation (14.130) by the thickness weighted meridional velocity, $v dz tn$ (see Section 14.5.2 for definition of thicknesses $dz te$ and $dz tn$), and integrate over the tracer cells to render

$$\begin{aligned} \mathcal{P} \equiv & - \sum_{i,j,k} \text{dat} [u dz te FDX_T(p_a + p_{\text{surf}}) + v dz tn FDY_T(p_a + p_{\text{surf}})] \\ & - \sum_{i,j,k} \text{dat} [u dz te FDX_T(p') + v dz tn FDY_T(p')] \\ & - \sum_{i,j,k} \text{dat} [u dz te FAX(\rho') FDX_T(\Phi) + v dz tn FAY(\rho') FDY_T(\Phi)]. \end{aligned} \quad (14.85)$$

Note the integration is over the area of the tracer cell, dat , which contrasts to the B-grid approach in Section 14.6, where integration is over the velocity cell area, dau . The reason for integrating here over the tracer area is that there is no analogous velocity cell control volume for the C-grid.

14.7.2 The vertically integrated term \mathcal{P}_1

The first line in equation (14.85) can be vertically integrated to yield

$$\begin{aligned} \mathcal{P}_1 & \equiv - \sum_{i,j,k} \text{dat} [u dz te FDX_T(p_a + p_{\text{surf}}) + v dz tn FDY_T(p_a + p_{\text{surf}})] \\ & = - \sum_{i,j} \text{dat} [U FDX_T(p_a + p_{\text{surf}}) + V FDY_T(p_a + p_{\text{surf}})] \end{aligned} \quad (14.86)$$

where

$$(U, V) = \sum_k (u dz te, v dz tn) \quad (14.87)$$

is the vertically integrated horizontal velocity field. The \mathcal{P}_1 term represents the work of applied pressure and geopotential moving vertical columns of fluid.

14.7.3 Advection velocity components for tracers

Focus on the zonal piece of the baroclinic pressure term appearing in equation (14.85), in which

$$\begin{aligned}
 \mathcal{P}_{2x} &\equiv - \sum_{i,j,k} \text{dat } u \text{ dzte } \text{FDX_T}(p') \\
 &= - \sum \text{dxt dyt } u \text{ dzte } \left(\frac{\delta_i p'_i}{\text{dxt e}} \right) \\
 &= - \sum \delta_i p'_i \text{dyte} \left(\frac{\text{dxt}}{\text{dxt e}} \right) \left(\frac{\text{dyt}}{\text{dyte}} \right) \text{dzte } u.
 \end{aligned} \tag{14.88}$$

See Figure 14.3 for definitions of grid distances. Now *define* the zonal thickness weighted advective transport velocity placed on the eastern face of a tracer cell as

$$\text{uh_et}_{i,j,k} = u \text{ dzte} \left(\frac{\text{dxt}}{\text{dxt e}} \right) \left(\frac{\text{dyt}}{\text{dyte}} \right), \tag{14.89}$$

which leads to

$$\mathcal{P}_{2x} = - \sum \delta_i p' (\text{dyte } \text{uh_et}). \tag{14.90}$$

This is the exact same form as that arrived at for the B-grid in equation (14.49). All subsequent manipulations performed on the B-grid in Sections 14.6.2 and 14.6.4 follow through with the modified form of the advective transport given by equation (14.89), and the corresponding meridional transport

$$\text{vh_nt}_{i,j,k} = v \text{ dztn} \left(\frac{\text{dxt}}{\text{dxt n}} \right) \left(\frac{\text{dyt}}{\text{dytn}} \right). \tag{14.91}$$

In particular, the vertical transport w_bt satisfies the same continuity equation (14.55) as for the B-grid.

14.7.4 Divergence operator for surface height evolution

We follow the discussion in Section 14.6.3 to develop the divergence operator for the C-grid sea level evolution. For this purpose, we start from the general form of the finite difference divergence operator (14.61)

$$\text{DIV_UD}(U, V) = \sum_{k=1}^{nk} [\text{BDX_ET}(\text{uh_et}_k) + \text{BDY_NT}(\text{vh_nt}_k)], \tag{14.92}$$

with this form valid for either the B-grid or C-grid. Use of the operator definitions (14.50) and (14.54), as well as the advection velocity components (14.89) and (14.91) leads to the form appropriate for the C-grid

$$\boxed{\text{DIV_UD}(U, V) = \left(\frac{\left(\frac{\text{dat}}{\text{dxt e}} U \right)_{i,j} - \left(\frac{\text{dat}}{\text{dxt e}} U \right)_{i-1,j}}{\text{dat}_{i,j}} \right) + \left(\frac{\left(\frac{\text{dat}}{\text{dytn}} V \right)_{i,j} - \left(\frac{\text{dat}}{\text{dytn}} V \right)_{i,j-1}}{\text{dat}_{i,j}} \right)} \tag{14.93}$$

where (U, V) is the vertically integrated horizontal velocity field defined by equation (14.87). Note that this is the same divergence operator that is used for the bottom pressure evolution when implementing the non-Boussinesq mass conserving form of MOM on a C-grid. The only difference is that the arguments become the density-weighted horizontal velocity (see Section 14.9.2).

14.7.5 The geopotential gradient term \mathcal{P}_3

Now consider the zonal piece of the geopotential gradient from equation (14.85)

$$\begin{aligned}
 \mathcal{P}_{3x} &= - \sum_{i,j,k} \text{dat dzte } u \text{ FAX}(\rho') \text{FDX_T}(\Phi) \\
 &= - \sum_{i,j,k} \left(u \text{ dzte} \frac{\text{dxt}}{\text{dxt e}} \frac{\text{dyt}}{\text{dyte}} \right) \text{dyte } \text{FAX}(\rho') \delta_i \Phi.
 \end{aligned} \tag{14.94}$$

Introduce the zonal transport uh_{et} given by equation (14.89) to render

$$\mathcal{P}_{3x} = - \sum_{i,j,k} uh_{et} dyte FAX(\rho') \delta_i \Phi. \quad (14.95)$$

This expression has the identical form as equation (14.76) derived for the B-grid. Consequently, all subsequent manipulations conducted for the B-grid now follow for the C-grid.

14.7.6 Summary for the Boussinesq pressure conversion

The summary for the C-grid pressure conversion terms is very similar to that for the B-grid given in Section 14.6.6. For the energetically based method for computing hydrostatic pressure, the projection of the horizontal velocity onto the downgradient pressure field is given by

$$\begin{aligned} \mathcal{P} = & - \sum_{i,j} \text{dat} [U \text{FDX_T}(p_a + p_{\text{surf}}) + V \text{FDY_T}(p_a + p_{\text{surf}})] \\ & - \sum_{i,j} \text{dat} p'_{k=1} w_{bt_{k=0}} \\ & - g \sum_{i,j,k} \text{dat} dzwt_k w_{bt_k} \overline{\rho'_k}^z \\ & - \sum_{i,j,k} \text{dat} p'_k [\partial_t (dz t_k) - \mathcal{S}^{(V)} dz t_k] \\ & + \sum_{i,j,k} \Phi \text{dat} [\text{BDX_ET}(FAX(\rho') uh_{et}) + \text{BDY_NT}(FAY(\rho') vh_{nt})]. \end{aligned} \quad (14.96)$$

In a similar manner, for the finite volume approach to computing hydrostatic pressure, we have the projection of the horizontal velocity onto the downgradient pressure field is given by

$$\begin{aligned} \mathcal{P} = & - \sum_{i,j} \text{dat} [U \text{FDX_T}(p_a + p_{\text{surf}}) + V \text{FDY_T}(p_a + p_{\text{surf}})] \\ & - \sum_{i,j} \text{dat} p'_{k=1} w_{bt_{k=0}} \\ & - g \sum_{i,j,k} \text{dat} w_{bt_k} (dz t_{lo_k} \rho'_k + g dz t_{up_{k+1}} \rho'_{k+1}) \\ & - \sum_{i,j,k} \text{dat} p'_k [\partial_t (dz t_k) - \mathcal{S}^{(V)} dz t_k] \\ & + \sum_{i,j,k} \Phi \text{dat} [\text{BDX_ET}(FAX(\rho') uh_{et}) + \text{BDY_NT}(FAY(\rho') vh_{nt})]. \end{aligned} \quad (14.97)$$

14.8 B-grid non-Boussinesq pressure work conversions

Now consider manipulations of the globally integrated discrete B-grid representation of $\mathbf{u} \cdot \nabla_z p$ for the non-Boussinesq case. Here, we are concerned with the domain integrated scalar product

$$\begin{aligned} \mathbf{u} \cdot \nabla_z p &= \mathbf{u} \cdot (\nabla_s p + \rho \nabla_s \Phi) \\ &= (\rho/\rho_o) \mathbf{u} \cdot \nabla (p_b + \rho_o \Phi_b) + \mathbf{u} \cdot [-(\rho'/\rho_o) \nabla_s p + \rho \nabla_s \Phi'], \end{aligned} \quad (14.98)$$

where equation (3.27) was used for the pressure gradient as implemented in pressure based vertical coordinate models. To determine the proper discrete form of the pressure conversion, recall from Section 3.3.2

the momentum equations for a non-Boussinesq fluid on a B-grid with just the impacts from pressure acting

$$\partial_t (u \text{ rho_dzu})^{\text{pressure}} = -\text{rho_dzu FDX_NT}(\text{FAY}(p_b/\rho_o + \Phi_b + \Phi')) + \text{dzu FAY}[\delta_i p \text{ FAX}(\rho'/\rho_o)]/\text{dxu}_{i,j} \quad (14.99)$$

$$\partial_t (v \text{ rho_dzu})^{\text{pressure}} = -\text{rho_dzu FDY_ET}(\text{FAX}(p_b/\rho_o + \Phi_b + \Phi')) + \text{dzu FAX}[\delta_j p \text{ FAY}(\rho'/\rho_o)]/\text{dyu}_{i,j}. \quad (14.100)$$

The pressure conversion

$$\mathcal{P} = - \int \mathbf{u} \cdot \nabla_z p \, dV \quad (14.101)$$

for a non-Boussinesq fluid thus has the following discrete representation on the B-grid

$$\begin{aligned} \rho_o \mathcal{P} \equiv & - \sum_{i,j,k} \text{dau rho_dzu} [u \text{ FDX_NT}(\text{FAY}(p_b + \rho_o \Phi_b)) + v \text{ FDY_ET}(\text{FAX}(p_b + \rho_o \Phi_b))] \\ & - \sum_{i,j,k} \text{dau rho_dzu} [u \text{ FDX_NT}(\text{FAY}(\Phi')) + v \text{ FDY_ET}(\text{FAX}(\Phi'))] \\ & + \sum_{i,j,k} \text{dau dzu} [u \text{ FAY}(\text{FAX}(\rho') \delta_i p)/\text{dxu} + v \text{ FAX}(\text{FAY}(\rho') \delta_j p)/\text{dyu}]. \end{aligned} \quad (14.102)$$

We now consider these terms individually.

14.8.1 The vertically integrated term \mathcal{P}_1

The first term in equation (14.102) can be vertically integrated as

$$\begin{aligned} \mathcal{P}_1 \equiv & - \sum_{i,j,k} \text{dau rho_dzu} [u \text{ FDX_NT}(\text{FAY}(p_b/\rho_o + \Phi_b)) + v \text{ FDY_ET}(\text{FAX}(p_b/\rho_o + \Phi_b))] \\ = & - \sum_{i,j} \text{dau} [U^\rho \text{ FDX_NT}(\text{FAY}(p_b/\rho_o + \Phi_b)) + V^\rho \text{ FDY_ET}(\text{FAX}(p_b/\rho_o + \Phi_b))] \end{aligned} \quad (14.103)$$

where

$$(U^\rho, V^\rho) = \sum_k \text{rho_dzu}(u, v) \quad (14.104)$$

is the vertically integrated density weighted horizontal velocity field. Equivalently, it is the vertically integrated horizontal momentum per horizontal area. The \mathcal{P}_1 term represents the effects of applied pressure and geopotential working on moving vertical columns of fluid.

14.8.2 Defining the advective mass transport

To motivate the definition of the advection velocity for the non-Boussinesq case, we focus on the zonal part of the geopotential term

$$\begin{aligned} \mathcal{P}_{2x} \equiv & - \sum_{i,j,k} \text{dau rho_dzu} u \text{ FDX_NT}(\text{FAY}(\Phi')) \\ = & - \frac{1}{2} \sum \text{dyu rho_dzu} u \delta_i (\Phi'_j + \Phi'_{j+1}) \\ = & - \sum \text{BAY}(\text{dyu rho_dzu} u) \delta_i \Phi'_j. \end{aligned} \quad (14.105)$$

The boundary terms were dropped since they vanish for either periodic or solid wall conditions. Now *define* the thickness weighted and density weighted zonal advective transport on the eastern face of a tracer cell

$$\boxed{\text{uhrho_et}_{i,j,k} = \frac{\text{BAY}(\text{dyu rho_dzu} u)}{\text{dyte}_{i,j}}.} \quad (14.106)$$

This definition of the non-Boussinesq advective mass transport leads to

$$\begin{aligned}\mathcal{P}_{2x} &= - \sum \delta_i \Phi' (\text{dyte uhrho_et}) \\ &= \sum \Phi' \delta_i (\text{dyte uhrho_et}) \\ &= \sum \Phi' \text{dat BDx_ET}(\text{uhrho_et}),\end{aligned}\tag{14.107}$$

where boundary terms vanish. Similar manipulations with the meridional term $v \partial_y p$, and reintroducing the two-dimensional pieces, leads to

$$\mathcal{P}_2 = \sum \Phi' \text{dat} (\text{BDx_ET}(\text{uhrho_et}) + \text{BDy_NT}(\text{vhrho_nt})),\tag{14.108}$$

with

$$\boxed{\text{vhrho_nt}_{i,j,k} = \frac{\text{BAX}(\text{dxu rho_dzu } v)}{\text{dxtn}_{i,j}}}\tag{14.109}$$

the meridional density and thickness weighted advective tracer velocity on the north face of the tracer cell.

As for the Boussinesq case, the horizontal advective velocities `uhrho_et` and `vhrho_nt` are defined at the sides of the tracer cells. They are the discrete representation of the thickness and density weighted advective velocity transporting tracer and seawater mass through the east and north cell faces. When mass converges horizontally to a tracer cell, there is a corresponding dia-surface velocity component and a generally nonzero time tendency for the cell thickness. The thickness weighted mass budgets given by equations (14.22), (14.23), and (14.24) describe these effects.

Given that the advective mass transport `uhrho_et` and `vhrho_nt` are defined at the sides of the tracer cells, we are led to define a density weighted dia-surface velocity component `wrho_btk` at the bottom of the cell. It is determined by the discrete form of equation (14.23)

$$\text{wrho_bt}_k = \partial_t(\text{rho_dzt})_k - \text{rho_dzt } \mathcal{S}^{(M)} + \text{BDx_ET}(\text{uhrho_et}_k) + \text{BDy_NT}(\text{vhrho_nt}_k) + \text{wrho_bt}_{k-1}.\tag{14.110}$$

For the non-Boussinesq version of MOM, we use pressure-based vertical coordinates so that the tracer and velocity cells maintain the identity

$$\rho \, dz = \rho \left(\frac{\partial z}{\partial s} \right) ds\tag{14.111}$$

where $\rho \, \partial z / \partial s$ is depth independent. The time tendency on density weighted tracer cell thickness $(\text{rho_dzt})_k$ is known in MOM from information about the vertically integrated mass budget (Section 10.6.2). So equation (14.110) is indeed a diagnostic expression for `wrho_btk`, evaluated from the surface down to the bottom. At the ocean surface, the dia-surface velocity component is determined by the input of fresh water to the system

$$\text{wrho_bt}_{k=0} = -Q_m.\tag{14.112}$$

At the ocean bottom, we are ensured of a proper discretization so long as

$$\text{wrho_bt}_{k=kbot} = 0\tag{14.113}$$

is diagnosed.

14.8.3 Completing the manipulations for \mathcal{P}_2

Substituting expression (14.110) for the vertical advective velocity component into equation (14.108) leads to

$$\mathcal{P}_2 = \sum \Phi'_k \text{dat} (\text{wrho_bt}_k - \text{wrho_bt}_{k-1} - \partial_t(\text{rho_dzt})_k + \text{rho_dzt}_k \mathcal{S}^{(M)}).\tag{14.114}$$

Now move the vertical difference operator from the dia-surface velocity to the hydrostatic pressure via the following identity

$$\sum_{k=1}^{kbot} \Phi'_k \text{dat} (\text{wrho_bt}_k - \text{wrho_bt}_{k-1}) = -\text{dat} \Phi'_{k=1} \text{wrho_bt}_{k=0} + \sum_{k=1}^{kbot} \text{dat} \text{wrho_bt}_k (\Phi'_k - \Phi'_{k+1}),\tag{14.115}$$

where we used the lower boundary condition $\Phi'_{k_{bot}+1} \text{wrho_bt}_{k_{bot}} = 0$ to reach this result. The next step requires us to specify how the anomalous geopotential height is computed. There are two ways, described in Sections 10.1.3 and 10.1.4.

14.8.3.1 Energetic approach

In Section 10.1.3, we noted that the older energetically based method specifies the anomalous geopotential height at the tracer point depth (Figure 14.2) according to

$$\Phi'_k = \Phi'_{k+1} - (g/\rho_o) \text{dzw}_{\text{t}_k} \bar{\rho}'_k{}^z. \quad (14.116)$$

In contrast to the hydrostatic pressure calculation in equations (14.65) and (14.66), the geopotential calculation proceeds from the bottom upwards. Substituting equation (14.116) into equation (14.115) renders

$$\sum_{k=1}^{k_{bot}} \Phi'_k \text{dat} (\text{wrho_bt}_k - \text{wrho_bt}_{k-1}) = -\Phi'_{k=1} \text{wrho_bt}_{k=0} \text{dat} - (g/\rho_o) \sum_{k=1}^{k_{bot}} \text{dat} \text{dzw}_{\text{t}_k} \text{wrho_bt}_k \bar{\rho}'_k{}^z, \quad (14.117)$$

which then leads to

$$\begin{aligned} \mathcal{P}_2 = & -(g/\rho_o) \sum_{i,j,k} \text{dat} \text{dzw}_{\text{t}_k} \text{wrho_bt}_k \bar{\rho}'_k{}^z - \sum_{i,j} \text{dat} \Phi'_{k=1} \text{wrho_bt}_{k=0} \\ & + \sum_{i,j,k} \text{dat} \Phi' [(\text{rho_dzt})_k \mathcal{S}^{(M)} - \partial_t (\text{rho_dzt})_k]. \end{aligned} \quad (14.118)$$

14.8.3.2 Finite volume approach

In Section 10.1.4, we noted that the finite volume based method specifies the anomalous geopotential height at the tracer point depth (Figure 14.2) according to

$$\Phi'_{k=k_{bot}} = -(g/\rho_o) \text{dztl}_{\text{o}_{k_{bot}}} \rho'_{k=k_{bot}} \quad (14.119)$$

$$\Phi'_k = \Phi'_{k+1} - (g/\rho_o) \text{dztp}_{\text{t}_{k+1}} \rho'_{k+1} - (g/\rho_o) \text{dztl}_{\text{o}_k} \rho'_k. \quad (14.120)$$

Substituting equations (14.119) and (14.120) into equation (14.115) renders

$$\sum_{k=1}^{k_{bot}} \Phi'_k \text{dat} (\text{wrho_bt}_k - \text{wrho_bt}_{k-1}) = -\Phi'_{k=1} \text{wrho_bt}_{k=0} \text{dat} - (g/\rho_o) \sum_{k=1}^{k_{bot}} \text{dat} \text{wrho_bt}_k (\text{dztp}_{\text{t}_{k+1}} \rho'_{k+1} + \text{dztl}_{\text{o}_k} \rho'_k), \quad (14.121)$$

which then leads to

$$\begin{aligned} \mathcal{P}_2 = & -(g/\rho_o) \sum_{i,j,k} \text{dat} \text{wrho_bt}_k (\text{dztp}_{\text{t}_{k+1}} \rho'_{k+1} + \text{dztl}_{\text{o}_k} \rho'_k) - \sum_{i,j} \text{dat} \Phi'_{k=1} \text{wrho_bt}_{k=0} \\ & + \sum_{i,j,k} \text{dat} \Phi' [(\text{rho_dzt})_k \mathcal{S}^{(M)} - \partial_t (\text{rho_dzt})_k]. \end{aligned} \quad (14.122)$$

14.8.4 The pressure gradient term \mathcal{P}_3

Now consider the zonal piece of the geopotential gradient from equation (14.102)

$$\rho_o \mathcal{P}_{3x} = \sum_{i,j,k} \text{dxu dyu dzu } u \text{FAY}(\text{FAX}(\rho') \delta_i p) / \text{dxu} \quad (14.123)$$

Transferring the forward average FAY to a backward average BAY leads to

$$\rho_o \mathcal{P}_{3x} = - \sum \text{BAY}(\text{dyu dzu } u) \text{FAX}(\rho') \delta_i p, \quad (14.124)$$

where boundary terms vanish. Further manipulations, analogous to the Boussinesq case in Section 14.6.5, do not appear possible since the density weighted advection velocity will not appear. Instead, the ρ' weighted velocity appears, and this is not relevant. So we simply write this term in its unmanipulated form

$$\rho_o \mathcal{P}_3 = \sum_{i,j,k} \text{dau dzu} \left[u \text{FAY}(\text{FAX}(\rho') \delta_i p) / \text{dxu} + v \text{FAX}(\text{FAY}(\rho') \delta_j p) / \text{dyu} \right]. \quad (14.125)$$

14.8.5 Summary for the non-Boussinesq pressure conversion

In summary, for the energetically based method for computing the anomalous geopotential, the projection of the horizontal velocity onto the downgradient pressure field in the non-Boussinesq case is given by

$$\begin{aligned} \mathcal{P} = & - \sum_{i,j} \text{dau} \left[U^p \text{FDX_NT}(\text{FAY}(p_b/\rho_o + \Phi_b)) + V^p \text{FDY_ET}(\text{FAX}(p_b/\rho_o + \Phi_b)) \right] \\ & - \sum_{i,j} \text{dat} \Phi'_{k=1} \text{wrho_bt}_{k=0} \\ & - (g/\rho_o) \sum_{i,j,k} \text{dat dzwt}_k \text{wrho_bt}_k \overline{\rho'_k}^z \\ & - \sum_{i,j,k} \text{dat} \Phi'_k \left[\partial_t (\text{rho_dzt})_k - \rho \text{dzt}_k \mathcal{S}^{(M)} \right] \\ & + \rho_o^{-1} \sum_{i,j,k} \text{dau dzu} \left[u \text{FAY}(\text{FAX}(\rho') \delta_i p) / \text{dxu} + v \text{FAX}(\text{FAY}(\rho') \delta_j p) / \text{dyu} \right]. \end{aligned} \quad (14.126)$$

In a similar manner, for the finite volume approach to computing anomalous geopotential height, the projection of the horizontal velocity onto the downgradient pressure field is given by

$$\begin{aligned} \mathcal{P} = & - \sum_{i,j} \text{dau} \left[U^p \text{FDX_NT}(\text{FAY}(p_b/\rho_o + \Phi_b)) + V^p \text{FDY_ET}(\text{FAX}(p_b/\rho_o + \Phi_b)) \right] \\ & - \sum_{i,j} \text{dat} \Phi'_{k=1} \text{wrho_bt}_{k=0} \\ & - (g/\rho_o) \sum_{i,j,k} \text{dat wrho_bt}_k (\text{dztup}_{k+1} \rho'_{k+1} + \text{dztl}_k \rho'_k) \\ & - \sum_{i,j,k} \text{dat} \Phi'_k \left[\partial_t (\text{rho_dzt})_k - \text{rho_dzt}_k \mathcal{S}^{(M)} \right] \\ & + \rho_o^{-1} \sum_{i,j,k} \text{dau dzu} \left[u \text{FAY}(\text{FAX}(\rho') \delta_i p) / \text{dxu} + v \text{FAX}(\text{FAY}(\rho') \delta_j p) / \text{dyu} \right]. \end{aligned} \quad (14.127)$$

14.9 C-grid non-Boussinesq pressure work conversions

We now consider the discrete representation of the pressure conversion

$$\mathcal{P} = - \int \mathbf{u} \cdot \nabla_z p \, dV \quad (14.128)$$

on the C-grid for a non-Boussinesq fluid. To start, consider the expressions given in Section 3.4.2 for the horizontal pressure gradient force acting on the C-grid momentum in a non-Boussinesq fluid

$$\partial_t (u \text{rho_dzte})^{\text{pressure}} = -\text{rho_dzte} \text{FDX_T}(p_b/\rho_o + \Phi_b + \Phi') + \text{dzte} \text{FAX}(\rho'/\rho_o) \text{FDX_T}(p) \quad (14.129)$$

$$\partial_t (v \text{rho_dztn})^{\text{pressure}} = -\text{rho_dztn} \text{FDY_T}(p_b/\rho_o + \Phi_b + \Phi') + \text{dztn} \text{FAY}(\rho'/\rho_o) \text{FDY_T}(p). \quad (14.130)$$

Recall that the zonal momentum per horizontal area, $u \rho dz$, sits at the zonal face of the tracer cell, and the meridional momentum per horizontal area, $v \rho dz$, sits at the meridional face (Figure 9.2). The discrete form of the pressure work conversion for a non-Boussinesq fluid on a C-grid is thus given by

$$\begin{aligned} \mathcal{P} \equiv & - \sum_{i,j,k} \text{dat} [u \text{ rho_dz te FDX_T}(p_b/\rho_o + \Phi_b) + v \text{ rho_dz tn FDY_T}(p_b/\rho_o + \Phi_b)] \\ & - \sum_{i,j,k} \text{dat} [u \text{ rho_dz te FDX_T}(\Phi') + v \text{ rho_dz tn FDY_T}(\Phi')] \\ & + \sum_{i,j,k} \text{dat} [u \text{ dz te FAX}(\rho'/\rho_o) \text{ FDX_T}(p) + v \text{ dz tn FAY}(\rho'/\rho_o) \text{ FDY_T}(p)]. \end{aligned} \quad (14.131)$$

Note the integration is over the area of the tracer cell, dat , which contrasts to the B-grid approach in Section 14.8, where integration is over the velocity cell area, dau . The reason for integrating here over the tracer area is that there is no analogous velocity cell control volume for the C-grid.

14.9.1 The vertically integrated term \mathcal{P}_1

The first term in equation (14.131) can be vertically integrated as

$$\begin{aligned} \mathcal{P}_1 \equiv & - \sum_{i,j,k} \text{dat} [u \text{ rho_dz te FDX_T}(p_b/\rho_o + \Phi_b) + v \text{ rho_dz tn FDY_T}(p_b/\rho_o + \Phi_b)] \\ = & - \sum_{i,j} \text{dat} [U^\rho \text{ FDX_T}(p_b/\rho_o + \Phi_b) + V^\rho \text{ FDY_T}(p_b/\rho_o + \Phi_b)] \end{aligned} \quad (14.132)$$

where

$$(U^\rho, V^\rho) = \sum_k (u \text{ rho_dz te}, v \text{ rho_dz tn}) \quad (14.133)$$

is the vertically integrated density weighted horizontal velocity field for the C-grid. Equivalently, it is the vertically integrated horizontal momentum per horizontal area. The \mathcal{P}_1 term represents the effects of applied pressure and geopotential working on moving vertical columns of fluid.

14.9.2 Defining the advective mass transport

To motivate the definition of the advection velocity for the non-Boussinesq case, we focus on the zonal part of the geopotential term

$$\mathcal{P}_{2x} \equiv - \sum_{i,j,k} \text{dat} u \text{ rho_dz te FDX_T}(\Phi'). \quad (14.134)$$

We can immediately transfer the results from the Boussinesq case in equation (14.88) to render the advective mass transport for the C-grid non-Boussinesq fluid

$$\boxed{\text{uhrho_et}_{i,j,k} = u \text{ rho_dz te} \left(\frac{dx}{dx \text{ te}} \right) \left(\frac{dy}{dy \text{ te}} \right)}, \quad (14.135)$$

which leads to

$$\mathcal{P}_{2x} = - \sum \delta_i \Phi' (\text{dyte uhrho_et}). \quad (14.136)$$

The corresponding meridional advective mass transport for the C-grid non-Boussinesq fluid is given by

$$\boxed{\text{vhrho_nt}_{i,j,k} = v \text{ rho_dz tn} \left(\frac{dx}{dx \text{ tn}} \right) \left(\frac{dy}{dy \text{ tn}} \right)}, \quad (14.137)$$

These results place the C-grid results in line with that already considered in Section 14.8.2 for the B-grid. All subsequent manipulations thus follow just as for the B-grid.

14.9.3 The pressure gradient term \mathcal{P}_3

As for the B-grid considerations in Section 14.8.4, we do not find manipulations of the pressure gradient term in equation (14.131) to be possible, so we simply write the unmanipulated form as

$$\rho_o \mathcal{P}_3 = \sum_{i,j,k} \text{dat} [u \, dz \, te \, FAX(\rho') \, FDX_T(p) + v \, dz \, tn \, FAY(\rho') \, FDY_T(p)]. \quad (14.138)$$

14.9.4 Summary for the non-Boussinesq pressure conversion

In summary, for the energetically based method for computing the anomalous geopotential, the projection of the horizontal velocity onto the downgradient pressure field in the non-Boussinesq C-grid case is given by

$$\begin{aligned} \mathcal{P} = & - \sum_{i,j} \text{dat} [U^\rho \, FDX_T(p_b/\rho_o + \Phi_b) + V^\rho \, FDY_T(p_b/\rho_o + \Phi_b)] \\ & - \sum_{i,j} \text{dat} \, \Phi'_{k=1} \, wrho_bt_{k=0} \\ & - (g/\rho_o) \sum_{i,j,k} \text{dat} \, dzwt_k \, wrho_bt_k \, \overline{\rho'_k}^z \\ & - \sum_{i,j,k} \text{dat} \, \Phi'_k [\partial_t (rho_dzt)_k - \rho \, dzt_k \, \mathcal{S}^{(M)}] \\ & + \rho_o^{-1} \sum_{i,j,k} \text{dat} [u \, dz \, te \, FAX(\rho') \, FDX_T(p) + v \, dz \, tn \, FAY(\rho') \, FDY_T(p)]. \end{aligned} \quad (14.139)$$

In a similar manner, for the finite volume approach to computing anomalous geopotential height, the projection of the horizontal velocity onto the downgradient pressure field is given by

$$\begin{aligned} \mathcal{P} = & - \sum_{i,j} \text{dat} [U^\rho \, FDX_T(p_b/\rho_o + \Phi_b) + V^\rho \, FDY_T(p_b/\rho_o + \Phi_b)] \\ & - \sum_{i,j} \text{dat} \, \Phi'_{k=1} \, wrho_bt_{k=0} \\ & - (g/\rho_o) \sum_{i,j,k} \text{dat} \, wrho_bt_k (dztup_{k+1} \, \rho'_{k+1} + dztlo_k \, \rho'_k) \\ & - \sum_{i,j,k} \text{dat} \, \Phi'_k [\partial_t (rho_dzt)_k - rho_dzt_k \, \mathcal{S}^{(M)}] \\ & + \rho_o^{-1} \sum_{i,j,k} \text{dat} [u \, dz \, te \, FAX(\rho') \, FDX_T(p) + v \, dz \, tn \, FAY(\rho') \, FDY_T(p)]. \end{aligned} \quad (14.140)$$

14.10 Effective Coriolis force and mechanical energy

In the continuum, the Coriolis force and the advection metric frequency combine to yield the effective Coriolis force per volume (see equation (14.1))

$$\mathbf{F}^{\text{eff-coriolis}} = \tilde{f} (\hat{\mathbf{z}} \wedge \rho \mathbf{u}), \quad (14.141)$$

where

$$\tilde{f} = f + \mathcal{M} \quad (14.142)$$

is the effective Coriolis parameter, and (see equation (4.49) in Griffies (2004))

$$\mathcal{M} = v \left(\frac{\partial_x dy}{dy} \right) - u \left(\frac{\partial_y dx}{dx} \right) \quad (14.143)$$

is the advective metric frequency arising from the sphericity of the earth. The effective Coriolis force has zero impact on mechanical energy, since it is directed perpendicular to the horizontal velocity

$$\mathbf{u} \cdot \mathbf{F}^{\text{eff-coriolis}} = \mathbf{u} \cdot \tilde{f} (\hat{\mathbf{z}} \wedge \rho \mathbf{u}) = 0. \quad (14.144)$$

However, assumptions about time and space discretizations generally compromise this continuum result.

14.10.1 B-grid

On the B-grid, the effective Coriolis force per horizontal area takes the form

$$F^{\text{x-eff-coriolis}} = -(f + \mathcal{M}) v \rho \text{d}z \text{d}u \quad (14.145)$$

$$F^{\text{y-eff-coriolis}} = (f + \mathcal{M}) u \rho \text{d}z \text{d}u, \quad (14.146)$$

where the Coriolis parameter f is computed at the B-grid velocity point, as is the advective metric frequency

$$\mathcal{M} = v \text{d}h2\text{d}x - u \text{d}h1\text{d}y \quad (14.147)$$

with

$$\text{d}h2\text{d}x = \frac{\text{d}y_{ue,i,j} - \text{d}y_{ue,i-1,j}}{\text{d}x_u \text{d}y_u} \quad (14.148)$$

$$\text{d}h1\text{d}y = \frac{\text{d}x_{un,i,j} - \text{d}x_{un,i,j-1}}{\text{d}x_u \text{d}y_u}. \quad (14.149)$$

(see Figure 14.4 for grid distances). Naively, the inner product of the horizontal velocity field, $\mathbf{u}(\tau)$ and the effective Coriolis force will vanish on the discrete B-grid. However, temporal staggering compromises this result in the following cases.

- The Coriolis force on the B-grid is generally evaluated using a semi-implicit approach (Section 12.2). We may also use an Adams-Bashforth approach, as for the momentum advection (Section 11.2.3 and (Durran, 1999)). Either approach leaves a nonzero contribution of the Coriolis force in the mechanical energy budget. Only with the older, and nearly obsolete, leap-frog method with an explicit Coriolis force will the Coriolis force vanish for the B-grid mechanical energy budget.
- The advection metric frequency is evaluated in MOM as part of the momentum advection operator. With the leap-frog, this operator is evaluated at the present time step, and so the advection metric frequency will drop out from the mechanical energy budget. However, using the staggered time stepping preferred in MOM, the advection of momentum is temporally discretized using an Adams-Bashforth scheme (Section 11.2.3 and (Durran, 1999)), which means the advection metric frequency will not drop out from the mechanical energy budget.

14.10.2 C-grid

We presented the C-grid form of the Coriolis force per area in Section 12.1.2. We add to that force the force arising from the advective metric frequency to consider the effective Coriolis force per area

$$\begin{aligned} (\tilde{f} v \rho \text{d}z)^{\text{x-eff coriolis}} &\approx (1/4) [\tilde{f}_{i,j} (v \rho \text{d}z)_{i,j} + \tilde{f}_{i,j} (v \rho \text{d}z)_{i+1,j} + \tilde{f}_{i,j-1} (v \rho \text{d}z)_{i,j-1} + \tilde{f}_{i,j-1} (v \rho \text{d}z)_{i+1,j-1}] \\ -(\tilde{f} u \rho \text{d}z)^{\text{y-eff coriolis}} &\approx -(1/4) [\tilde{f}_{i-1,j} (u \rho \text{d}z)_{i-1,j} + \tilde{f}_{i-1,j} (u \rho \text{d}z)_{i-1,j+1} + \tilde{f}_{i,j} (u \rho \text{d}z)_{i,j} + \tilde{f}_{i,j+1} (u \rho \text{d}z)_{i,j+1}]. \end{aligned} \quad (14.150)$$

$$(14.151)$$

The effective Coriolis parameter \tilde{f} (equation (14.142)) is evaluated at the B-grid velocity point, which is the same as the C-grid vorticity point. For the advective metric frequency (14.147), we use the following discrete expression evaluated at the B-grid velocity point

$$\mathcal{M}_{i,j} = \text{d}h2\text{d}x \left(\frac{v_{i+1,j} + v_{i,j}}{2} \right) - \text{d}h1\text{d}y \left(\frac{u_{i,j+1} + u_{i,j}}{2} \right). \quad (14.152)$$

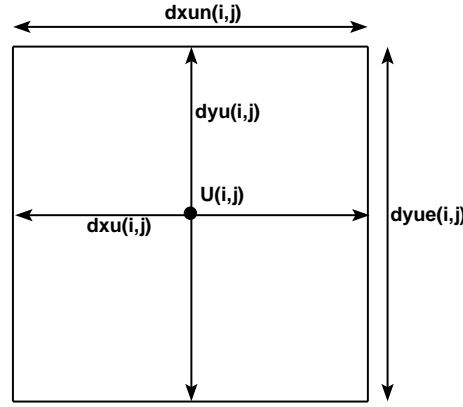


Figure 14.4: Time independent horizontal grid distances (meters) used for the B-grid velocity cell $U_{i,j}$ in MOM. $dxu_{i,j}$ and $dyu_{i,j}$ are the grid distances of the velocity cell in the generalized zonal and meridional directions, and $dau_{i,j} = dxu_{i,j} dyu_{i,j}$ is the area of the cell. The grid distance $dxue_{i,j}$ is the zonal width of the north face of a velocity cell, and $dyue_{i,j}$ is the meridional width of the east face. Note that the velocity point $U_{i,j}$ is not generally at the center of the velocity cell. Distances are functions of both i and j due to the use of generalized orthogonal coordinates.

We now introduce the C-grid advective mass transports (14.135) and (14.137) to write the C-grid Coriolis force per area as

$$(\tilde{f} v \rho dz)^{\text{x-eff coriolis}} \approx (1/4) \left[\tilde{f}_{i,j} \text{vhrho_nt}_{i,j} + \tilde{f}_{i,j} \text{vhrho_nt}_{i+1,j} + \tilde{f}_{i,j-1} \text{vhrho_nt}_{i,j-1} + \tilde{f}_{i,j-1} \text{vhrho_nt}_{i+1,j-1} \right] \quad (14.153)$$

$$-(\tilde{f} u \rho dz)^{\text{y-eff coriolis}} \approx -(1/4) \left[\tilde{f}_{i-1,j} \text{uhrho_et}_{i-1,j} + \tilde{f}_{i-1,j} \text{uhrho_et}_{i-1,j+1} + \tilde{f}_{i,j} \text{uhrho_et}_{i,j} + \tilde{f}_{i,j} \text{uhrho_et}_{i,j+1} \right]. \quad (14.154)$$

The Coriolis force and momentum advection for the C-grid in MOM are time discretized using the Adams-Bashforth scheme. As for the B-grid, multi-time discretizations means that the Coriolis force contributes to the discrete mechanical energy budget. Furthermore, the spatial averaging used in equations (14.153) and (14.154) means that the effective Coriolis force contributes locally to the discrete C-grid mechanical energy, regardless how time is discretized. Note that if using a single time for the Coriolis force², there will in fact be no contribution to global mechanical energy from the effective Coriolis force for the special case of spatially constant ρdz over each depth level. This property can be verified by shifting indices.

14.10.3 Comments

It is unsatisfying that whenever the Coriolis force is discretized using more than a single time level, it contributes to the discrete mechanical energy budget. Spatial averaging on the C-grid furthermore retains a non-vanishing contribution to energy at each grid point, regardless how time is discretized. Such properties of the discrete Coriolis force remain one of the many areas where discrete primitive equation modelling could be made more elegant.

14.11 B-grid kinetic energy advection

We now consider how kinetic energy is advected in the discrete case. We start from the Boussinesq manipulations, which are readily generalized to the non-Boussinesq.

²A single time for the Coriolis force is stable with a leap-frog, but unstable with MOM's staggered time scheme. But the leap-frog is not available for the C-grid in MOM.

There are two limitations of the manipulations presented here.

- Momentum advection is evaluated at a single time step, as is the case for a leap-frog version of MOM. However, for the preferred time staggered scheme, momentum advection is implemented according to a third order Adams-Bashforth (chapter 12 of [Griffies, 2004](#)). In this case, the results here are relevant when we take the scalar product of the horizontal velocity with just one of the three Adams-Bashforth terms contributing to the advection tendency. Even though the energy conversion properties are compromised, the utility of the kinetic energy conversion diagnostic remains. It is for this reason that we present these manipulations, as the diagnostics are useful when developing algorithms to identify numerical bugs.
- Velocity advection is discretized using second order centered advection. This is the original approach used by [Bryan \(1969\)](#), and remains the default approach in MOM.

The following is a generalization of material presented in [Griffies et al. \(2004\)](#).

14.11.1 B-grid momentum equation contribution from advection

The contribution to the momentum time tendency arising from second order centered advection fluxes on a B-grid is given by

$$\partial_t (\mathbf{u} \rho_o dzu)_{i,j} = -BDX_EU[uh_et FAX(\mathbf{u})] - BDY_NU[vh_nt FAY(\mathbf{u})] - \left[w_bu_{k-1} \left(\frac{\mathbf{u}_k + \mathbf{u}_{k-1}}{2} \right) - w_bu_k \left(\frac{\mathbf{u}_{k+1} + \mathbf{u}_k}{2} \right) \right]. \quad (14.155)$$

The forward algebraic averaging operators FAX and FAY are given in equations (14.42) and (14.43). They are used to estimate velocity on the velocity cell faces to construct the centered difference advective fluxes of velocity. MOM also uses the backward derivative operators

$$BDX_EU(A) = \frac{dyue_{i,j} A_{i,j} - dyue_{i-1,j} A_{i-1,j}}{dau_{i,j}} \quad (14.156)$$

$$BDY_NU(A) = \frac{dxue_{i,j} A_{i,j} - dxue_{i,j-1} A_{i,j-1}}{dau_{i,j}}. \quad (14.157)$$

These backward derivative operators act on fields defined at the east and north face of velocity cells, respectively (see Figure 14.4 for definitions of grid distances). Note that the contribution from the momentum advective frequency \mathcal{M} is discussed in Section 14.10.1. We focus in this section just with contributions from advective flux components.

14.11.2 Horizontal convergence

To get started, we consider the scalar product of the horizontal convergence term with the horizontal velocity \mathbf{u} , and integrate over the full ocean

$$\mathcal{A}_{\text{horz}} = - \sum_{i,j,k} dau \mathbf{u} \cdot [BDX_EU(uh_eu FAX(\mathbf{u})) + BDY_NU(vh_nu FAY(\mathbf{u}))]. \quad (14.158)$$

Note the use of thickness weighted advection velocity components uh_eu and vh_nu provides for the vertical grid increment dzu needed for the discrete volume integral.

14.11.3 Diagnosing the vertical transport for U-cells

Thickness weighted horizontal advective velocities uh_eu and vh_nu are defined in MOM by remapping the horizontal advective velocities uh_et and vh_nt , defined in Section 14.6.2, onto the velocity cell faces. The satisfy volume conservation via a U-grid version of the T-grid result (14.55)

$$w_bu_k = w_bu_{k-1} + BDX_EU(uh_eu_k) + BDY_NU(vh_nu_k) + \partial_t (dzu_k) - \mathcal{S}^{(V)} dzu_k. \quad (14.159)$$

In this equation, w_{bu} is the dia-surface advective velocity component defined at the bottom face of a velocity cell, and the volume source $S^{(V)}$ and tendency $\partial_t(dz u_k)$ have been mapped from their counterparts on the tracer grid.

To diagnose the vertical transport w_{bu} , we need to start at either the bottom or top of the ocean column, given a boundary condition. On the B-grid, the bottom of the bottom-most U-cell does not live on the ocean bottom, unless the ocean bottom is flat. Hence, a nontrivial mass or volume transport generally occurs through the bottom of a velocity cell column. That is, in general

$$w_{bu_{k=k_{bot}}} \neq 0, \quad (14.160)$$

which contrasts with the case on the T-cells (equations (14.57) and (14.113)).

A thorough discussion of this issue is provided in Section 22.3.3.2 of the MOM3 Manual (Pacanowski and Griffies, 1999). For present purposes, we note that it is sufficient to start the integration at the ocean surface and integrate downwards, just as for the tracer cells. Equivalently, we can use MOM's horizontal remapping operator to map w_{bt_k} to w_{bu_k} . Either way, the continuity equation (14.159) is maintained.

14.11.4 Discrete integration by parts on horizontal convergence

We now perform the discrete analog of integration by parts. For this purpose, expand the backwards derivative and average operators on the zonal flux terms, dropping the j, k labels for brevity

$$\begin{aligned} 2 \sum_i \text{dau} \mathbf{u}_i \cdot \text{BDX_EU}(\text{uh_eu FAX}(\mathbf{u})) &= \\ \sum_i \mathbf{u}_i \cdot [\text{dyue}_i \text{uh_eu}_i \mathbf{u}_{i+1} + \text{dyue}_i \text{uh_eu}_i \mathbf{u}_i & \\ - \text{dyue}_{i-1} \text{uh_eu}_{i-1} \mathbf{u}_i - \text{dyue}_{i-1} \text{uh_eu}_{i-1} \mathbf{u}_{i-1}] & \\ = \sum_i \mathbf{u}_i \cdot \mathbf{u}_i (\text{dyue}_i \text{uh_eu}_i - \text{dyue}_{i-1} \text{uh_eu}_{i-1}) & \\ + \sum_i \mathbf{u}_i \cdot (\text{dyue}_i \text{uh_eu}_i \mathbf{u}_{i+1} - \text{dyue}_{i-1} \text{uh_eu}_{i-1} \mathbf{u}_{i-1}). & \end{aligned} \quad (14.161)$$

Focus now on the second group of terms, where shifting sum labels leads to

$$\begin{aligned} \sum_{i=1}^{nx} \mathbf{u}_i \cdot \mathbf{u}_{i+1} \text{dyue}_i \text{uh_eu}_i - \sum_{i=1}^{nx} \mathbf{u}_i \cdot \mathbf{u}_{i-1} \text{dyue}_{i-1} \text{uh_eu}_{i-1} &= \\ \sum_{i=2}^{nx+1} \mathbf{u}_{i-1} \cdot \mathbf{u}_i \text{dyue}_{i-1} \text{uh_eu}_{i-1} - \sum_{i=1}^{nx} \mathbf{u}_i \cdot \mathbf{u}_{i-1} \text{dyue}_{i-1} \text{uh_eu}_{i-1} & \\ = \mathbf{u}_{nx} \cdot \mathbf{u}_{nx+1} \text{dyue}_{nx} \text{uh_eu}_{nx} - \mathbf{u}_1 \cdot \mathbf{u}_0 \text{dyue}_0 \text{uh_eu}_0. & \end{aligned} \quad (14.162)$$

This result vanishes for either solid wall or periodic boundary conditions. Similar manipulations apply for the meridional term, thus leading to

$$\begin{aligned} \mathcal{A}_{\text{horz}} &= - \sum \mathcal{K} (\text{dyue}_i \text{uh_eu}_i - \text{dyue}_{i-1} \text{uh_eu}_{i-1}) - \sum \mathcal{K} (\text{dxue}_j \text{vh_nu}_j - \text{dxue}_{j-1} \text{vh_nu}_{j-1}) \\ &= - \sum \text{dau} \mathcal{K} [\text{BDX_EU}(\text{uh_eu}) + \text{BDY_NU}(\text{vh_nu})], \end{aligned} \quad (14.163)$$

where

$$\mathcal{K}_{i,j,k} = \frac{1}{2} \mathbf{u}_{i,j,k} \cdot \mathbf{u}_{i,j,k} \quad (14.164)$$

is the discrete kinetic energy per mass carried by the horizontal flow.

14.11.5 Discrete integration by parts on the vertical convergence

Now focus on the vertical advection term, which takes the form

$$\begin{aligned}
 2\mathcal{A}_{\text{vert}} &= \sum_{k=1}^{kbot} \text{dau } \mathbf{u}_k \cdot [-\mathbf{w_bu}_{k-1} (\mathbf{u}_{k-1} + \mathbf{u}_k) + \mathbf{w_bu}_k (\mathbf{u}_k + \mathbf{u}_{k+1})] \\
 &= \sum_{k=1}^{kbot} \text{dau } \mathbf{u}_k \cdot \mathbf{u}_k (\mathbf{w_bu}_k - \mathbf{w_bu}_{k-1}) + \sum_{k=1}^{kbot} \text{dau } (\mathbf{w_bu}_k \mathbf{u}_{k+1} \cdot \mathbf{u}_k - \mathbf{w_bu}_{k-1} \mathbf{u}_k \cdot \mathbf{u}_{k-1}) \\
 &= 2 \sum_{k=1}^{kbot} \text{dau } \mathcal{K}_k (\mathbf{w_bu}_k - \mathbf{w_bu}_{k-1}) - \sum \text{dau } \mathbf{w_bu}_0 (\mathbf{u}_0 \cdot \mathbf{u}_1) + \sum \text{dau } \mathbf{w_bu}_{kbot} (\mathbf{u}_{kbot+1} \cdot \mathbf{u}_{kbot}).
 \end{aligned} \tag{14.165}$$

The horizontal velocity at $k = kbot + 1$ vanishes

$$\mathbf{u}_{kbot+1} = 0, \tag{14.166}$$

since $k = kbot + 1$ is interpreted as part of the solid earth. In contrast,

$$\mathbf{u}_{k=0} = \mathbf{u}_w \tag{14.167}$$

is the horizontal velocity of the fresh water. This velocity is often set equal to the surface ocean velocity $\mathbf{u}_w = \mathbf{u}_1$, yet MOM retains the option of providing a different value. This result then leads to

$$2\mathcal{A}_{\text{vert}} = 2 \sum_{k=1}^{kbot} \text{dau } \mathcal{K}_k (\mathbf{w_bu}_k - \mathbf{w_bu}_{k-1}) - \sum_{i,j} \text{dau } \mathbf{w_bu}_0 (\mathbf{u}_1 \cdot \mathbf{u}_w). \tag{14.168}$$

14.11.6 Final result for the Boussinesq case

Combining the results for $\mathcal{A}_{\text{horz}}$ and $\mathcal{A}_{\text{vert}}$ renders

$$\begin{aligned}
 \mathcal{A}_{\text{horz}} + \mathcal{A}_{\text{vert}} &= -(1/2) \sum_{i,j} \text{dau } \mathbf{w_bu}_0 (\mathbf{u}_1 \cdot \mathbf{u}_w) - \sum_{k=1}^{nk} \text{dau } \mathcal{K} [\text{BDX_EU}(\text{uh_eu}) + \text{BDY_NU}(\text{vh_nu}) + (\mathbf{w_bu}_{k-1} - \mathbf{w_bu}_k)] \\
 &= -(1/2) \sum_{i,j} \text{dau } \mathbf{w_bu}_0 (\mathbf{u}_1 \cdot \mathbf{u}_w) + \sum_{i,j,k} \text{dau } \mathcal{K} [\partial_t (\text{dzu}_k) - \mathcal{S}^{(V)} \text{dzu}_k],
 \end{aligned} \tag{14.169}$$

where we applied volume conservation over each U-cell as given by equation (14.159).

14.11.7 Non-Boussinesq kinetic energy advection

We now consider the conversion of kinetic energy advection in the discret non-Boussinesq discete. For this purpose, consider

$$\mathcal{A}_{\text{horz}} = - \sum_{i,j,k} \text{dau } \text{dzu } \mathbf{u} \cdot [\text{BDX_EU}(\text{uhrho_eu FAX}(\mathbf{u}))/\text{dzu} + \text{BDY_NU}(\text{vhrho_nu FAX}(\mathbf{u}))/\text{dzu}]. \tag{14.170}$$

Thickness weighted and density weighted horizontal advective velocities uhrho_eu and vhrho_nu are defined in MOM by remapping the horizontal advective velocities uhrho_et and vhrho_nt , defined by equations (14.106) and (14.109), onto the velocity cell faces. These horizontal transports satisfy continuity via a U-grid version of the T-grid result (14.110)

$$\text{wrho_bu}_k = \text{wrho_bu}_{k-1} + \text{BDX_EU}(\text{uhrho_eu}_k) + \text{BDY_NU}(\text{vhrho_nu}_k) + \partial_t (\text{rho_dzu})_k - \mathcal{S}^{(M)} \text{rho_dzu}_k \tag{14.171}$$

where the mass source has been mapped from the tracer to the velocity grid. In this equation, $w\rho_{bu}$ is the density weighted dia-surface advective velocity component defined at the bottom face of a velocity cell. As in the Boussinesq case, this vertical transport is diagnosed using the continuity equation, or equivalently via the MOM remap operator.

In general, results for the Boussinesq case transparently generalize to the non-Boussinesq case, which allows us to write by inspection

$$\begin{aligned} \mathcal{A} &= -(1/2) \sum \text{dau } w\rho_{bu_0} (\mathbf{u}_1 \cdot \mathbf{u}_w) \\ &\quad - \sum_{k=1}^{nk} \text{dau } \mathcal{K} [\text{BDX_EU}(\text{uhrho_eu}) + \text{BDY_NU}(\text{vhrho_nu}) + (w\rho_{bu_{k-1}} - w\rho_{bu_k})] \\ &= -(1/2) \sum \text{dau } w\rho_{bu_0} (\mathbf{u}_1 \cdot \mathbf{u}_w) + \sum_{i,j,k} \text{dau } \mathcal{K} [\partial_t (\rho_{dz} u_k) - \mathcal{S}^{(M)} \rho_{dz} u_k]. \end{aligned} \quad (14.172)$$

14.12 C-grid kinetic energy advection

The C-grid manipulations are similar to the B-grid, though we start from a slightly different form of the momentum advection as appropriate for the C-grid.

14.12.1 C-grid momentum equation contribution from advection

The contribution to the zonal momentum time tendency arising from second order centered advection on a C-grid is given by (see equation (55) of [Blumberg and Mellor \(1987\)](#))

$$\partial_t (u \rho \, dz \, te)_{i,j} = - \left(\frac{F^{(x)}(u)_{i,j,k} - F^{(x)}(u)_{i-1,j,k}}{dx \, te_{i,j}} \right) - \left(\frac{F^{(y)}(u)_{i,j,k} - F^{(y)}(u)_{i,j-1,k}}{dy \, te_{i,j}} \right) - (F^{(z)}(u)_{i,j,k-1} - F^{(z)}(u)_{i,j,k}), \quad (14.173)$$

where the flux components are given by

$$F^{(x)}(u)_{i,j,k} = \frac{(\text{uhrho_et}_{i+1,j} + \text{uhrho_et}_{i,j}) (u_{i+1,j,k} + u_{i,j,k})}{4} = \text{FAX}(\text{uhrho_et}) \text{FAX}(u) \quad (14.174)$$

$$F^{(y)}(u)_{i,j,k} = \frac{(\text{vhrho_nt}_{i,j} + \text{vhrho_nt}_{i+1,j}) (u_{i,j,k} + u_{i,j+1,k})}{4} = \text{FAX}(\text{vhrho_nt}) \text{FAY}(u) \quad (14.175)$$

$$F^{(z)}(u)_{i,j,k} = \frac{(\text{wrho_bt}_{i,j,k} + \text{wrho_bt}_{i+1,j,k}) (u_{i,j,k} + u_{i,j,k+1})}{4} = \text{FAX}(\text{wrho_bt}) \text{FAZ}(u). \quad (14.176)$$

We offer the following observations.

- We purposefully did not write the expression (14.173) in terms of MOM's derivative operators, since the indexing convention is slightly non-standard for MOM, given the MOM was originally written solely for a B-grid.
- An alternative expression for the zonal flux is $\text{FAX}(\text{uhrho_et } u)$. But with uniform zonal resolution, this form will lead to a time tendency proportional to $(\text{uhrho_et } u)_{i+1} - (\text{uhrho_et } u)_{i-1}$, which skips the central term $(\text{uhrho_et } u)_i$. Skipping the central term allows for a computational null mode in which the even and odd grid terms decouple, and this behaviour is undesirable.
- The contribution from the momentum advective frequency \mathcal{M} is discussed in Section 14.10.2. We focus in this section on contributions from advective flux components.

The meridional momentum tendency on the C-grid is given by (see equation (56) of [Blumberg and Mellor \(1987\)](#))

$$\partial_t (v \rho \, dz \, tn)_{i,j} = - \left(\frac{G^{(x)}(v)_{i,j,k} - G^{(x)}(v)_{i-1,j,k}}{dx \, tn_{i,j}} \right) - \left(\frac{G^{(y)}(v)_{i,j,k} - G^{(y)}(v)_{i,j-1,k}}{dy \, tn_{i,j}} \right) - (G^{(z)}(v)_{i,j,k-1} - G^{(z)}(v)_{i,j,k}), \quad (14.177)$$

where the flux components are given by

$$G^{(x)}(v)_{i,j,k} = \frac{(\text{uhrho_et}_{i,j} + \text{uhrho_et}_{i,j+1})(v_{i,j,k} + v_{i+1,j,k})}{4} = \text{FAY}(\text{uhrho_et}) \text{FAX}(v) \quad (14.178)$$

$$G^{(y)}(v)_{i,j,k} = \frac{(\text{vhrho_nt}_{i,j} + \text{vhrho_nt}_{i,j+1})(v_{i,j,k} + v_{i,j+1,k})}{4} = \text{FAY}(\text{vhrho_nt}) \text{FAY}(v) \quad (14.179)$$

$$G^{(z)}(v)_{i,j,k} = \frac{(\text{wrho_bt}_{i,j,k} + \text{wrho_bt}_{i,j+1,k})(v_{i,j,k} + v_{i,j,k+1})}{4} = \text{FAY}(\text{wrho_bt}) \text{FAZ}(v). \quad (14.180)$$

14.12.2 Energetic manipulations not generally useful

We have attempted to follow the B-grid procedure using discrete integration by parts to reveal an analog to the continuum results of Sections 14.2.3 and 14.2.4. However, in a manner similar to that encountered for the Coriolis Force (Section 14.10.2), useful manipulations are available only when grid factors are assumed horizontally constant. Given that such restrictions are rarely realized, we do not pursue the manipulations. We have thus not been able to derive a suitable advection diagnostic for the C-grid.

Chapter 15

ADVECTION VELOCITY AND HORIZONTAL REMAPPING FOR THE B-GRID

Contents

15.1	General considerations	221
15.1.1	Two main issues	222
15.1.2	Constraints for discrete vertical velocities	222
15.2	Remapping operators for horizontal fluxes	222
15.2.1	Uniformly distributed volume flux across a face	223
15.2.2	Lever-rule and the horizontal remapping operators	223
15.3	Remapping operator for vertical fluxes	224
15.4	Remapping error	225
15.4.1	Linear grids	227
15.4.2	Nonlinear grids	227
15.5	Subtleties at the southern-most row	228

The purpose of this chapter is to discuss the computation of advection velocity components for the advection of momentum on the B-grid. This material follows from that presented in Chapter 14 which derived advection velocity components based on energetic arguments. Linear remapping operators provide the means to compute the advection velocity components for the advection of momentum, and we detail these operators in this chapter. In particular, these operators map tracer cell advection velocity components, prescribed according to the arguments of Sections 14.6 and 14.8, to the B-grid velocity cell, as well as to map selected other fields. The material in this chapter is taken largely from the MOM4.0 manual of Griffies et al. (2004), which employs a Boussinesq formulation. Generalization to non-Boussinesq option is trivial, with introduction of *in situ* density factors multiplying the thickness factor. Note that similar treatments for C-grid advection of momentum is not needed, with the formulation given in Chapter 14 sufficient.

The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_advection_velocity.F90

15.1 General considerations

Advective fluxes are fundamental to the Eulerian evolution of tracer and momentum. How these fluxes are discretized represents a basic problem in computational fluid dynamics. Notably, because of the interpretation of model velocity discussed in Griffies (2004), there is no distinction between the advective fluxes

for the Boussinesq and non-Boussinesq versions of MOM: they are computed using the same numerical considerations detailed in this chapter.

15.1.1 Two main issues

There are two considerations required to compute advective fluxes of tracer or momentum. First, there is the question of how to compute the advective velocity. Such is the focus of this chapter. For computing fluxes across cell faces, the three components to the advective velocity must be known on the corresponding face of the tracer and velocity cells. However, on the B-grid, both horizontal prognostic velocity components are placed at the velocity cell point, not at the cell faces. Hence, an averaging operation must be prescribed to diagnose the horizontal advective velocity components from the prognostic B-grid velocity. MOM computes the horizontal components of the advection velocity on the faces of T-cells in a manner necessitated by equating pressure work to buoyancy (see Chapter 14). The vertical advective velocity component is then diagnosed at the bottom face of the tracer cell, based on the needs of volume or mass conservation across the tracer cell (see Chapter 14 or Griffies (2004)). Computing the advective velocity on the faces of the velocity cell remains to be determined, and that is the main technical subject of this chapter.

Once the advective velocity is computed on the cell faces, it remains to approximate the tracer and momentum values on these faces for use in constructing the advective tracer and momentum fluxes. There are many different approaches available. As with previous versions of the GFDL ocean model, MOM chooses to compute the advective flux of momentum according to the requirements of energetic consistency described in Chapter 14. These constraints necessitate a second order centered approach, as in Bryan (1969). The advective flux of tracer, however, is not so constrained and there are hence many options available, some of which are detailed in *The MOM3 Manual* of Pacanowski and Griffies (1999).

15.1.2 Constraints for discrete vertical velocities

One important constraint for self-consistency of the discretization is that the vertical velocity at the T-cell bottom topography must vanish: $w_{bt_{i,j,k=N_k}} = 0$, since the T-cell top and bottom faces are horizontally oriented. A vanishing bottom velocity on T-cells is necessitated by the requirements of volume or mass conservation (see Griffies (2004)). Many ocean models choose to set $w_{bt_{i,j,k=N_k}} = 0$. However, MOM chooses to start from the ocean surface and integrate the continuity equation downwards. Verification that the computed $w_{bt_{i,j,k=N_k}}$ indeed vanishes has been found to be a very useful check on code integrity. Relatedly, for a flat bottomed ocean $w_{bu_{i,j,k=N_k}} = 0$. However, with topography, w_{bu} is generally nonzero at the bottom, since the bottom on velocity cells is not flat. Section 22.3 in *The MOM3 Manual* of Pacanowski and Griffies (1999) details this point.

Furthermore, since the interior of the ocean domain uses constant cell thicknesses, in a Boussinesq model volume should be conserved (the ocean surface conserves volume when also incorporating the possibly nonzero fresh water fluxes). Hence, integrating $w_{bt_{i,j,k}}$ across a particular depth $k > 1$ should leave no net volume flux upward or downward: $\sum_{i,j} dx_{i,j} dy_{i,j} w_{bt_{i,j,k}} = 0$ for all levels k .

Finally, volume conservation warrants the MOM approach for diagnosing surface height on the U-cell, η^u , according to an area weighted average of the surrounding T-cell heights η^t , instead of using the minimum operation used in MOM3 and described in the Griffies et al. (2001) paper. This issue is relevant for the Boussinesq and non-Boussinesq versions of MOM. We visit this issue in Section 15.3.

15.2 Remapping operators for horizontal fluxes

As stated in Section 15.1, MOM computes the horizontal components of the T-cell advection velocity in a manner necessitated by equating pressure work to buoyancy (see Chapter 14). The vertical component is diagnosed based the needs of continuity. Hence, we assume the T-cell advective velocity components are known. We thus need to determine the corresponding advective velocity on the face of velocity cells.

Advective velocities represent fluxes of volume per unit area. There are three remapping operators that take discrete volume fluxes defined at tracer points or sides of tracer cells, to discrete fluxes defined at velocity points or sides of velocity cells. Although MOM is generally non-Boussinesq, we use the ideas of volume conservation to generate algorithms for coupling advective velocities on the sides of tracer cells

to those on the sides of velocity cells. Here, we describe the linear remapping operator taking horizontal advective velocities centered on the face of a tracer cell to the corresponding face of a velocity cell.

15.2.1 Uniformly distributed volume flux across a face

Reference to Figure 15.1 reveals four eastward fluxes of volume per area leaving a tracer cell that surround the single flux per volume leaving the corresponding velocity cell. The flux leaving a tracer cell is denoted by Et in the figure, which is a shorthand for the model's thickness weighted advective velocity $uh.et$, with the thickness factor dropped since we are concerned here with fluxes at a fixed depth. Eu denotes the corresponding eastward flux leaving the velocity cell, and this flux is to be determined in terms of the surrounding Et and appropriate grid distances.

We assume that along the face of a tracer cell, volume leaves through the face with a uniform distribution. Hence, the volume per unit length per time passing across the meridional face through the velocity point $U_{i,j}$ is given by

$$Et(i, j) \text{dus}(i, j) + Et(i, j + 1) \text{dun}(i, j), \quad (15.1)$$

where the distances dus and dun are lengths along sides of the four quarter-cells comprising a single velocity cell (Figure 15.2). Likewise, the volume per unit length per time passing across the meridional face through the velocity point $U_{i+1,j}$ is given by

$$Et(i + 1, j) \text{dus}(i + 1, j) + Et(i + 1, j + 1) \text{dun}(i + 1, j), \quad (15.2)$$

and the volume per unit length per time passing across the eastern face of the velocity cell $U_{i,j}$ is given by

$$Eu(i, j) \text{dytn}(i + 1, j), \quad (15.3)$$

where Eu is to be determined in terms of Et , and the grid distance dytn is the meridional distance between tracer points, as defined in Figure 15.3.

15.2.2 Lever-rule and the horizontal remapping operators

We now employ linear interpolation, or a lever-rule average, to construct the volume per time passing across the east face of the $U_{i,j}$ cell, thus leading to

$$Eu(i, j) \text{dytn}(i + 1, j) \text{dxtn}(i + 1, j) = [Et(i, j) \text{dus}(i, j) + Et(i, j + 1) \text{dun}(i, j)] \text{duw}(i + 1, j) \\ [Et(i + 1, j) \text{dus}(i + 1, j) + Et(i + 1, j + 1) \text{dun}(i + 1, j)] \text{due}(i, j),$$

where dxtn is the zonal distance along the north face of a tracer cell (Figure 15.4). Solving for Eu leads to the remapping operator

$$\begin{aligned} Eu(i, j) &= \text{REMAP_ET_TO_EU}(Et)(i, j) \\ &\equiv [Et(i, j) \text{dus}(i, j) \text{duw}(i + 1, j) + Et(i, j + 1) \text{dun}(i, j) \text{duw}(i + 1, j) \\ &+ Et(i + 1, j) \text{dus}(i + 1, j) \text{due}(i, j) + Et(i + 1, j + 1) \text{dun}(i + 1, j) \text{due}(i, j)] \\ &\quad \text{datnr}(i + 1, j), \end{aligned} \quad (15.4)$$

where datnr is the reciprocal area at the north face of a T-cell given by

$$\text{datnr}(i, j) = \frac{1}{\text{dxtn}(i, j) \text{dytn}(i, j)} \quad (15.5)$$

Analogous considerations lead to the remapping operator that takes a volume flux Nt defined at the north face of T-cells to a flux leaving the north face of U-cells

$$\begin{aligned} Nu(i, j) &= \text{REMAP_NT_TO_NU}(Nt)(i, j) \\ &\equiv [Nt(i, j) \text{duw}(i, j) \text{dus}(i, j + 1) + Nt(i + 1, j) \text{due}(i, j) \text{dus}(i, j + 1) \\ &+ Nt(i, j + 1) \text{duw}(i, j + 1) \text{dun}(i, j) + Nt(i + 1, j + 1) \text{due}(i, j + 1) \text{dun}(i, j)] \\ &\quad \text{dater}(i, j + 1). \end{aligned} \quad (15.6)$$

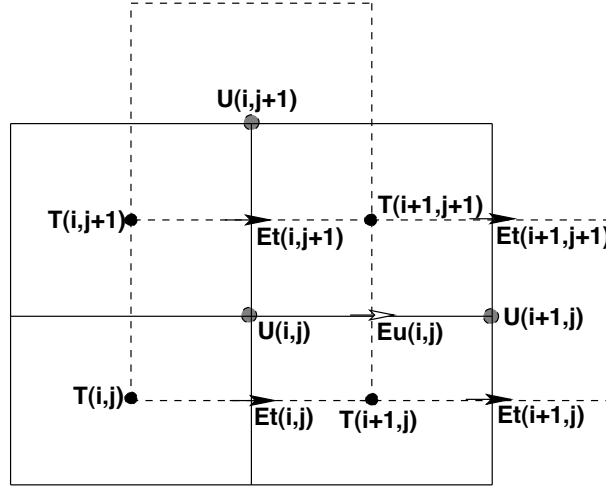


Figure 15.1: Schematic representation of the remapping function REMAP_ET_TO_EU defined by equation (15.4). This function is used to remap a horizontal flux of volume defined at the east face of T-cells (denoted by Et in this figure) onto a horizontal flux of volume defined at the east face of U-cells (denoted by Eu in this figure). The four fluxes $Et(i, j)$, $Et(i + 1, j)$, $Et(i, j + 1)$, and $Et(i + 1, j + 1)$ are used to construct the flux $Eu(i, j)$.

In this expression, $dater$ is the reciprocal area at the east face of a T-cell given by

$$dater(i, j) = \frac{1}{dxte(i, j) dyte(i, j)}, \quad (15.7)$$

$dxte$ is the zonal distance between the T-cell points (Figure 15.3) and $dyte$ is the meridional distance along the east face of the T-cell (Figure 15.4).

15.3 Remapping operator for vertical fluxes

We now consider the remapping taking vertical volume fluxes passing across the bottom face of tracer cells to the bottom face of velocity cells. This operator also maps surface height from T-cells to U-cells. This remapping is distinguished from the horizontal remapping in that there is no analogous lever-rule step. The distinction boils down to noting that the vertical remapping REMAP_BT_TO_BU moves vertical fluxes horizontally, whereas the east and north remapping operators move horizontal fluxes horizontally.

Reference to Figures 15.2 and 15.3, and again assuming fluxes are distributed uniformly across a cell face, indicates that the vertical flux of volume per unit length passing across the southern face of the velocity cell $U_{i,j}$ is given by

$$Bt(i, j) dte(i, j) + Bt(i + 1, j) dtw(i + 1, j), \quad (15.8)$$

the vertical flux of volume per unit length passing across the northern face of the velocity cell $U_{i,j}$ is given by

$$Bt(i, j + 1) dte(i, j + 1) + Bt(i + 1, j + 1) dtw(i + 1, j + 1), \quad (15.9)$$

and the vertical flux of volume passing through the velocity cell is given by

$$Bu(i, j) dxu(i, j) dyu(i, j). \quad (15.10)$$

Assuming that the total flux passing through the velocity cell is equivalent to that passing across the north-

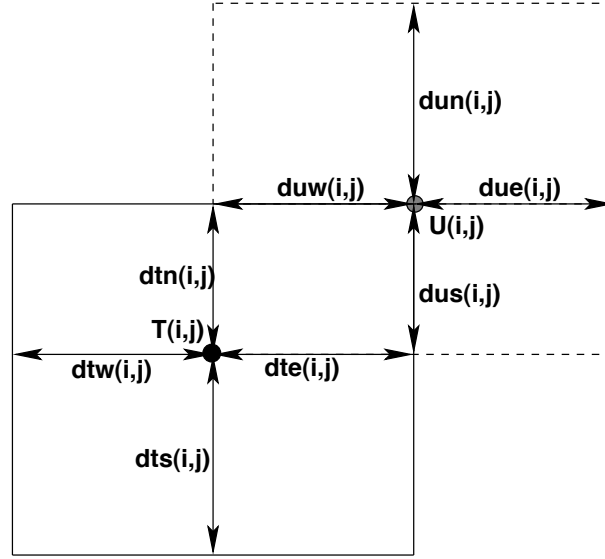


Figure 15.2: Time independent horizontal grid distances (meters) used for the tracer cell $T_{i,j}$ and velocity cell $U_{i,j}$ in MOM. These “quarter-cell” distances are refined relative to those shown in Figures 15.4 and 15.5, and they are needed for the remapping between T and U cells when computing advection velocities. All distances are functions of both i and j due to the use of generalized orthogonal coordinates. Comparing with Figures 15.4 and 15.5 reveals the identities $dtw(i, j) + dte(i, j) = dx_t(i, j)$, $dts(i, j) + dtn(i, j) = dy_t(i, j)$, $duw(i, j) + due(i, j) = dx_u(i, j)$, and $dus(i, j) + dun(i, j) = dy_u(i, j)$.

ern plus southern parts of the cell leads to

$$Bu(i, j) dx_u(i, j) dy_u(i, j) = [Bt(i, j) dte(i, j) + Bt(i + 1, j) dtw(i + 1, j)] dus(i, j) \\ + [Bt(i, j + 1) dte(i, j + 1) + Bt(i + 1, j + 1) dtw(i + 1, j + 1)] dun(i, j).$$

Solving for Bu yields the vertical remapping operator

$$Bu(i, j) = \text{REMAP_BT_TO_BU}(Bt)(i, j) \\ \equiv [Bt(i, j) dte(i, j) dus(i, j) + Bt(i + 1, j) dtw(i + 1, j) dus(i, j) \\ + Bt(i, j + 1) dte(i, j + 1) dun(i, j) + Bt(i + 1, j + 1) dtw(i + 1, j + 1) dun(i, j)] \\ \text{daur}(i, j) \quad (15.11)$$

with daur the reciprocal area of the U-cell

$$\text{daur}(i, j) = \frac{1}{dx_u(i, j) dy_u(i, j)} \quad (15.12)$$

15.4 Remapping error

There are two ways to compute the vertical velocity on the velocity cell. The first method is to compute this velocity according to the requirements of continuity over the velocity cell, using the convergence of the remapped horizontal advective velocities entering the velocity cell. The second method is to use the vertical remap operator REMAP_BT_TO_BU to move the vertical velocity on the tracer cells to the velocity cells. The result of these two approaches is identical *when* the tracer and velocity grids are related by a linear average operator, as is the case for a spherical grid. The need to maintain a linear relation between the tracer and velocity grids is based on the use of linear methods to derive the remapping operators.

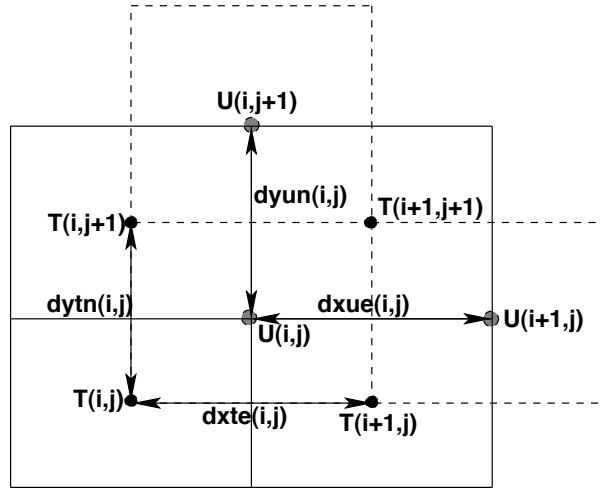


Figure 15.3: Time independent horizontal grid distances (meters) setting the spacing between tracer and velocity points in MOM. All distances are functions of both i and j due to the use of generalized orthogonal coordinates. When these distances are combined with those in Figures 15.4 and 15.5, and the quarter-cell distances given in Figure 15.2, we then have full information about the discrete horizontal T and U cells on the model grid. Note there is some redundancy with the distances defined in Figures 15.4 and 15.5, where we have $dytn(i, j) = dyue(i-1, j)$, $dxte(i, j) = dxun(i, j-1)$, $dxue(i, j) = dxtn(i+1, j)$, and $dyun(i, j) = dyte(i, j+1)$. Additionally, comparison with Figure 15.2 leads to the identities $dyun(i, j) = dun(i, j) + dus(i, j+1)$, $dxue(i, j) = due(i, j) + duw(i+1, j)$, $dytn(i, j) = dtn(i, j) + dts(i, j+1)$, and $dxte(i, j) = dte(i, j) + dtw(i+1, j)$.

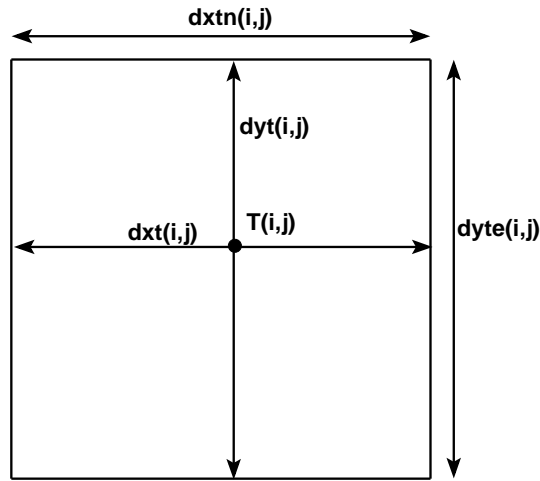


Figure 15.4: Time independent horizontal grid distances (meters) used for the tracer cell $T_{i,j}$ in MOM. $dxt_{i,j}$ and $dyt_{i,j}$ are the grid distances of the tracer cell in the generalized zonal and meridional directions, and $dat_{i,j} = dxt_{i,j} dyt_{i,j}$ is the area of the cell. The grid distance $dxtn_{i,j}$ is the zonal width of the north face of a tracer cell, and $dyte_{i,j}$ is the meridional width of the east face. Note that the tracer point $T_{i,j}$ is not generally at the center of the tracer cell. Distances are functions of both i and j due to the use of generalized orthogonal coordinates.

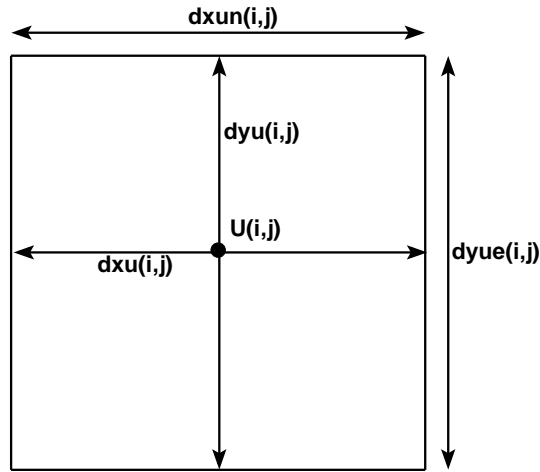


Figure 15.5: Time independent horizontal grid distances (meters) used for the velocity cell $U_{i,j}$ in MOM. $dxu_{i,j}$ and $dyu_{i,j}$ are the grid distances of the velocity cell in the generalized zonal and meridional directions, and $dau_{i,j} = dxu_{i,j} dyu_{i,j}$ is the area of the cell. The grid distance $dxun_{i,j}$ is the zonal width of the north face of a velocity cell, and $dyue_{i,j}$ is the meridional width of the east face. Note that the velocity point $U_{i,j}$ is not generally at the center of the velocity cell. Distances are functions of both i and j due to the use of generalized orthogonal coordinates.

15.4.1 Linear grids

MOM computes a diagnostic that examines the differences between the two approaches for computing vertical advective velocities. It reports the difference as a “remapping error.” If the numerical discretization is self-consistent, then the remapping error for a spherical grid will be roundoff, with values on the order of $10^{-20} \text{ m s}^{-1}$ common. Therefore, with a spherical grid, the remapping error provides a check on the self-consistency of the grid distances and the remapping operators. Effectively, what is done is to check that volume is conserved with the remapping operators. Even when running a non-Boussinesq model, the remapping operators are constructed to respect volume conservation.

15.4.2 Nonlinear grids

For a grid defined via a nonlinear transformation of the spherical grid, such as the bipolar region of the tripolar grid, the grid no longer maintains a linear relation between tracer and velocity cell distances. The result is a nontrivial remapping error. This error can be reduced by defining new remapping operators that account for a generally nonlinear relation between tracer and velocity grid distances. Such remains to be done.

One consequence of the nonzero remapping error is that for a flat bottom model in regions where the grid distances are nonlinearly related, $w_{bu_{i,j,k=Nk}}$ does not vanish, even though continuity is maintained for all the grid cells. The problem is that $w_{bu_{i,j,k=0}}$ is defined by

$$w_{bu_{i,j,k=0}} = \text{REMAP_BT_TO_BU}(w_{bt_{i,j,k=0}}). \quad (15.13)$$

For nonlinear grids, the linear operator REMAP_BT_TO_BU results in a slightly different value for $w_{bu_{i,j,k=0}}$ than would result from an integration of the continuity equation upwards from the bottom, assuming $w_{bu_{i,j,k=Nk}} = 0$. Because the vertical advective flux at the ocean bottom is masked so that no momentum will spuriously leak out the bottom of the ocean, having $w_{bu_{i,j,k=Nk}}$ slightly nonzero is of no consequence. Nonetheless, it would be more satisfying to have a general remapping operator to clean-up this issue.

15.5 Subtleties at the southern-most row

Consider the special case of $j = 0$ in Figure 15.1. This row is strictly south of the southern-most latitude comprising the computational domain of the model. However, there is a subtlety related to the treatment of the eastward volume flux leaving the velocity cell $U_{i,j=0}$. That is, since this cell straddles the tracer cells $T_{i,j=0}$ and $T_{i,j=1}$, it contains some portion that is within the computational domain. Thus, the eastward volume flux leaving this cell is nonzero, as it is comprised of weighted average of the four surrounding eastward fluxes leaving the tracer cells. Because the remapping function (15.4) is normalized with the area $\text{datn} = \text{dxtndytn}$ for $j = 0$, it is necessary to know the grid factors $\text{dxtn}_{i,j=0}$ and $\text{dytn}_{i,j=0}$. In particular, $\text{dytn}_{i,j=0}$ is the distance between the computed tracer point $T_{i,j=1}$ and the tracer point $T_{i,j=0}$ that lives outside the computational domain.

The need to know $\text{dytn}_{i,j=0}$ presents a problem with the MOM method for computing grid specifications. Grids in MOM are computed in two steps. First, there is a preprocessing step whereby grid factors are computed in a generic manner compatible with other models used at GFDL. This step knows nothing about halo regions, so it only computes grid information over the computational domain. The result of this step is a NetCDF grid specification file. The second step is to read the grid specification file into MOM and translate the generic grid information into grid arrays used by MOM. Since there is no halo information contained in the grid specification file, we cannot unambiguously specify values for the grid outside the computational domain. And because we need $\text{dytn}_{i,j=0}$ to be known consistently with the values for $\text{dytn}_{i,j}$ with $j > 0$, we cannot simply fill $\text{dytn}_{i,j=0}$ with an arbitrary placeholder. If we do so, then the remapping function used to compute the eastward flux leaving the velocity cell $U_{i,j=0}$ will be incorrect, thus compromising the vertical velocity leaving $U_{i,j=0}$. The symptom will be most notable in spuriously large values of the vertical velocity on the velocity cell at the computational row $j = 1$, as well as huge remapping errors at $j = 1$.

There are three solutions to this problem. First, we could extend the definition of the grid within the grid specification file to include the extra $j = 0$ row. This solution has been rejected since it adds an extra calculation that is specific to the northeast B-grid used in MOM. As the grid specification file is designed for use by all grid point models, it is not desirable to corrupt it with special cases. The second solution is to require the southern-most row in MOM to be filled with land. This solution is arguably inelegant, and it has indeed prompted some debate with the MOM developers. Yet this is the solution used in the GFDL ice model, which is also on a B-grid, and so it has been the most popular solution thus far in MOM when aiming to couple MOM to other models. The third solution is to extend the grid southward within MOM after reading in the grid specification file. This solution is appropriate if we can assume a spherical grid in the southern part of the domain, as true in most cases.

Chapter 16

OPEN BOUNDARY CONDITIONS FOR THE B-GRID

Contents

16.1	Introduction	230
16.2	Types of open boundary conditions	231
16.2.1	Open boundaries in the Arakawa B-grid	231
16.2.1.1	Notation	232
16.2.1.2	Boundary conditions for the sea level	232
16.2.1.3	Boundary conditions for tracers	232
16.2.1.4	Boundary conditions for velocity	232
16.2.2	Sommerfeld radiation condition	232
16.2.3	Clamped boundary conditions	233
16.2.4	No gradient boundary conditions	233
16.2.5	Interior cell no gradient boundary conditions	234
16.2.6	Enhanced friction and diffusion near the boundary	234
16.3	Implementation of sea level radiation conditions	234
16.3.1	Sign convention for the phase speed	234
16.3.2	Gravity wave radiation condition for the phase speed	234
16.3.3	Orlanski radiation condition for the phase speed	235
16.3.4	Camerlengo and O'Brien radiation condition	235
16.3.5	Radiation condition after Miller & Thorpe	235
16.3.6	Raymond and Kuo radiation condition	236
16.3.7	The IOW-radiation condition	236
16.3.8	Phase speed smoothing	237
16.3.9	Relaxation to data	237
16.3.9.1	Relaxation towards prescribed profiles	237
16.3.9.2	Relaxation of the sea level average - conservation of geostrophic currents	238
16.3.9.3	Variable relaxation for incoming and outgoing waves	238
16.4	OBC for tracers	238
16.4.1	Reduced tracer equations at open boundaries	238
16.4.2	Upstream advection of tracers near the boundary	239
16.4.3	Relaxation towards external data	240
16.4.4	Flow relaxation scheme of Martinsen and Engedahl	240
16.4.5	Radiation conditions	241

16.4.6	Vertical mixing and viscosity co-efficients	241
16.4.7	Enhanced horizontal mixing and viscosity co-efficients	241
16.5	The namelist obc_nml	241
16.6	Topography generation - Preparation of boundary data	243
16.6.1	Topography generation with open boundaries	243
16.6.2	Preparation of input data files	243
16.6.3	Consistency of input data and model configuration	245
16.6.3.1	The sea level in external data and the model zero level	245
16.6.3.2	The sea level and the problem of air pressure	245

The purpose of this chapter is to present the method used for the B-grid version of MOM to prescribe open boundary conditions (OBCs), with further documentation and examples presented by [Herzfeld et al. \(2011\)](#). The main point to take from this chapter, it is the following:

The numerical schemes for OBCs depend on details of the model setup. Hence, MOM has many options.

This chapter was written by Mike Herzfeld, Martin Schmidt, and Stephen Griffies. The algorithm and code developers for the MOM OBC are

Mike.Herzfeld@csiro.au
 Martin.Schmidt@io-warnemuende.de
 Zhi.Liang@noaa.gov
 Matthew.Harrison@noaa.gov

Please email them directly for queries about the documentation or the OBC code.

The following MOM modules are directly connected to the material in this chapter:

ocean_core/ocean_obc.F90
 ocean_core/ocean_obc_barotropic.F90

CAVEAT: AS OF MAY 2012, MOM'S OPEN BOUNDARY CONDITION HAS YET TO BE PORTED TO THE C-GRID. OPEN BOUNDARY CONDITIONS ARE ONLY AVAILABLE USING THE B-GRID OPTION. IMPLEMENTING THE OPEN BOUNDARY CONDITIONS FOR THE C-GRID VERSION OF MOM IS A HIGH PRIORITY.

16.1 Introduction

Numerical circulation models of marginal seas with biological, chemical and sediment dynamic components require a high model resolution and involve a large number of variables. Working with regional models is one method to meet this challenge with a reasonable amount of computer resource consumption. Mostly, the exchange of mass, heat, momentum and dissolved or suspended matter with the outer ocean is important. At the model boundary an open boundary condition (OBC) must apply, which permits flux out of - and into the model area.

This chapter describes the numerical schemes implemented for this purpose in MOM. They are described in detail in the corresponding literature, in particular in [Herzfeld et al. \(2011\)](#). It seems, that an universal open boundary condition suitable for all kinds of regional models does not exist. Hence, we have chosen to implement several schemes, which can be selected and modified by namelist parameters.

Open boundary conditions for a regional model is a complex problem. To be more specific, consider a large model ocean, subdivided by a virtual boundary into a western and an eastern sub-basin. Wind forcing, heat flux or fresh water flux in the eastern subbasin drives elevation of the sea surface, currents and changes in the density field as well. The information on such events in the eastern part is transmitted to the west by waves, at large time scales also by advection. If the virtual boundary is replaced with the open western boundary of a regional submodel of the eastern subbasin, the results of the regional submodel and

those of the larger model must be the same. Hence, waves generated in the eastern subbasin must be able to pass an open boundary without reflection and refraction, just as if it was not there. In the same manner, processes forced in the western subbasin influence the eastern part by waves too. If this is of importance for the eastern model part, the western open boundary condition must generate these waves.

Hence, the required boundary condition is solution of the hydrodynamic equations at the boundary itself and is basically unknown. Simplifying assumptions on the nature of the flow near the boundary are needed to close the numerical schemes at the boundary. For ocean models many different methods are known. Here we confine ourselves to methods, which combine a radiation condition, to facilitate outward directed wave propagation through open boundaries in combination with relaxation to prescribed values of ocean variables, to simulate the influences from outside the model domain. Relaxation of boundary values helps also to eliminate numerical errors of the boundary scheme and to prevent the model from divergence by the accumulation of numerical errors over longer model integration time.

16.2 Types of open boundary conditions

The purpose of this section is to describe the sorts of boundary conditions implemented in MOM. For realistic applications the numerical solution near a boundary is always a superposition of outgoing and incoming waves, which cannot be separated. Applying the radiation condition and relaxation of boundary values to this complex variables, turns OBC into a mathematically ill-posed problem and there is no universally perfect scheme for open boundary conditions. Hence, often the OBC configuration must be established by trial and error on a case by case basis. So what may work fine in one application may not work if one alters the bathymetry, geography, forcing, subgrid scale parameterizations, or numerical implementation of the OBC.

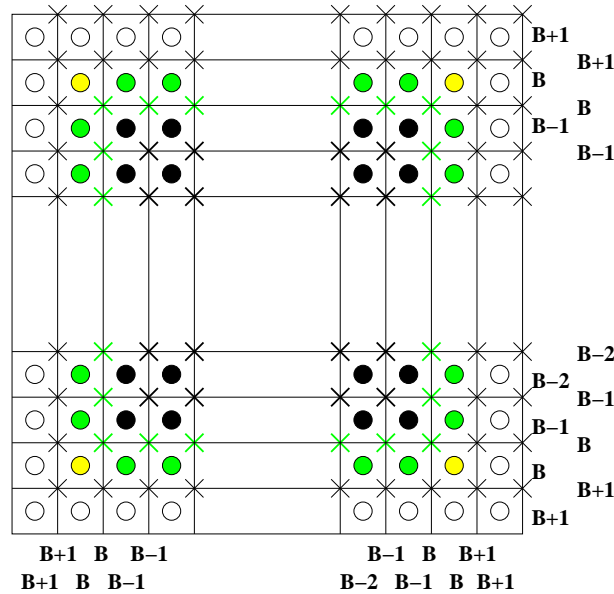


Figure 16.1: Open boundary conditions in the Arakawa B-grid. Circles mark tracer points, crosses velocity points. Open boundary conditions apply at green points.

16.2.1 Open boundaries in the Arakawa B-grid

Because MOM uses the ARAKAWA B-grid, tracers and sea level points are the outmost points, where the numerical scheme has to be closed by an open boundary condition. Velocity points are within the model domain. At points adjacent to the boundary the non-linear advective terms and diffusion terms are undefined. To close the numerical scheme for these terms, velocity points beyond the boundaries are defined

by appropriate extrapolation. The remaining terms in the momentum equations can be calculated using pre-existing fields. This formulation of the open boundary conforms to a stencil originally proposed by [Stevens \(1990\)](#), [Stevens \(1991\)](#). This approach effectively limits any error introduced by the OBC to the two non-linear terms, thus preventing any error associated with the OBC from rapidly propagating into the interior via the Coriolis terms.

16.2.1.1 Notation

Boundary points are marked with a capital B . The first points beyond the boundary outside the model domain is $B + 1$, the first internal point in the model domain is $B - 1$. See also [Figure 16.1](#)!

16.2.1.2 Boundary conditions for the sea level

The boundary conditions presented here refer to circulation models which use explicit solvers for the sea surface height where the variability of the sea level is governed by waves. Hence, the boundary condition for the sea level is based on the wave like properties of the solution and has the form of a radiation condition for outgoing waves. Boundary conditions for the sea level apply at points marked with green circles in [Figure figure:grid](#).

16.2.1.3 Boundary conditions for tracers

The tracer propagation is described by an advection-diffusion equation, which does not have a wave like solution itself. However, the underlying velocity field may be wave like. Hence, especially vertical tracer advection, which is not well defined near an open boundary, can be approximated by a radiation condition. The boundary condition used here combines a radiation condition, approximations for horizontal advection and relaxation towards prescribed data. Boundary conditions for the tracers apply at points marked with green circles in [Figure 16.1](#).

16.2.1.4 Boundary conditions for velocity

Boundary conditions for velocity vectors apply at points marked with a green cross in [Figure figure:grid](#). At these points horizontal and vertical velocity advection is set to zero. An exception is the metric term, which is well defined. Removing it implies horizontal inhomogeneity in the flow fields, which show up especially in nearly uniform flow.

To get a well defined friction operator, velocity beyond the boundary must be defined too. This is done with a no gradient condition. This condition applies at points marked with a this black cross in [Figure 16.1](#).

For some application it may help, to take vertical advection of tracers at boundary points into consideration. A no gradient condition for the tangential velocity enhances \pm -structures, which are typical for the Arakawa B-grid, and couples such structures into the tracer equations. Hence, the tangential velocity at points beyond the boundary should be set to the value of the second interior point, which prevents infiltration of \pm -structures from the velocity field into the tracer fields from the OBC.

16.2.2 Sommerfeld radiation condition

Most open boundary conditions are based on the Sommerfeld radiation condition ([Sommerfeld, 1949](#)). This kind of boundary condition was originally derived for the theory of electromagnetic waves to remove incoming waves from the far field solution for an oscillating dipole antenna. The Sommerfeld radiation condition takes the form:

$$\eta_{,t} = -c \eta_{,x} \quad \text{at } x = x_B. \quad (16.1)$$

In this equation, $\eta(x, t)$ is the space-time dependent free surface height, c is the wave phase speed determined via a method discussed in [Section 16.3](#), $x = x_B$ is the spatial position of the open boundary in

question. Finally, we use the shorthand notation

$$\begin{aligned}\eta_{,t} &= \frac{\partial \eta}{\partial t} \\ \eta_{,x} &= \frac{\partial \eta}{\partial x}\end{aligned}\tag{16.2}$$

to denote partial derivatives. The performance of OBC's based on the Sommerfeld condition (16.1) has been extensively assessed in the literature (see, for example [Chapman, 1985](#); [Roed and Cooper, 1987](#); [Tang and Grimshaw, 1996](#); [Palma and Matano, 1998, 2001](#)). Unfortunately, OBC's based on the Sommerfeld condition often exhibit inaccuracies. The key reason is that the model solution is a superposition of several waves which have different phase speeds and which are dispersive (in contrast to the linear electromagnetic waves that Sommerfeld was concerned). However, the Sommerfeld condition is based on just a single wave condition. The OBC behaviour in practice is thus very sensitive to how the phase speed in equation (16.1) is determined.

We can classify the boundary conditions as *passive*, in which case the OBC is determined solely from information within the computational domain, or *active*, so that data is prescribed from an external source. Since the behaviour of the model interior is rarely consistent with data prescribed at the boundary, the model may become prone to errors due to under-specification (not enough information describing external processes is provided) or over-specification (OBC information is incompatible with interior equations). [Marchesiello and Shchepetkin \(2001\)](#) provide a thorough discussion of the active versus passive boundaries, and over versus under-specification of data.

To alleviate problems with over-specification, an active boundary condition may be rendered partially passive by coupling to a radiation condition. This approach was used by [Blumberg and Kantha \(1985\)](#). Here, relaxation towards externally prescribed data is performed with an associated relaxation timescale, so that

$$\eta_{,t} = -c \eta_{,x} - (\eta - \eta_o)/\tau_f.\tag{16.3}$$

Here, η_o is the prescribed data for the surface height, and τ_f is a timescale. Even with this prescription for the OBC, the behaviour of the simulation can be sensitive to the choice of radiation condition and relaxation timescale used.

Likewise a fresh water flux may be added, but its influence will most probably disappear behind the relaxation term.

16.2.3 Clamped boundary conditions

A simple boundary condition is the clamped boundary condition, i.e., sea level or tracers are kept at a fixed value. For the sea level no physical justification is given for using this condition. It is motivated solely from the fact that it keeps the numerical scheme stable. For salinity and temperature it may be a reasonable approximation. The clamped condition requires the user to supply a single time and space independent value to be imposed on the boundary. This type of condition corresponds to a zero phase speed, $c = 0$, in the Sommerfeld radiation condition (16.1). It creates many reflections at the boundary, which can be undesirable.

In a modified form time and space dependent values for the sea level or tracers may be prescribed. Keeping in mind, that most ocean flow is geostrophically balanced, this defines the baroclinic and barotropic geostrophic transport through the boundary. This may be desired, but implies also the possibility of unwanted numerical effects, which may corrupt the numerical solution in the model domain.

16.2.4 No gradient boundary conditions

This condition imposes a smooth solution near the boundary. The no-gradient OBC assumes that there does not exist a gradient of a variable across the open boundary. It is sometimes referred to as a *Neumann* boundary condition. This condition corresponds to setting the phase speed to infinity in the Sommerfeld radiation condition (16.1). It is specified by setting the value at the open boundary equal to the value immediately adjacent to the open boundary in the model interior, so that

$$\eta(t+1, x_B) = \eta(t+1, x_{B-1}).\tag{16.4}$$

In this equation, $t + 1$ is the updated time step, x_B signifies the spatial position of the open boundary, with $\eta(t + 1, x_B)$ is the surface height at that boundary. As shown in Figure 16.1 x_{B-1} signifies the spatial position of the nearest point interior from the open boundary, with the sign determined by the relative position of the open boundary.

For velocity a no gradient boundary condition applies across the boundary, to get a defined viscosity operator

$$u(t + 1, x_{B+1}) = u(t + 1, x_B). \quad (16.5)$$

16.2.5 Interior cell no gradient boundary conditions

This condition is similar to the no-gradient boundary condition (16.4). However, instead of using the value at the nearest interior grid cell, the boundary is set to the value at the next nearest interior ($B - 2$) cell (i.e., two grid points away from the boundary)

$$\eta(t + 1, x_B) = \eta(t + 1, x_{B-2}). \quad (16.6)$$

For the tangential velocity a no gradient boundary condition across the boundary is used to get a defined viscosity operator

$$u(t + 1, x_{B+1}) = u(t + 1, x_{B-1}). \quad (16.7)$$

16.2.6 Enhanced friction and diffusion near the boundary

This method increases the dissipation from tracer diffusion and momentum friction in regions near the open boundary. This approach acts to dissipate spurious reflections at the boundary. It also may be useful to remove artificial currents near the boundary, which may grow to be large in some cases. However, this approach has the detrimental effect of slowing cross boundary transport.

16.3 Implementation of sea level radiation conditions

In this section, we discuss various radiation conditions that are used to specify the phase speed. We also discuss how to specify the tracers across the open boundary.

16.3.1 Sign convention for the phase speed

The phase speed is a vector quantity. Here it is always directed perpendicularly to the model boundary and notation can be simplified considerably by considering the projection of the phase speed onto the normal vector of the model boundary. The sign of the phase speed is positive for eastward or northward travelling waves and negative for westward or southward directed waves. However, the quantity of interest is the projection of the phase speed onto the boundary normal vector. In all radiation conditions given below, phase speed means this projection, which is positive for outgoing waves and negative for incoming waves.

16.3.2 Gravity wave radiation condition for the phase speed

This formulation computes a phase speed relevant for a flat bottom barotropic shallow water configuration, so that

$$c = \sqrt{g D_B}, \quad (16.8)$$

where g is the acceleration from gravity, and D_B is the depth at the boundary (Chapman, 1985)). Hence, in this approximation it is assumed, that processes near the boundary are always governed by outgoing waves. The OBC is implemented in an implicit form, so that

$$\eta(t + 1, x_B) = \frac{\eta(t, x_B) + \mu \eta(t + 1, x_{B-1})}{1 + \mu}, \quad (16.9)$$

where

$$\mu = c \frac{\Delta t}{\Delta x}, \quad (16.10)$$

where Δx is the horizontal grid spacing. That is, equation (16.9) is the implicit solution to equation (16.1) using c as the gravity wave speed.

Note: phase speed is always positive for gravity wave radiation, implying that waves are always outgoing. This means that outgoing relaxation, τ_{out} , is always used when relaxing to data (see Section 16.3.9.3).

16.3.3 Orlanski radiation condition for the phase speed

We aim to account for the most important part of the wave spectrum. There are various means for doing so, with the Orlanski radiation condition (Orlanski, 1976) one of the most common. Here, the phase speed of disturbances approaching the boundary is diagnosed at every time-step from the distribution of the interior values of the surface height near the boundary, so that

$$\vec{c} = -\eta_{,t}/\eta_{,x}. \quad (16.11)$$

The Orlanski radiation condition theoretically has a zero reflection coefficient. This property is desired to reduce spurious reflected waves at the open boundary. Unfortunately, in practice reflections occur due to inaccuracies in the phase speed computation.

The form employed by MOM is the implicit formulation based on (Chapman, 1985)

$$\eta(t+1, x_B) = \frac{(1-\mu)\eta(t-1, x_B) + 2\mu\eta(t, x_{B-1})}{1+\mu}. \quad (16.12)$$

Here, the dimensionless parameter μ is set according to

$$\mu = \begin{cases} 1 & \text{if } C \geq 1 \\ C & \text{if } 0 < C < 1 \\ 0 & \text{if } C \leq 0, \end{cases} \quad (16.13)$$

where

$$C = \frac{\eta(t-1, x_{B-1}) - \eta(t+1, x_{B-1})}{\eta(t+1, x_{B-1}) + \eta(t-1, x_{B-1}) - 2\eta(t, x_{B-2})}. \quad (16.14)$$

16.3.4 Camerlengo and O'Brien radiation condition

Camerlengo and O'Brien (1980) suggested a modified form of the Orlanski radiation condition, where only the extreme values of the phase speed, zero or h/t , so that;

$$\eta(t+1, x_B) = \begin{cases} \eta(t, x_{B-1}) & \text{if } C > 0 \\ \eta(t-1, x_B) & \text{if } C \leq 0, \end{cases} \quad (16.15)$$

with C given by equation (16.14).

16.3.5 Radiation condition after Miller & Thorpe

The Orlanski scheme is modified here so that time differences are evaluated using a forward scheme and space differences with an upwind scheme (see equation 15 in Miller and Thorpe, 1981)

$$\eta(t+1, x_B) = \eta(t, x_B) - \mu(\eta(t, x_B) - \eta(t, x_{B-1})). \quad (16.16)$$

In this case, the dimensionless coefficient

$$\mu = \mu_1 + \mu_2 + \mu_3, \quad (16.17)$$

with

$$\mu_1 = \frac{\eta(t+1, x_{B-1}) - \eta(t, x_{B-1})}{\eta(t, x_{B-2}) - \eta(t, x_{B-1})} \quad (16.18)$$

$$\mu_2 = \frac{\eta(t, x_B) - \eta(t-1, x_B)}{\eta(t-1, x_{B-1}) - \eta(t-1, x_B)} \quad (16.19)$$

$$\mu_3 = \frac{\eta(t, x_{B-1}) - \eta(t-1, x_{B-1})}{\eta(t-1, x_{B-2}) - \eta(t-1, x_{B-1})}. \quad (16.20)$$

The scheme is implemented in an explicit temporal form.

16.3.6 Raymond and Kuo radiation condition

This radiation condition was introduced by Raymond and Kuo (1984) and used in adaptive form by Marchesiello et al. (2001). This scheme calculates the phase velocity for multidimensional flows using a projection of each coordinate direction, i.e. not just the normal component. The scheme is implemented in implicit form. The Sommerfeld radiation condition takes the form:

$$\eta_{,t} = -c_x \eta_{,x} - c_y \eta_{,y} \quad (16.21)$$

where x and y are directions normal and tangential to the boundary respectively. The phase speeds c_x and c_y are projections given by:

$$c_x = -\eta_{,t} \frac{\eta_{,x}}{\eta_{,x}^2 + \eta_{,y}^2}, \quad (16.22)$$

$$c_y = -\eta_{,t} \frac{\eta_{,y}}{\eta_{,x}^2 + \eta_{,y}^2}. \quad (16.23)$$

This is discretised following Marchesiello et al (2001);

$$\begin{aligned} \eta(t+1, x_B, y_B) &= \frac{1}{1 + r_x} \\ &\begin{cases} \eta(t, x_B, y_B) + r_x \eta(t+1, x_{B-1}, y_B) - r_y (\eta(t, x_B, y_B) - \eta(t, x_B, y_{B-1})) & r_y > 0, \\ \eta(t, x_B, y_B) + r_x \eta(t+1, x_{B-1}, y_B) - r_y (\eta(t, x_B, y_{B+1}) - \eta(t, x_B, y_B)) & r_y < 0. \end{cases} \end{aligned} \quad (16.24)$$

where:

$$\begin{aligned} r_x &= -\frac{\Delta \eta_t \Delta \eta_x}{\Delta \eta_x^2 + \Delta \eta_y^2} \\ r_y &= -\frac{\Delta \eta_t \Delta \eta_y}{\Delta \eta_x^2 + \Delta \eta_y^2} \end{aligned} \quad (16.25)$$

$$\Delta \eta_t = \eta(t+1, x_{B-1}, y_B) - \eta(t, x_{B-1}, y_B) \quad (16.26)$$

$$\Delta \eta_x = \eta(t+1, x_{B-1}, y_B) - \eta(t+1, x_{B-2}, y_B) \quad (16.27)$$

$$\begin{aligned} \Delta \eta_y &= \begin{cases} \eta(t, x_{B-1}, y_B) - \eta(t, x_{B-1}, y_{B-1}) & \text{if } D > 0, \\ \eta(t, x_{B-1}, y_{B+1}) - \eta(t, x_{B-1}, y_B) & \text{if } D < 0 \end{cases} \\ D &= \Delta \eta_t (\eta(t, x_{B-1}, y_{B+1}) - \eta(t+1, x_{B-1}, y_{B-1})) \end{aligned} \quad (16.28)$$

The adaptive for relaxation takes on a form similar to Equation (16.3),

$$\eta_{,t} = -c_x \eta_{,x} - c_y \eta_{,y} - (\eta - \eta_o) / \tau_f. \quad (16.29)$$

where $\tau_f = \tau_{out}$ if $c_x > 0$ and $\tau_f = \tau_{in}$ with $c_x = c_y = 0$ if $c_x < 0$. The relaxation time scale $\tau_{out} \gg \tau_{in}$ such that during outward phase propagation a weak relaxation exists to avoid boundary values drifting excessively but also preventing problems of over-specification, while during inward phase propagation stronger relaxation is applied that avoids shock issues.

16.3.7 The IOW-radiation condition

As for the gravity wave radiation condition an implicate scheme is used,

$$\eta(t+1, x_B) = \frac{\eta(t, x_B) + \mu \eta(t+1, x_{B-1})}{1 + \mu}, \quad (16.30)$$

where

$$\mu = C \frac{\Delta t}{\Delta x}. \quad (16.31)$$

The spatial and time derivative of η are

$$\Delta\eta_{,x} = \frac{\eta(t+1, x_{B-1}) - \eta(t+1, x_{B-2})}{\Delta x}, \quad (16.32)$$

$$\Delta\eta_{,t} = \frac{\eta(t_s, x_{B-1}) - \eta(t+1, x_{B-1})}{\Delta t}. \quad (16.33)$$

If the predictor-corrector scheme is used, the phase speed is calculated twice. In this case $\eta(t_s)$ denotes η at the starting time step of this scheme. In the predictor step Δt is the reduced time step $\Delta t = \gamma dt$. For the leapfrog scheme $t_s = t - 1$ and $\Delta t = 2dt$.

To ensure a well defined phase speed for small horizontal gradients this case is treated separately,

$$C(t+1) = \begin{cases} C^*(t+1) & \text{if } \Delta\eta^x > a, \\ 0.99\tilde{C}(t) & \text{if } \Delta\eta^x \leq a, \end{cases} \quad (16.34)$$

where $\Delta\eta^x = |\eta(t+1, x_{B-1}) - \eta(t+1, x_{B-2})|$ and a is a small length, typically 10^{-8} m. \tilde{C} is the time smoothed phase speed from the previous time step. The scheme allows to control the minimum and maximum value of the phase speed. For incoming waves, negative phase speed, positive values (or zero) are assumed,

$$C^*(t+1) = \begin{cases} C_{inc} & \text{if } C^+ < 0, \\ C_{min} & \text{if } C^+ < C_{min}, \\ C^+ & \text{if } C_{max} > C^+ > C_{min}, \\ C_{max} & \text{if } C^+ > C_{max}. \end{cases} \quad (16.35)$$

$C^+(t+1)$ is calculated from the derivatives of η ,

$$C^+(t+1) = \frac{\Delta\eta_{,t}}{\Delta\eta_{,x}}. \quad (16.36)$$

C_{max} , C_{min} and C_{inc} are defined in terms of the gravity wave speed,

$$C_{max} = c_{max} \sqrt{g D_B} \quad (16.37)$$

$$C_{min} = c_{min} \sqrt{g D_B} \quad (16.38)$$

$$C_{inc} = c_{inc} \sqrt{g D_B}. \quad (16.39)$$

The factors c_{max} , c_{min} and c_{inc} can be modified via the namelist.

Relaxation is done as described in Section 16.3.9.

16.3.8 Phase speed smoothing

The diagnosed phase speed may be very noisy with altering sign every time step. A time smoother

$$\tilde{C}(t+1) = F\tilde{C}(t) + (1-F)C(t+1), \quad (16.40)$$

helps to reduce numerical noise. The default value is $F = 0.7$. The application of the smoother requires to save \tilde{C} in a restart file, to ensure reproducibility across model restarts.

16.3.9 Relaxation to data

16.3.9.1 Relaxation towards prescribed profiles

The radiation conditions may be coupled to prescribed data as described by Equation (16.3). This approach is implemented implicitly as

$$\eta(t+1, x_B) = \frac{\tilde{\eta}(t+1, x_B) + \eta_0 \frac{\Delta t}{\tau_f}}{1 + \frac{\Delta t}{\tau_f}}, \quad (16.41)$$

where $\tilde{\eta}(t+1, x_B)$ is the solution on the boundary derived from the radiation conditions. τ_f is the time scale for the relaxation process.

The relaxation may occur a given number of cells into the interior, in which case $\tilde{\eta}(t + \Delta t, x_B)$ is the interior solution to the continuity equation.

16.3.9.2 Relaxation of the sea level average - conservation of geostrophic currents

If the model area is a semi-enclosed sea, which is connected to the ocean by a narrow channel, the open boundary may be placed within this channel. By prescribing the sea level profile across the channel, one defines also the geostrophic volume transport through this channel. In this case the incertitude of the OBC may dominate the volume budget of the model. To overcome this serious shortcoming, an alternative relaxation scheme may be used, which prescribes only the average sea level at the boundary. Doing so, the geostrophic transport defined by the cross channel sea level gradient is not affected by the relaxation. Only the ageostrophic flow may react to the prescribed sea level variation, geostrophic adjustment happens through the internal model dynamics. The scheme works explicitly,

$$\eta(t+1, x_B) = \tilde{\eta}(t+1, x_B) + \frac{\Delta t}{\tau_f} (\eta_o - \langle \tilde{\eta}(t+1, x_B) \rangle) \quad (16.42)$$

$\langle \tilde{\eta}(t+1, x_B) \rangle$ is sea level averaged over the boundary.

16.3.9.3 Variable relaxation for incoming and outgoing waves

Likewise, the value of τ_f may be different for prevailing incoming or outgoing waves,

$$\tau_f^{-1} = r \tau_{out}^{-1} + (1-r) \tau_{in}^{-1}, \quad (16.43)$$

$$r(t) = \frac{\tilde{C}(t)}{C_{max}}. \quad (16.44)$$

16.4 OBC for tracers

16.4.1 Reduced tracer equations at open boundaries

The tracer equations in MOM are strongly linked with the sea level equation and tracers and sea level are treated consistently. Because the radiation condition for the sea level is a rough approximation, a similar consistency cannot be achieved for boundary points. Hence, the tracer equations at the boundaries are simplified.

To avoid double coding, the normal code should be used at boundaries as far as possible. Tracers are updated as

$$T(t+1)\rho(t+1)h(t+1) = T(t)\rho(t)h(t) + \Delta t \delta(h(t)\rho(t)T(t)) \quad (16.45)$$

The time tendency of the vertically integrated tracer in the grid cell $\delta(\rho T)$ is the combined time tendency of tracer concentration, density and cell thickness. It consists of an advective and diffusive contribution and eventually of source terms from radiation and special convective schemes (kpp). We rewrite these terms in such a manner, that for a constant tracer the sea level equation is retained. Especially for a uniform tracer $T = 1$, $\delta(h\rho T)$ must be the time tendency of the cell thickness, $\delta(h\rho)$, i.e. $\delta(h\rho 1) \rightarrow \delta(h\rho) = \frac{\Delta h\rho}{\Delta t}$.

$$\begin{aligned} \delta(h\rho T)_k &= -(\nabla_s \cdot h\rho(\mathbf{u}T + \mathbf{F}))_k - (\rho(wT + F^{(s)}))_{k-1} + (\rho(wT + F^{(s)}))_k + (h\rho S^T)_k \\ \delta(h\rho)_k &= -(\nabla_s \cdot h\rho\mathbf{u})_k - (\rho w)_{k-1} + (\rho w)_k + (h\rho S^M)_k \end{aligned} \quad (16.46)$$

Advection is rewritten to separate the different contributions to the time tendency,

$$\begin{aligned} Adv(T)_k &= -(\nabla_s \cdot h\rho\mathbf{u}T)_k - (\rho wT)_{k-1} + (\rho wT)_k \\ &= -(T\nabla_s \cdot h\rho\mathbf{u})_k - (\rho wT)_{k-1} + (\rho wT)_k - (h\rho\mathbf{u} \cdot \nabla_s T)_k. \end{aligned} \quad (16.47)$$

Applying approximations for open boundaries, the first three terms have to be kept consistent with the equation for the level thickness. The level thickness is calculated from approximations only and it is appropriate to express the convergence of the flow in terms of the level thickness time tendency. With of

$$-(\nabla_s \cdot h\mathbf{p}\mathbf{u})_k = \delta(h\rho)_k + (\rho w)_{k-1} - (\rho w)_k - (h\rho S^M)_k \quad (16.48)$$

this renders to

$$\begin{aligned} \delta(h\rho T)_k &= \delta(h\rho)_k T_k + (h\rho(S^T - S^M T))_k + SGS \\ &\quad - (h\rho\mathbf{u} \cdot \nabla_s T)_k - (\rho w T)_{k-1} + (\rho w T)_k + ((\rho w)_{k-1} - (\rho w)_k) T_k. \end{aligned} \quad (16.49)$$

i.e., which ensures consistency between tracers and layer thickness for a uniform tracer. Also the approximation of zero vertical advection and horizontal advection with an upwind scheme is consistent with the thickness equation for cells of constant thickness. Diffusion and source terms apply unchanged and will not be specified here.

For surface cells the vertical advection at the surface is expressed in terms of sea level variation, fresh water flux and turbulent tracer flux,

$$\begin{aligned} \delta(h\rho T)_1 &= (\delta(h\rho)_1 T_1 + \rho_w q_w (T_w - T_1) + (h\rho(S^T - S^M T))_1 - Q_T^{turb} + SGS \\ &\quad - (h\rho\mathbf{u} \cdot \nabla_s T)_1 + (\rho w T)_1 - (\rho w)_1 T_1. \end{aligned} \quad (16.50)$$

With this approximation the tracer concentration remains unchanged, if only the sea surface height is undulating. If currents are zero, but fresh water flux and diffusion are present, a horizontally uniform tracer distribution will not be disturbed near an open boundary. The consistency between sea level equation and tracer equation is not broken, if approximations for the horizontal advection term are made, or if radiation terms are added. Those terms vanish for horizontally uniform tracers especially for $T = 1$ and the sea level time tendency is trivially retained.

Because vertical velocity is not well defined it is left out of consideration. It is replaced by a radiation term, which accounts for the propagation of wave like undulations of internal interfaces from baroclinic waves.

$$\begin{aligned} \delta(h\rho T)_k &= \delta(h\rho)_k T_k + (h\rho(S^T - S^M T))_k + SGS \\ &\quad - (h\rho\mathbf{u} \cdot \nabla_s T)_k + c\rho h \frac{\partial T_k}{\partial x} + \rho h \frac{T^{ref} - T_k}{\tau_f} \end{aligned} \quad (16.51)$$

$$\begin{aligned} \delta(h\rho T)_1 &\approx (\delta(h\rho)_1 T_1 + \rho_w q_w (T_w - T_1) + (h\rho(S^T - S^M T))_1 - Q_T^{turb} + SGS \\ &\quad - (h\rho\mathbf{u} \cdot \nabla_s T)_1 + c\rho h \frac{\partial T_1}{\partial x} + \rho h \frac{T^{ref} - T_1}{\tau_f}. \end{aligned} \quad (16.52)$$

Implementation of the radiation condition and of the relaxation towards prescribed data is documented below. Diagnostics of the phase speed is based on the tracer concentration, because the total tracer content may undulate rapidly from the barotropic mode in the cell thickness variability.

16.4.2 Upstream advection of tracers near the boundary

For advection across the open boundary, an upstream scheme with

$$T(x_{B+1}) = T(x_B) \quad (16.53)$$

means that the incoming tracer has the same concentration as the tracer at the boundary point. This approximation may give poor results for long model runs and can cause model drifts.

After running a *passive* boundary over a long period, the tracer near the boundary will be determined completely by processes in the model domain. As an example consider a marginal sea with a strong fresh water surplus. There will be an estuarine circulation with a more or less permanent outflow of brackish water in a surface layer and inflow near the bottom. However, the salinity of the inflowing water will be

reduced as well after some time by vertical mixing processes. The model results will suffer from underestimated stratification. To overcome this problem, information on the tracer concentration in the adjacent sea must be provided for the model. The simple approximation

$$T(x_{B+1}) = T(x_0) \quad (16.54)$$

where T_0 may stem from a database, improves the performance of the diffusion and the advection operators, which in turn may invoke wave like processes spreading from the boundary into the model. Using an upstream formulation for the tracer gradient in the advective term, this can switch on an inflow through the open boundary. However, waves of a small amplitude but with a high phase speed may disturb this scheme. Thus, the tracer source term \mathcal{S}_T can be used for a controlled restoring to prescribed boundary values. The upstream advection condition is discretized as:

$$T(t+1, x_B) = T(t, x_B) + \frac{\Delta t}{\Delta x} [(u_n - |u_n|)(T(t, x_{B-1}) - T(t, x_B)) + (u_n + |u_n|)(T(t, x_B) - T(t, x_0))] \quad (16.55)$$

where u_n is the velocity normal to the boundary times density and the cell height, $T(x_B)$ is the tracer on the boundary, $T(x_{B-1})$ is the tracer one cell into the interior and $T(x_0)$ is a tracer value that must be supplied externally.

16.4.3 Relaxation towards external data

If external data are prescribed boundary values may be relaxed towards there data as

$$T(t+1, x_B) = (T_0 - T(t, x_B)) \frac{\Delta t}{\tau_f} + T(t, x_B) \quad (16.56)$$

The relaxation time τ_f depends on the flow direction near the boundary. If the sum of advection velocity and phase speed at the boundary is directed inwards, one has $\tau_f = \tau_f^{in}$ and $\tau_f = \tau_f^{out}$ otherwise. τ_f^{in} and τ_f^{out} can be specified in the namelist for each tracer and boundary separately.

16.4.4 Flow relaxation scheme of Martinsen and Engedahl

The flow relaxation scheme of Martinsen and Engedahl (1987) has been included to relax boundary data to interior data. This is accomplished over a region NN cells wide (typically $NN=10$) where the tracer variables are updated according to:

$$T = \alpha_i T_B + (1 - \alpha_i) T_{B \pm i} \quad (16.57)$$

where T_B is the boundary specified value, $T_{B \pm i}$ are the interior variable values and α_i is a relaxation parameter given by:

$$\alpha_i T_B = 1 - \tanh \frac{i-1}{2} \quad i = 1, 2, \dots, NN \quad (16.58)$$

Note that the flow relaxation scheme is used in conjunction with another boundary condition and T_B may be obtained from the FILEIN or NOGRAD condition; whatever is specified on the boundary is relaxed to the model integrated values over NN cells. The flow relaxation scheme is only implemented if UPSTRM is included in the tracer obc. If T_B is equal to zero (clamped boundary condition) then this flow relaxation scheme acts as a sponge type condition. An example of the flow relaxation scheme implementation in the namelist is given below:

```
obc_flow_relax(:,1) = 10, 1, 1
obc_flow_relax(:,2) = 10, 1, 1
```

16.4.5 Radiation conditions

If the velocity field near the boundary is wave like, vertical advection may result in a wave like tracer motion. Hence, a radiation condition may improve the numerical scheme at the boundary.

The radiation condition for tracers is applied implicitly,

$$T(t+1, x_B) = \frac{T(t, x_B) + \mu T(t+1, x_{B-1})}{1 + \mu}, \quad (16.59)$$

where

$$\mu = C \frac{\Delta t}{\Delta x}. \quad (16.60)$$

The phase speed C is set to zero for incoming waves and is limited by C_{max} ,

$$\mu = \begin{cases} 0 & \text{if } C^* < 0, \\ C^* & \text{if } 0 < C^* < C_{max}, \\ C_{max} & \text{if } C^* > C_{max}. \end{cases} \quad (16.61)$$

C^* is calculated either bei the Orlanski scheme,

$$C^* = \frac{\Delta x}{\Delta t} \frac{T(t-1, x_{B-1}) - T(t+1, x_{B-1})}{T(t+1, x_{B-1}) + T(t-1, x_{B-1}) - 2T(t, x_{B-2})}, \quad (16.62)$$

or by used in MOM-31 (IOW).

$$C^* = \frac{\Delta x}{\Delta t} \frac{T(t-1, x_{B-1}) - T(t+1, x_{B-1})}{T(t, x_{B-1}) - T(t, x_{B-2})}. \quad (16.63)$$

For the time staggered scheme the index $t-1$ points to the same filed as t . The maximum phase speed, C_{max} , is given by the CFL-criterion,

$$C_{max} = \frac{\Delta x}{\Delta t}. \quad (16.64)$$

16.4.6 Vertical mixing and viscosity co-efficients

Vertical mixing at boundary points my be enabled or diabled. However, the mixing co-efficients at these points are used to define viscosity at adjacent velocity points. The namelist parameter `obc_mix` specifies, how the mixing coefficient at boundary points is defined. Options are `NOTHIN`, `NOGRAD`, `INGRAD` and `CLAMPD`, `obc_mix=NOGRAD` is the default.

16.4.7 Enhanced horizontal mixing and viscosity co-efficients

To damp unwanted numerically generated flow patterns near open boundaries viscosity and diffusivity may be enhanced within a stripe near the boundary.

16.5 The namelist `obc_nml`

MOM requires that two components of velocity be prescribed on each open boundary (normal and tangential velocities to the boundary) for both the 3D and 2D modes. Surface elevation and the values of any tracers present are also required. Here are the flags that specify the various fields:

- Normal velocity: `obc_nor`
- Tangential velocity: `obc_tan`
- Sea level elevation: `obc_eta`
- Tracers: `obc_tra(:)`.

The OBC specification is determined via text strings as listed in Table 16.1, where the variables these conditions may be applied to are listed as u_n = normal velocity, u_t = tangential velocity, η = surface elevation, and T = tracers. The condition specified for velocities is used for both the 2D and 3D modes. Note that the text identifier strings have been truncated to the same length while attempting to describe the OBC condition to accommodate neat alignment in the namelist.

CONDITION NAME	TEXT IDENTIFIER	APPLICABLE VARIABLES
Relaxation to data	FILEIN	η, T
Relaxation of mean	MEANIN	η
Clamped	CLAMPD	u_n, u_t, η
No-gradient	NOGRAD	u_n, u_t, η, T
Interior-gradient	INGRAD	u_n, u_t
Linear extrapolation	LINEAR	u_n, u_t
Gravity wave radiation	GRAVITY	η
Orlanski	ORLANS	η, T
Camerlengo and O'Brien	CAMOB	η
Miller and Thorpe	MILLER	η
Raymond and Kuo	RAYMND	η
Schmidt	MARTIN	η, T
Upstream advection	UPSTRM	T

Table 16.1: Namelist settings for the OBC specification.

An example of open boundary specification for a domain containing three open boundaries and two tracers is given below:

```
nobc = 3
direction = 'west', 'south', 'north'
is = 2, 2, 2
ie = 2, 10, 10
js = 2, 2, 20
je = 20, 2, 20
obc_nor = 'NOGRAD', 'NOGRAD', 'NOGRAD'
obc_tan = 'INGRAD', 'INGRAD', 'INGRAD'
obc_eta = 'GRAVITY', 'FILEIN', 'MEANIN|ORLANS'
obc_tra(:,1) = 'UPSTRM|FILEIN', 'UPSTRM|FILEIN', 'UPSTRM|FILEIN'
obc_tra(:,1) = 'NOGRAD', 'NOGRAD', 'UPSTRM|NOGRAD'
```

Note that by 'or-ing' two conditions together then these two conditions will be invoked sequentially (order is not important). Hence the condition `UPSTRM|FILEIN` will invoke an upstream advection condition, using data from file when flow is into the domain. The condition `UPSTRM|NOGRAD` will invoke an upstream advection condition, using the value one cell into the domain when flow is into the domain. A wave-like contribution to the OBC can also be added for tracers, e.g. invoked by `UPSTRM|FILEIN|ORLANS`. This wave-like contribution is added implicitly.

If partially passive conditions are to be used for elevation, then the `FILEIN` or `MEANIN` condition is 'or-ed' with the desired radiation condition, e.g. using the Orlanski partially passive condition with data prescribed from file use `FILEIN|ORLANS`. Note that in-going and out-coming relaxation timescales are also required to be prescribed for these partially passive conditions.

Some compilers do not like the colon (:) syntax in field specifications. In this case each element must be specified separately.

16.6 Topography generation - Preparation of boundary data

16.6.1 Topography generation with open boundaries

Open boundary conditions require modifications of the topography near the boundary. Gradients of the depth normal to the boundary should be zero to avoid large vertical velocity in the boundary area. Because vertical velocity usually is set to zero at boundary points, this approximation is less serious, if vertical velocity is zero anyway.

The grid and topography generator *ocean_grid_generator* closes all model boundaries, if the model is neither cyclic or global. This has to be modified for open boundaries. Hence, open boundaries need to be specified in this early stage of model preparation. This ensures, that initial fields and boundary values, which may use the information in the grid specification file *grid_spec.nc*, are fully consistent with the topography used during model run time. For this purpose, the ocean grid generator *ocean_grid_generator* is able to read those parts of the namelist *obc_nml*, which define the open boundaries geographically. Here is an example:

```
\&obc_nml
  nobc                = 3
  direction            = 'north', 'south', 'west',
  is                   =      2,      2,      2,
  ie                   =     39,     63,      2,
  js                   =     74,      2,      2,
  je                   =     74,      2,     74,
  name                 = 'northern', 'southern', 'western'
```

It is not recommended, to define the boundary conditions at outmost model points.

16.6.2 Preparation of input data files

Having the *grid_spec.nc* file ready, one may proceed with preparing obc input data files. The grid of input data files should match exactly the size of the open boundary. However, it is also possible, that the grid of the input files may cover a larger area. In this case, those start and end index of the model grid, which matches the first and the last index in the input file, must be specified in the namelist *obc_nml* (but not for *ocean_grid_generator*). The default is, that the input data files match the size of the boundary exactly. In the example below, the input files for sea level and tracers have the same size as the model itself, 75×65 . They may be either preprocessed as described below, or used directly with help of namelist specifications:

```
\&obc_nml
  nobc                = 3
  direction            = 'north', 'south', 'west',
  is                   =      2,      2,      2,
  ie                   =     39,     63,      2,
  js                   =     74,      2,      2,
  je                   =     74,      2,     74,
  name                 = 'northern', 'southern', 'western'
  iers                 =      1,      1,      2,
  iere                 =     65,     65,      2,
  jers                 =      2,      2,      1,
  jere                 =      2,      2,     75,
  itr                  =      1,      1,      2,
  itre                 =     65,     65,      2,
  jtr                  =     74,      2,      1,
  jtre                 =     74,      2,     75,
```

It is supposed, that some data suitable for OBC are ready in netcdf-format. There are many tools to process such files, here *ferret* is used. Alternatives may be *grads* and possibly *matlab* in combination with

the *netcdf* toolbox. *Ferret* is available from <http://ferret.wrc.noaa.gov/Ferret>. It is recommended to use *Ferret* 6 or a later version, because previous versions do not permit full access to all netcdf attributes. However, as long as the file are not too large, some fine tuning in the file structure could also be done with a combination of the programs *ncdump*, a good editor, which can handle large files and *ncgen*. *ncdump* and *ncgen* come with the netcdf library. Also the *nco-tools* are of great help.

Examples for *ferret* scripts are given below. For details of the syntax visit <http://ferret.wrc.noaa.gov/Ferret>.

The following directory structure is assumed:

```
preprocessing/grid_spec.nc
preprocessing/OBCDATA
preprocessing/OBCDATA/1999/your_input.dta.nc
preprocessing/OBCDATA/2000/your_input.dta.nc
...
```

The working directory is for example

```
preprocessing/OBC/2000/
```

Then the following *ferret* commands should be sufficient to generate the input file for the sea level at a northern boundary at $j=74$:

```
SET MEMORY/SIZE=30
use ".../grid_spec.nc"
use ".../OBCDATA/2000/your_input.dta.nc"
! the input file has units "cm", mom4 needs "m"
let/units=m/title=eta_t eta_t = eta[d=3,gx=wet[d=1,j=74]]/100
can axis/modulo `eta_t,return=xaxis`
! add a calendar
SET AXIS/CALENDAR=JULIAN `eta_t,return=taxis`
save/clobber/file=obc_trop_north.dta.nc/2:39 eta_t
```

It may happen, that the model landmask differs from the land mask in the input data. In this case one may have land information from the input file at ocean points in the OBC input file, which would let the model crash. In this case *ferret* functions could be used to fill these values with ocean data. Suitable tools are the *@fmr* transformation or the new external function *fill_xy* which is in the latest *Ferret* 6 release.

For depth dependent data as temperature and salinity more care is needed to avoid gaps in the input data near the bottom. Most likely, topography representation in the model differs from topography representation in the input data. So two things are needed, to organise the input data

- a mask, to define ocean points in your model
- an input data set, which covers all model ocean points with ocean data.

The mask can be derived from the *grid_spec.nc* file:

```
SET MEMORY/SIZE=30
use ".../grid_spec.nc"
let mask_t=if k[gz=zb] le NUM_LEVELS then 1 else (-1)/0
save/clobber/file=tempfile.nc mask_t
```

For velocity data *NUM_LEVELS_C* can be used in the same manner.

Extrapolation into the bottom should be mostly sufficient, to extend the input data, so that all model ocean points are covered with input ocean data later:

```
use ".../OBCDATA/2000/your_input.dta.nc"
let temp_n = temp[k=@fmr:5] ! 5 should be sufficient
let salt_n = salinity[k=@fmr:5]
save/append/file=tempfile.nc temp_n, salt_n
```

The names of variables in the input file may be different. Saving into a temporary file is not needed in any case, but it helps to avoid problems from ambiguous indeces in variables with different co-ordinate definitions.

Now use *tempfile.nc* as new input file:

```
can data/all
can/var/all
!

---


SET MEMORY/SIZE=55
use tempfile.nc
let/unit=Celsius/title=temperature temp = temp_n[g=mask_t]*mask_t[j=74]
let/unit=PSU/title=salinity salt = salt_n[g=mask_t]*mask_t[j=74]
save/clobber/file=obc_clin_north.dta.nc/2:39 temp, salt
```

Multiplying with the mask ensures, that only model ocean points contain tracer information. The grid information is implicitly in *mask_t*. Do not specify the range of the grid index for writing *tempfile.nc*. This may disturb the horizontal interpolation.

16.6.3 Consistency of input data and model configuration

16.6.3.1 The sea level in external data and the model zero level

The models zero motion sea level is the average of the initial sea level. This value needs to be consistent with boundary sea level data. If boundary and initial data come from a larger model, this should be the trivially the case. Otherwise some adjustment is needed, because even small artificial gradients between boundaries and the model interior may drive large currents, which would rapidly corrupt the initial stratification. This requires an initial run, with advection of tracers switched off. This can be done with the options *zero_tracer_advect_horz* and *zero_tracer_advect_vert* enabled in *ocean_tracer_advect.nml*. The resulting model sea level should be a reasonable choice for model initialisation. Eventually the procedure could be repeated.

16.6.3.2 The sea level and the problem of air pressure

Air pressure gradients are part of geostrophic balance of current systems. If the air pressure gradients vary only slowly, a corresponding negative sea level gradient develops, which may compensate its influence, so that the currents calculated with and without air pressure are approximately the same. This is the reason, why air pressure is often omitted in circulation models.

The sea level however may differ considerably in both cases. This has to be taken into account, if sea level data are prescribed at open boundaries. MOM-4 permits the input of sea level air pressure, which is added to the sea level elevation. Hence, after geostrophic adjustment air pressure gradients and sea level gradients partially balance each other.

Subgrid scale parameterizations for vertical processes

The purpose of this part of the manual is to describe certain of the subgrid scale (SGS) parameterizations of physical processes used in MOM, with focus here on vertical and/or dianeutral processes.

SURFACE AND PENETRATIVE SHORTWAVE HEATING

Contents

17.1	General considerations and model implementation	249
17.2	The Paulson and Simpson (1977) irradiance function	250
17.3	Shortwave penetration based on chlorophyll-a	251
17.3.1	Solar penetration in the ocean	251
17.3.2	Morel and Antoine (1994) shortwave penetration model	251
17.3.3	SeaWiFS based chlorophyll-a climatology	252
17.4	Diagnosing shortwave heating in MOM	252

The purpose of this chapter is to discuss the numerical implementation of shortwave heating. Sensitivity of the ocean solution to the penetration of shortwave radiation into the ocean column can be quite large. This chapter, especially Section 17.3, benefitted from contributions by Colm Sweeney.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_sbc.F90
ocean_param/sources/ocean_shortwave.F90
ocean_param/sources/ocean_shortwave_gfdl.F90
ocean_param/sources/ocean_shortwave_csiro.F90
ocean_param/sources/ocean_shortwave_jerlov.F90
```

Notably, the documentation here is somewhat out of date, with the code modules the best source for how the various optical models have been implemented. In particular, the [Morel and Antoine \(1994\)](#) optical scheme detailed in Section 17.3.2 has largely been supplanted at GFDL by the [Manizza et al. \(2005\)](#) scheme available in MOM. Nonetheless, this chapter provides a useful overview of general methods for how shortwave penetration is attenuated in MOM.

17.1 General considerations and model implementation

Solar penetration brings solar shortwave heating downward in the ocean column, thus providing a heating at depth. The parameterization of the oceanic absorption of downward solar radiation is generally written as

$$I(x, y, z) = I_0^-(x, y) \mathcal{F}(z), \quad (17.1)$$

where I_{0-} , in units of W m^{-2} , is the total shortwave downwelling radiative heating per unit area incident at the earth surface, and $\mathcal{F}(z)$ is a dimensionless attenuation function. Note that the total downwelling radiation I_{0-} is to be distinguished from the total shortwave heating I_0 , where $I_{0-} = (1 - \alpha) I_0$, with $\alpha \approx 0.06$ the sea surface albedo.

Shortwave heating affects the heat budget locally according to

$$\frac{\partial(\rho \Theta)}{\partial t} = -(\rho_o/c_p) \partial_z (c_p F^z - I). \quad (17.2)$$

In this equation, F^z accounts for vertical processes such as advection and diffusion, c_p is the heat capacity of seawater, ρ is the *in-situ* density which for a Boussinesq fluid is set to the Boussinesq reference density ρ_o . Finally, Θ is the conservative temperature of [McDougall \(2003\)](#), which is commonly approximated by potential temperature θ .

Shortwave heating leads to the following net heat flux over a column of ocean fluid

$$(\rho_o/c_p) \int_{-H}^{\eta} dz \partial_z I = (\rho_o/c_p) [I(\eta) - I(-H)]. \quad (17.3)$$

where $I(\eta)$ is often approximated as $I(0)$. We assume there is no shortwave heating of the solid rock underneath the ocean fluid, so $I(z = -H) = 0$ is appropriate, with this boundary condition set via masks in MOM. Although the expression (17.5) suggests the upper boundary condition $I(0) = I_{0-}$, we must be careful to not double-count the shortwave source in since it is typically carried as part of the surface temperature flux array `stf` and thus handled by the module `ocean_core/ocean_sbc.F90`. We now present the two approaches available in MOM.

The vertical convergence of penetrative shortwave radiation, $(\rho_o/c_p) \partial_z I$, is incorporated into MOM's potential temperature equation. Additionally, it is typical to include the total downwelling shortwave heating I_{0-} within the surface flux array `stf`, where other forms of heating such as those from latent and long-wave affects are also incorporated. Hence, for proper accounting of the shortwave heating, the upper boundary condition for the irradiance function must be specified as

$$I(\eta) = \begin{cases} 0 & \text{if } I_{0-} \text{ is already included in } \text{stf} \\ I_{0-}(x, y) & \text{if } I_{0-} \text{ is NOT already included in } \text{stf}. \end{cases} \quad (17.4)$$

The typical practice at GFDL is to set $I(\eta) = 0$ since I_{0-} is already included in `stf`. Care should be exercised by those using the opposite convention.

17.2 The Paulson and Simpson (1977) irradiance function

[Jerlov \(1968\)](#) classified water into five types according to its optical properties, which determines the extent that solar radiation penetrates into a vertical fluid column. For example, clear water allows for deeper penetration, whereas murky water, as occurs in the presence of active biology, more rapidly attenuates the radiation. Studies by [Paulson and Simpson \(1977\)](#) then suggest a form for the attenuation function $\mathcal{F}(z)$ based on the optical properties of water.

The parameterization of solar shortwave absorption of downwelling solar radiation used by [Rosati and Miyakoda \(1988\)](#) is given by the [Paulson and Simpson \(1977\)](#) form

$$I(x, y, z) = I_{0-}(x, y) [R e^{z/\zeta_1} + (1 - R) e^{z/\zeta_2}], \quad (17.5)$$

where I_{0-} is the total shortwave downwelling radiative heating per unit area incident at the earth surface, ζ_1 and ζ_2 are attenuation lengths and R is an empirical constant dependent on the optical properties of the water. In the [Rosati and Miyakoda \(1988\)](#) study, they chose $R = 0.58$, $\zeta_1 = 0.35\text{m}$, and $\zeta_2 = 23\text{m}$, corresponding to Jerlov Type I water (clear water). This is the form of shortwave penetration originally implemented in MOM. However, MOM4.0 and later releases implement the scheme discussed in [Section 17.3](#).

17.3 Shortwave penetration based on chlorophyll-a

This Section was contributed by Colm Sweeney (cos@gfdl.noaa.gov).

Recent investigations with solar penetration have indicated a strong sensitivity of ocean simulations to how light penetrates into the ocean and warms the upper few tens of meters. This section describes recent work at GFDL whereby the use of a spatially and temporally dependent penetration depth is specified according to climatological chlorophyll data. Although this data is taken from a climatology, its use is believed to be preferable, even for climate change simulations, to the use of a space-time independent optical type. Research into bio-optical influences on ocean climate remains an active area, and the methods presented here remain under investigation by various groups.

17.3.1 Solar penetration in the ocean

Observations and models describing the distribution of solar radiation with depth in seawater have demonstrated that nearly all (99.9%) of the downwelling infrared radiation (wavelengths of 750nm-2500nm) is absorbed in the upper 2 meters of the ocean. While variability in the penetration depth of long wave radiation has large implications for skin temperature and surface layer heat fluxes, this IR radiation has little effect below the surface over the depths (roughly 10 m) used in present day ocean models (Morel and Antoine (1994), Ohlmann and Siegel (2000)).

In contrast, attenuation of shortwave solar radiation (wavelengths < 750nm) is both spatially and temporally variable at scales important to ocean climate models. In waters with high particulate and dissolved organic matter concentrations, the 1% light penetration depth can be less than 10 meters while in the clear, biologically unproductive waters of the subtropical gyres, solar radiation can directly contribute to the heat content at depths greater than 100 meters.

Previously, ocean climate models have parameterized the penetration of light into the ocean by identifying six water types distinguished by the "clarity" of the water (Paulson and Simpson (1977)). This method is described in Section 17.2. More recently, the correlation between chlorophyll pigment concentrations and short wave penetration depths has enabled us to parameterize shortwave penetration of solar radiation with satellite observations of ocean color (Morel (1988), Morel and Antoine (1994), Ohlmann and Siegel (2000)). By assimilating satellite observations of ocean color (e.g., SeaWiFS, OCTS, MODIS), it is possible to more accurately simulate the spatial and temporal variability of penetrating radiation.

17.3.2 Morel and Antoine (1994) shortwave penetration model

Present modeling work at GFDL (Spring 2003) makes use of a parameterization developed by Morel and Antoine (1994). This scheme was explicitly developed for large-scale ocean models with coarse depth resolution (i.e., grids with $\Delta z > 5\text{m}$). Morel and Antoine (1994) expand the exponential function used by Denman (1973) into three exponentials to describe solar penetration into the water column.

The first exponential is for wavelengths > 750nm (i.e., I_{IR}) and assumes a single attenuation of 0.267meter with a solar zenith angle $\theta = 0$. At present we assume that the solar zenith angle stays constant throughout the daily integration of the model. Thus, the resulting equation for the infrared portion of the downwelling radiation is:

$$I_{IR}(x, y, z) = I_{IR-}(x, y) e^{-z/(0.267 \cos \theta)} \quad (17.6)$$

where again $\theta = 0$. This relationship assumes z is the depth in meters and that $I_{IR-}(x, y)$ is a fraction, of the total downwelling radiation, I_{0-} such that

$$I_{IR-}(x, y) = F_{IR} I_{0-} \quad (17.7)$$

and

$$F_{IR} + F_{VIS} = 1 \quad (17.8)$$

where F_{IR} and F_{VIS} are the fractions of infrared (750nm to 2500nm) and visible (300nm to 750nm) radiation downwelling from the surface ocean. Although Morel and Antoine (1994) note that water vapor, zenith angle, and aerosol content each can effect the fraction of incoming radiation that is represented by infrared

and visible light, in the present implementation we have chosen to keep these fractions constant such that $F_{IR} = 0.46$.

The second and third exponentials represent a parameterization of the attenuation function for downwelling radiation in the visible range (300nm -750nm) in the following form:

$$I_{VIS}(x, y, z) = I_{VIS-}(x, y) (V_1 e^{z/\zeta_1} + V_2 e^{z/\zeta_2}). \quad (17.9)$$

This form further partitions the visible radiation into long (V_1) and short (V_2) wavelengths assuming

$$V_1 + V_2 \equiv 1. \quad (17.10)$$

V_1 , V_2 , ζ_1 and ζ_2 are calculated from an empirical relationship as a function of chlorophyll-a concentration using methods from [Morel and Antoine \(1994\)](#). Throughout most of the ocean $V_1 < 0.5$ and $V_2 > 0.5$. The e-folding length scales ζ_1 and ζ_2 are the e-folding depths of the long (ζ_1) and short visible and ultra violet (ζ_2) wavelengths. Based on the chlorophyll-a climatology used in the GFDL models, ζ_1 should not exceed 3m while ζ_2 will vary between 30m in oligotrophic waters and 4m in coastal regions. All of these constants are based on satellite estimates of chlorophyll-a plus Pheophytin-a, as well as parameterizations which have “nonuniform pigment profiles” ([Morel and Antoine \(1994\)](#)). The “nonuniform pigment profiles” have been proposed to account for deep chlorophyll maxima that are often observed in highly stratified oligotrophic waters ([Morel and Berthon \(1989\)](#)).

17.3.3 SeaWiFS based chlorophyll-a climatology

A “non- El Niño” chlorophyll-a climatology was produced from estimates of the Sea-viewing Wide Field-of-view Sensor (SeaWiFS). The “non - El-Niño” climatology is based on 8-day composites of SeaWiFS images taken from 1999 to the end of 2001. The climatology calculates the weighted average chlorophyll-a concentration on the 15th day of each month considering all data 16 days before, and after the 15th day of each month. Each 8-day composite is assigned a gaussian ($\alpha = 2.5$) weight based on its proximity to the 15th day of each month. This data set is available from the GFDL NOMADS server where MOM datasets are distributed.

17.4 Diagnosing shortwave heating in MOM

It is of interest to diagnose the impacts of shortwave heating on the ocean fluid, both at the surface and beneath. As detailed here, there are two terms needed to fully diagnose the impacts of shortwave heating.

- **SURFACE NET DOWNWARD SHORTWAVE FLUX:** This is the surface net downward shortwave flux $I_0-(x, y)$ discussed in equation (17.1), and it is the shortwave heat flux entering through the top of the surface ocean model grid cell. This heat flux is handled by `ocean_core/ocean_sbc.F90`, and it contributes to the `stf` array as part of the temperature derived type.

The diagnostic table entry for $I_0-(x, y)$ is `swflx`, which is the surface net downward shortwave flux. If there was zero penetrative shortwave radiation through the ocean column, then this flux would represent the density and thickness weighted convergence of shortwave flux impacting the surface model grid cell. However, penetration is the norm, so there is another contribution to the net impacts of shortwave radiation on the ocean.

- **DOWNWARD SHORTWAVE FLUX IN SEAWATER:** This is the density and thickness weighted flux convergence (units of $W\ m^{-2}$) of shortwave heat that impacts a tracer grid cell, arising from the penetration of shortwave radiation beneath the surface grid cell. This flux convergence is computed in `ocean_param/sources/ocean_shortwave.F90`.

Because MOM typically includes the surface shortwave flux $I_0-(x, y)$ in the treatment of surface boundary conditions ($I(\eta) = 0$ in equation (17.4)), the surface shortwave flux is excluded from the flux convergence computed in `ocean_param/sources/ocean_shortwave.F90` (equation (17.2)) in order to avoid double counting. The diagnostic `sw_heat` from `ocean_param/sources/ocean_shortwave.F90` will thus have a negative flux convergence in the surface grid cell, since this convergence is computed with a zero flux crossing the top of the surface grid cell.

- TOTAL IMPACTS OF SHORTWAVE RADIATION: Assuming the traditional MOM approach whereby surface shortwave flux $I_0^-(x, y)$ is included in the `ocean_core/ocean_sbc.F90` module ($I(\eta) = 0$ in equation (17.4)), to diagnose the full impacts of shortwave radiation requires the sum of two terms

$$\text{net shortwave radiation heating at } k = 1 = \text{swflx} + \text{sw_heat}(k = 1). \quad (17.11)$$

$$\text{net shortwave radiation heating at } k > 1 = \text{sw_heat}(k > 1). \quad (17.12)$$

KPP FOR THE SURFACE OCEAN BOUNDARY LAYER (OBL)

Contents

18.1	Elements of the K-profile parameterization (KPP)	256
18.1.1	Conventions	257
18.1.2	General form of the parameterization	257
18.1.3	The vertical diffusivity	258
18.1.3.1	Boundary layer thickness	258
18.1.3.2	Measuring vertical distances within the OBL	258
18.1.3.3	Vertical turbulent velocity scale w_λ	259
18.1.3.4	Non-dimensional vertical shape function $G_\lambda(\sigma)$	260
18.1.4	The non-local transport γ_λ	260
18.2	Surface ocean boundary momentum fluxes	261
18.3	Surface ocean boundary buoyancy fluxes	262
18.3.1	General features of buoyancy forcing	262
18.3.2	Temperature, salinity, and mass budget for a surface ocean model grid cell	263
18.3.3	Salt fluxes from sea ice melt and formation	264
18.3.4	Salt and heat fluxes associated with water transport	264
18.3.5	Non-penetrative surface heat fluxes	264
18.3.5.1	Longwave radiation	264
18.3.5.2	Latent heat fluxes	265
18.3.5.3	Sensible heat fluxes	265
18.3.5.4	Heating from frazil	265
18.3.6	Penetrative shortwave heating	265
18.3.7	Buoyancy budget for a surface ocean model grid cell	265
18.3.8	Surface boundary terms contributing to ocean buoyancy evolution	266
18.3.8.1	Heat carried by water transport	266
18.3.8.2	Salt carried by water transport	267
18.3.8.3	Penetrative radiation	267
18.3.8.4	Non-penetrative heating	267
18.3.8.5	Salt fluxes due to sea ice melt or formation	267
18.3.9	Buoyancy forcing that acts on the OBL	268
18.4	Surface layer and Monin-Obukhov similarity	268
18.4.1	The surface layer	269

18.4.2	Monin-Obukhov similarity theory	269
18.4.3	Similarity functions and length scale	270
18.5	Specifying the KPP parameterization	272
18.5.1	The turbulent vertical velocity scale w_λ	272
18.5.1.1	Velocity scale with stable buoyancy forcing	272
18.5.1.2	Velocity scale with unstable buoyancy forcing	273
18.5.1.3	Summarizing properties of the turbulent velocity scale	273
18.5.2	Similarity functions ϕ_λ	275
18.5.2.1	The Large et al. (1994) choices for unstable buoyancy forcing	275
18.5.2.2	Alternative choices for unstable buoyancy forcing	276
18.5.3	The shape function $G_\lambda(\sigma)$	276
18.5.4	The non-local transport γ_λ	278
18.5.4.1	General features of γ_λ with the KPP parameterization	278
18.5.4.2	Potential problems with the parameterized non-local transport	279
18.5.5	The bulk Richardson number and the OBL thickness h	279
18.5.5.1	Local gravitational stability	279
18.5.5.2	Non-local gravitational stability	281
18.5.5.3	The Ri_b calculation	281
18.5.5.4	Unresolved shear U_t	282
18.5.5.5	Restrictions on h unstable stable buoyancy forcing	282

This chapter summarizes the KPP scheme originally proposed for the ocean surface boundary layer by Large et al. (1994) as well as Large (1998). The material here forms the beginning of a community research and development effort to merge elements of the KPP parameterization as implemented in MOM, POP, and other codes. This community effort intends to remove some complications of the various KPP implementations, introduce hooks for new physical features, and remove limitations and/or bugs in the various codes.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_param/vertical/ocean_vert_kpp_mom4p0.F90
ocean_param/vertical/ocean_vert_kpp_mom4p1.F90
ocean_param/vertical/ocean_vert_kpp_test.F90.
```

The modules `ocean_vert_kpp_mom4p0.F90` and `ocean_vert_kpp_mom4p1.F90` are frozen based on implementations in MOM4.0 and MOM4p1. The module `ocean_vert_kpp_test.F90` will be changing as a function of the ongoing research and development mentioned above. It is notable that the documentation is somewhat further ahead than the code, in that some of the material presented in this chapter yet to be implemented and tested.

18.1 Elements of the K-profile parameterization (KPP)

The ocean surface boundary layer (OBL) mediates the exchange of properties between the ocean and other components of the climate system. Hence, parameterization of processes active in the OBL are fundamental to the integrity of a climate simulation. The K-profile parameterization (KPP) is a widely used method for parameterizing boundary layer processes in both the atmosphere and ocean. The paper by Large et al. (1994) introduced this scheme to the ocean community for use in parameterizing processes in the surface ocean boundary layer. The pedagogical lecture by Large (1998) provides added insight into the scheme that complements some of the material in Large et al. (1994).

The KPP scheme has been used by many ocean climate studies for parameterizing mixing in the OBL, with examples discussed in Large et al. (1997), Holland et al. (1998), Gent et al. (1998), Umlauf et al. (2005),

Li et al. (2001), Smyth et al. (2002), Durski et al. (2004), Chang et al. (2005)). We consider here only the implementation of KPP for the surface ocean boundary layer, as implementations for the bottom do not exist in MOM, nor are they well documented in the peer-review.

18.1.1 Conventions

We use the following conventions that are consistent with Large et al. (1994) and Large (1998).

- The fluid is assumed to be volume conserving Boussinesq, with extensions to a mass conserving non-Boussinesq fluid trivial.
- The vertical direction, z , increases up with $z = 0$ defining the resting ocean surface. The ocean free surface is defined by $z = \eta(x, y, t)$ and the static ocean bottom is at $z = -H(x, y)$.
- A lowercase λ is used to denote a turbulent fluctuation of an arbitrary field within the surface ocean boundary layer; e.g., a tracer such as potential or conservative temperature θ and salinity s , or a velocity component (u, v, w) . Note that x is the notation used in Large et al. (1994) and Large (1998), but we prefer the Greek letter λ to avoid confusion with the horizontal spatial coordinate.
- An uppercase Λ is used to denote the Eulerian mean of a tracer or velocity component within the surface ocean boundary layer; e.g., potential or conservative temperature Θ , salinity S , or velocity component (U, V, W) . The Eulerian mean fields are time stepped by an ocean climate model within the boundary layer, and correlations of turbulent variables must be parameterized to close the mean field equations.
- The expression $\overline{w\lambda}$ is used to symbolize the Eulerian correlation of the fluctuating turbulent vertical velocity and a fluctuating scalar or vector field. This correlation appears in the mean field time tendency equation for Λ in the Boussinesq primitive ocean equations (see equation (18.2)). KPP provides a parameterization of this vertical turbulent flux within the surface ocean boundary layer.
- The mean and turbulent vertical velocity components, W, w , are positive for upward motion. This sign convention implies that

$$\overline{w\lambda} > 0 \implies \text{turbulent flux for } \lambda \text{ transported vertically upward.} \quad (18.1)$$

If λ is the temperature, then a positive correlation at the ocean surface, $\overline{w\theta}^{(\eta)} > 0$, corresponds to surface cooling.

18.1.2 General form of the parameterization

Ignoring all terms except vertical advective transport in the prognostic equation for the mean field Λ , its time tendency is determined by

$$\frac{\partial \Lambda}{\partial t} = - \left(\frac{\partial (W \Lambda)}{\partial z} \right) - \left(\frac{\partial (\overline{w\lambda})}{\partial z} \right). \quad (18.2)$$

The advective flux by the mean vertical velocity, $W \Lambda$, is represented via a numerical advection operator. In contrast, the turbulent correlation, $\overline{w\lambda}$, is a subgrid scale flux that must be parameterized in order to close the equation for Λ . Here, the overbar signifies an Eulerian averaging operator over unresolved turbulent motions occurring within the OBL.

The KPP scheme provides a first order closure for $\overline{w\lambda}$ within the OBL. It does so by introducing two terms in the following manner

$$\overline{w\lambda} = -K_\lambda \left(\frac{\partial \Lambda}{\partial z} - \gamma_\lambda \right). \quad (18.3)$$

In effect, the KPP parameterization (18.3) splits the vertical turbulent flux into two terms

$$\overline{w\lambda} = \overline{w\lambda}^{\text{local}} + \overline{w\lambda}^{\text{non-local}}. \quad (18.4)$$

The first term provides for the familiar downgradient vertical diffusion determined by a vertical diffusivity and the local vertical derivative of the mean field. This term is referred to as the local portion of the parameterization

$$\overline{w\lambda}^{\text{local}} = -K_\lambda \left(\frac{\partial \Lambda}{\partial z} \right), \quad (18.5)$$

even though the diffusivity is a non-local function of boundary layer properties. The second term, γ_λ , accounts for non-local transport that is not directly associated with local vertical gradients of Λ , in which we have

$$\overline{w\lambda}^{\text{non-local}} = K_\lambda \gamma_\lambda \quad (18.6)$$

We next provide a general discussion of these two contributions to the KPP parameterization.

18.1.3 The vertical diffusivity

The vertical diffusivity arising from KPP in the OBL is determined as a non-local function of boundary layer properties. It is written in the following form

$$K_\lambda(\sigma) = h w_\lambda(\sigma) G_\lambda(\sigma). \quad (18.7)$$

The diffusivity is constructed as the product of three terms: the boundary layer thickness h , the vertical turbulent velocity scale $w_\lambda(\sigma)$, and the vertical shape function $G_\lambda(\sigma)$. Note that we introduce a dependence of the shape function on the field diffused. As discussed in Section 18.5.3, this dependence arises from matching to interior diffusivities, which generally differ as a function of λ .

18.1.3.1 Boundary layer thickness

The boundary layer thickness is denoted by

$$h \geq 0 \text{ is the boundary layer thickness.} \quad (18.8)$$

This is the thickness of the OBL prescribed by the KPP scheme, with details given in Section 18.5.5. The direct dependence of the vertical diffusivity in equation (18.7) on the OBL thickness manifests the common property of boundary layers, whereby thicker layers generally arise from stronger eddy motions and are thus associated with more rapid mixing of tracer concentration and momentum.

Figure 18.1 provides a schematic of the KPP boundary layer, the Monin-Obukhov surface layer, and the associated momentum, mass, and buoyancy fluxes impacting these layers. Details of this figure will be explored in the following.

18.1.3.2 Measuring vertical distances within the OBL

When measuring distances within the boundary layer, it is the thickness of the water as measured from the ocean surface that is important. Free surface undulations can be a nontrivial fraction of the boundary layer thickness, particularly under conditions of stable buoyancy forcing. Hence, we make explicit note that the ocean has an undulating free surface at $z = \eta(x, y, t)$, which contrasts to Large et al. (1994) and Large (1998), where it is assumed that $z = 0$ sets the upper ocean surface.

Following Large et al. (1994), we introduce the non-dimensional depth, σ , given by

$$\sigma = \frac{d}{h}. \quad (18.9)$$

In this definition, $d \geq 0$ is the distance from the ocean surface at $z = \eta$ to a point within the boundary layer

$$d = -z + \eta. \quad (18.10)$$

Likewise, $h \geq 0$ is the distance from the free surface to the bottom of the boundary layer

$$h = h_{\text{obl}} + \eta, \quad (18.11)$$

where h_{obl} is the depth of the boundary layer as measured from $z = 0$. That is, h is the thickness of the OBL, and it is this thickness, not h_{obl} , that is predicted by KPP (Section 18.5.5). Regions within the boundary layer are given by the non-dimensional depth range

$$0 \leq \sigma \leq 1 \quad \text{within boundary layer,} \quad (18.12)$$

with $\sigma = 0$ the ocean surface and $\sigma = 1$ the bottom of the boundary layer.

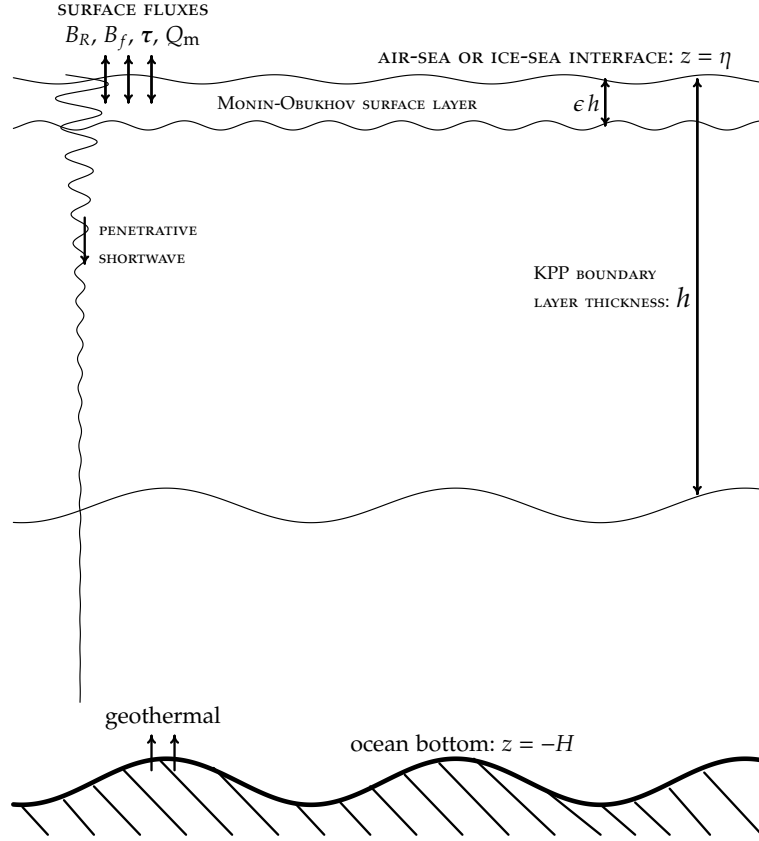


Figure 18.1: Schematic of the upper ocean boundary layer regions associated with the KPP boundary layer parameterization. The upper ocean is exposed to non-penetrative air-sea and ice-sea fluxes of momentum τ (Section 18.2), mass Q_m (Section 18.3), and buoyancy B_f (Section 18.3). In addition, there is penetrative shortwave radiation, $-\overline{w\theta_R}$ (Section 18.3), indicated by the exponentially decaying vertical sinusoidal. The Monin-Obukhov surface layer (Section 18.4) has a thickness ϵh , with $\epsilon \approx 0.1$. The surface layer is where turbulence delivers fluxes to the molecular skin layer for transfer to the atmosphere or ice. The surface layer starts from just beneath the surface roughness elements at the upper ocean interface. Since neither these roughness elements, nor the molecular viscous sublayer, are resolved in ocean models, we assume in practice that the Monin-Obukhov surface layer extends to the sea surface at $z = \eta(x, y, t)$. The KPP boundary layer includes the surface layer, and it has a thickness $h(x, y, t)$ determined by the KPP parameterization (Section 18.5.5). The ocean bottom at $z = -H(x, y)$ is rigid and is exposed to geothermal heating. Presently, the KPP boundary layer scheme has not been implemented in MOM to parameterize bottom boundary layer physics, though nothing fundamental precludes such. In fact, Durski et al. (2004) provide just such an implementation.

18.1.3.3 Vertical turbulent velocity scale w_λ

The velocity scale w_λ is a function of depth within the boundary layer, and a function of the field to which it refers. We return to its specification in Section 18.5.1.

18.1.3.4 Non-dimensional vertical shape function $G_\lambda(\sigma)$

Non-dimensional vertical shape function $G_\lambda(\sigma)$ is used to smoothly transition from the ocean surface to the bottom of the boundary layer. [Large et al. \(1994\)](#) chose a cubic polynomial

$$G_\lambda(\sigma) = a_0 + a_1 \sigma + a_2 \sigma^2 + a_3 \sigma^3. \quad (18.13)$$

Since turbulent eddies do not cross the ocean surface at $\sigma = 0$, we should correspondingly have a vanishing diffusivity at $\sigma = 0$. This constraint is satisfied by setting

$$a_0 = 0. \quad (18.14)$$

We detail in Section later how to specify the remaining expansion coefficients a_1, a_2, a_3 . In particular, we simplify the specification of [Large et al. \(1994\)](#), with their approach more complex than justified physically.

18.1.4 The non-local transport γ_λ

Section 2 of [Large et al. \(1994\)](#) notes the presence of many processes in the boundary layer that lead to nonlocal transport. This behaviour leads to a diffusivity K_λ that is a function of the surface fluxes and boundary layer thickness h . Furthermore, under convective forcing (negative surface buoyancy forcing; $B_f < 0$), fluxes can penetrate into stratified interior. This characteristic then motivates the introduction of a non-local transport term γ_λ to the KPP parameterization (equation (18.3)) when $B_f < 0$. To further identify the need for a non-local transport term γ_λ , we reproduce Figure 1 from [Large et al. \(1994\)](#), here shown as Figure 18.2. The caption to Figure 18.2 explores the many facets of this figure used to help justify the non-local term in KPP.

As part of the KPP parameterization, the non-local transport, γ_λ , aims to account for such processes as boundary layer eddies whose transport may be unrelated to the local vertical gradient of the mean field, and whose impacts may penetrate within the stratified ocean interior. In general, [Large et al. \(1994\)](#) prescribe the following characteristics to γ_λ .

- Page 371 of ([Large et al., 1994](#)) notes that there is no theory for non-local momentum transport, and so the non-local transport directly affects only the tracer fields:

$$\gamma_\lambda = \begin{cases} 0 & \text{if } \lambda = (u, v, w) \text{ a velocity component} \\ \neq 0 & \text{nonzero if } \lambda = \theta, s \text{ or another tracer.} \end{cases} \quad (18.15)$$

However, [Smyth et al. \(2002\)](#) consider a non-local term for momentum, thus motivating further research to see whether it is suitable for climate modeling.

- The non-local transport is non-zero only within the OBL:

$$\gamma_\lambda = \begin{cases} 0 & \text{if } \sigma > 1 \\ \neq 0 & \text{if } 0 \leq \sigma \leq 1. \end{cases} \quad (18.16)$$

- The non-local transport is non-zero only in the presence of destabilizing negative surface ocean buoyancy flux, whose presence gives rise to convective mixing:

$$\gamma_\lambda = \begin{cases} 0 & \text{for positive (stabilizing) surface buoyancy forcing} \\ \neq 0 & \text{for negative (destabilizing) surface buoyancy forcing.} \end{cases} \quad (18.17)$$

- The non-local transport can give rise, under certain conditions, to either down-gradient or up-gradient transport of the mean tracer field. Hence, it can either act to smooth gradients of mean fields (down-gradient non-local fluxes) or enhance gradients (upgradient non-local fluxes).

In Section 18.5.4, we provide to the KPP parameterization of γ_λ .

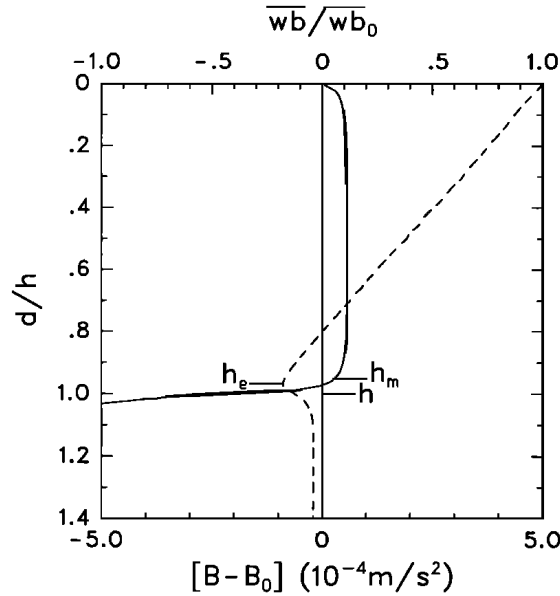


Figure 18.2: This is a reproduction of Figure 1 from [Large et al. \(1994\)](#). The figure is derived from a one-dimensional simulation after 3 days of convective deepening (zero winds; negative surface buoyancy forcing) into initially uniformly stratified water column. The vertical axis is vertical distance starting from the ocean surface interface at $z = \eta$ and $d = 0$, extending down to $d = h$ ($h = 13.6$ m at this point of the integration), which is the base of the boundary layer, and finally to $d = 1.4h$, which is beneath the boundary layer.

The horizontal axis on the bottom is the mean buoyancy, B , relative to that at the surface, B_0 , and the profile is depicted by the solid line. Positive values of $B - B_0$ indicate that the mean buoyancy at a point is larger than at the surface, with $B - B_0 > 0$ expected under negative buoyancy forcing at the ocean surface.

The horizontal axis on the top is the ratio of the local turbulent buoyancy flux \overline{wb} to the surface turbulent flux \overline{wb}^η (denoted \overline{wb}_0 by [Large et al. \(1994\)](#)). The dashed line depicts this ratio. Positive values of \overline{wb} represent upward turbulent buoyancy fluxes; e.g., upward fluxes of heat for the case where buoyancy is determined by temperature, and the thermal expansion coefficient is positive.

Positive values for \overline{wb} in regions between roughly $0.35 < d < 0.8$ represent upward turbulent buoyancy fluxes in a region where the mean vertical gradient of B is nearly zero, thus indicating non-local turbulent transport. In shallower regions with $d < 0.35$, the mean gradient is negative, $\partial_z B < 0$, and the fluxes are positive, $\overline{wb} > 0$, thus representing downgradient turbulent fluxes. Likewise, for $d > 0.8$, the turbulent fluxes are downgradient.

The mixed layer depth is denoted by h_m , though this depth is subject to arbitrary specification of the density difference. The entrainment depth is h_e , with this depth taken where the buoyancy flux reaches a negative extrema. Note that it is an empirical result that under pure convective forcing ($\tau = 0, B_f < 0$), the turbulent entrainment flux is roughly 20% of the surface flux: $\overline{wb}^{d=h_e} = -\beta_T \overline{wb}^{d=0}$, where $\beta_T = 0.2$. This situation is depicted in the figure.

18.2 Surface ocean boundary momentum fluxes

In this section and Section 18.3, we present features of how surface boundary fluxes force the upper ocean, largely following Appendix A of [Large et al. \(1994\)](#). The aim is to identify how surface boundary fluxes impact the upper ocean, with this characterization then used in Section 18.4 to help establish some basic features of ocean boundary layers. These ideas are then used in Section 18.5 to specify the diffusivity and

non-local transport from the KPP parameterization.

Vertical exchange of momentum across the atmosphere-ocean or sea-ice-ocean boundary occurs largely through turbulent processes. The resulting horizontal stress vector acting on the ocean, τ , is determined through application of a bulk formula (e.g., see Appendix C of [Griffies et al., 2009](#)). For our purposes, we assume τ is given, thus yielding the ocean kinematic fluxes associated with the turbulent transport of momentum across the ocean surface

$$-\overline{w\mathbf{u}}^\eta = \left(\frac{\tau}{\rho(\eta)} \right) \approx \left(\frac{\tau}{\rho_0} \right). \quad (18.18)$$

In this equation, $\rho(\eta)$ is the surface ocean density, which is commonly approximated by the constant Boussinesq reference density ρ_0 . A positive sign on a component of τ acts to accelerate the flow in the respective direction, whereas a positive sign to a component of $\overline{w\mathbf{u}}^\eta$ removes momentum from the ocean. These sign conventions give rise to the minus sign in the relation (18.18). In addition to defining the kinematic surface fluxes, knowledge of τ allows us to compute surface boundary layer velocity scales when working within the Monin-Obukhov similarity theory (Section 18.4.2).

In addition to turbulent momentum transfer, momentum is also transported through mass exchange across the ocean surface, since water transported across the ocean generally carries a nonzero momentum. This advective momentum boundary exchange is typically ignored for climate models, though [Kantha and Clayson \(2000\)](#) (see their page 431) make the case for including this effect, particularly when resolving strong atmospheric storms. They also make the case for including this effect in computing the Monin-Obukhov length scale defined by equation (18.65) (see their equation (4.3.11)).

18.3 Surface ocean boundary buoyancy fluxes

Turbulent and advective fluxes of momentum and buoyancy are transferred across the upper ocean surface boundary, with ocean processes such as advection and mixing then transporting the boundary momentum and buoyancy laterally as well as into the ocean interior. In contrast, penetrative shortwave radiation is absorbed into the ocean absent ocean transport processes, with such absorption a function of ocean optical properties. In the unphysical case of perfectly transparent seawater, shortwave radiation penetrates through the boundary layer and so has no influence on boundary layer processes. In realistic cases, much of the shortwave radiation is absorbed in the boundary layer, with only a fraction leaking through to the interior. In general, such non-turbulent and non-advective transport of buoyancy via penetrative radiation represents a fundamentally novel aspect of ocean boundary layer physics relative to the atmosphere. Namely, for the atmosphere, radiative absorption is far less relevant than in the upper ocean, since the atmosphere is largely transparent to radiation. We therefore consider penetrative shortwave radiation as distinct from other buoyancy fluxes when formulating how boundary fluxes impact the ocean.

18.3.1 General features of buoyancy forcing

The buoyancy of a fluid is commonly defined as (e.g., page 83 of [Large \(1998\)](#))

$$B = g \left(\frac{\rho_0 - \rho}{\rho_0} \right), \quad (18.19)$$

where g is the constant gravitational acceleration, and ρ_0 is a reference density, taken here to equal the Boussinesq reference density. A reduction in density is associated with an increase in buoyancy; that is, the water becomes more *buoyant*. Changes in buoyancy arise through changes in density associated with temperature and salinity changes, since buoyancy changes are computed relative to a fixed pressure level. In this way, buoyancy changes are directly related to processes that impact locally referenced potential density.

Ocean buoyancy is affected through surface ocean heat, salt, and water fluxes.

- Turbulent processes transfer heat through latent and sensible heating.
- Longwave radiation cools the upper ocean, with this radiation affected by the upper ocean boundary temperature.

- Penetrative shortwave radiation is absorbed in seawater.
- The transfer of salt occurs when sea ice melts and forms. This transfer is proportional to the water mass flux and the difference in salinity between the liquid ocean and sea ice. More generally, we simply consider this to be a salt flux between sea ice and ocean, with this flux operationally computed as part of a sea ice model.
- Advective processes transfer heat and salt across the ocean surface through the transfer of water mass across the interface.

We further detail these fluxes in the following.

18.3.2 Temperature, salinity, and mass budget for a surface ocean model grid cell

Buoyancy is not a prognostic variable in ocean models. So to develop a quantitative understanding of how buoyancy is impacted by surface fluxes, we consider the evolution of temperature, salinity, and mass in an arbitrary top model grid cell, and focus exclusively on evolution arising from surface boundary fluxes. We write these budgets in their finite volume sense (as implemented in MOM), which includes density and thickness weighting

$$\partial_t (\rho dz \Theta) = Q_m \Theta_m - Q_{\theta}^{\text{non-pen}} + (Q_{\theta}^{\text{pen}}(z = \eta) - Q_{\theta}^{\text{pen}}(z = -\Delta z)) \quad (18.20)$$

$$\partial_t (\rho dz S) = Q_m S_m - Q_S \quad (18.21)$$

$$\partial_t (\rho dz) = Q_m. \quad (18.22)$$

We now detail the terms appearing in these equations.

- ρdz is the mass per horizontal area of seawater in the grid cell. For a volume conserving Boussinesq fluid, ρ is set to the constant reference density ρ_0 .
- Θ is the grid cell potential temperature or conservative temperature.
- S is the grid cell salinity.
- Q_m is the mass flux ($\text{kg m}^{-2} \text{sec}^{-1}$) of water crossing the ocean surface, with $Q_m > 0$ for water entering the ocean (as when precipitation plus runoff exceeds evaporation).
- Θ_m is the temperature of water crossing the ocean surface, and $C_p Q_m \Theta_m$ is the associated heat flux (W m^{-2}). We further discuss this heat flux in Section 18.3.4.
- S_m is the salinity of water crossing the ocean surface, and $Q_m S_m$ is the associated mass flux. Note that S_m is typically taken to be zero, as for precipitation and evaporation. However, rivers can contain a nonzero salt concentration, so we keep S_m for the following formulation. We further discuss this salt flux in Section 18.3.4.
- C_p is the seawater heat capacity at constant pressure ($\text{J kg}^{-1} \text{°C}^{-1}$). IOC et al. (2010) provides the most precise value appropriate for an ocean with heat measured through conservative temperature.
- Q_S is the flux of salt ($\text{kg m}^{-2} \text{sec}^{-1}$) that leaves the ocean through the ocean surface. This flux arises in the transfer of salt when sea ice forms and melts. We further discuss this salt flux in Section 18.3.3.
- $C_p Q_{\theta}^{\text{non-pen}}$ is the non-penetrative surface heat flux associated with turbulent processes (latent and sensible) and radiative longwave cooling (W m^{-2}). The sign convention is chosen so that $Q_{\theta}^{\text{non-pen}} > 0$ for heat leaving the ocean surface (i.e., ocean cooling). We further discuss this heat flux in Section 18.3.5.
- $C_p Q_{\theta}^{\text{pen}}(z = \eta)$ is the radiative shortwave heat flux (W m^{-2}) entering the ocean through its surface at $z = \eta$, with $Q_{\theta}^{\text{pen}}(\eta) > 0$ warming the ocean surface. Likewise, $C_p Q_{\theta}^{\text{pen}}(z = -\Delta z)$ is the radiative shortwave heat flux leaving the top cell through its bottom face. We further discuss this heat flux in Section 18.3.6.

18.3.3 Salt fluxes from sea ice melt and formation

The mass flux of salt Q_S ($\text{kg m}^{-2} \text{sec}^{-1}$) is positive for salt leaving the ocean surface. There is transport of salt across the ocean surface when sea ice forms and melts, due to the nonzero salt content in sea ice. Otherwise, the surface salt flux is generally zero for the large scale ocean. For ocean models, however, the salt flux can be nonzero when formulating the surface boundary in terms of virtual salt fluxes rather than real water fluxes (Huang, 1993; Griffies et al., 2001). This formulation is not recommended, as it is distinctly unphysical and unnatural when using an explicit free surface or bottom pressure solver as in MOM.

18.3.4 Salt and heat fluxes associated with water transport

In most cases, salinity in the water fluxed across the ocean surface is zero, so that $S_m = 0$. However, there are some cases where rivers have a nonzero salinity so that $S_m \neq 0$ and the product $Q_m S_m$ leads to an advective transport of salt across the ocean surface.

Since water transported across the ocean has a nonzero heat content, this transport in turn affects the net heat content in the upper ocean. One can either prescribe the temperature of this water, Θ_m , or the product $Q_m \Theta_m$. Consider the case where the product is specified for river water entering the ocean, which is the case with the GFDL land model. In this case, the heat flux with respect to 0°C (in units of W m^{-2}) of liquid river runoff $\mathcal{H}^{\text{liquid runoff}}$ is given to the ocean from the land model, so that

$$Q_m \Theta_m = \frac{\mathcal{H}^{\text{liquid runoff}}}{C_p^{\text{liquid runoff}}}, \quad (18.23)$$

with $C_p^{\text{liquid runoff}}$ the heat capacity of the water coming in from the river runoff. Likewise, if the heat associated with frozen runoff (e.g., calving land ice) is provided by the land model, then we have

$$Q_m \Theta_m = \frac{\mathcal{H}^{\text{solid runoff}}}{C_p^{\text{solid runoff}}}, \quad (18.24)$$

with $C_p^{\text{solid runoff}}$ the heat capacity of the solid runoff. These two heat capacities are typically provided by the component model (i.e., the land model) used to compute the runoff fields. Similar considerations hold for transfer of water between sea ice models and the ocean.

18.3.5 Non-penetrative surface heat fluxes

The heat flux $C_p Q_\theta^{\text{non-pen}}$ (W m^{-2}) is defined with a sign so that it is positive for heat leaving the ocean. This flux is comprised of the following contributions (see page 34 of Gill, 1982)

$$C_p Q_\theta^{\text{non-pen}} = Q_{\text{long}} + Q_{\text{latent}} + Q_{\text{sens}} + Q_{\text{frazil}}. \quad (18.25)$$

Longwave, latent, and sensible heat fluxes are typically deposited or withdrawn from the ocean surface layer (Section 18.4). In practice, ocean models assume these fluxes are taken entirely from the surface grid cell. Frazil is slightly different, as it represents the heat exchanged during the production of sea ice, and sea ice can generally form at various levels in the upper ocean. Many ocean models assume frazil production occurs just in the top grid cell. But that assumption is not fundamental, nor is it correct when models refine their vertical grid spacing. We thus allow for frazil to have a depth dependence.

All of these fluxes are termed non-penetrative, since they are deposited or withdrawn from the liquid ocean at a particular depth. Transport of the boundary buoyancy to another depth occurs only through the action of ocean transport processes, such as advection or mixing. This behaviour contrasts to that of penetrative shortwave radiation, which is transferred to depths as a function of seawater optics, so does not depend on ocean transport. We now comment in a bit more detail on the various non-penetrative fluxes.

18.3.5.1 Longwave radiation

Q_{long} is the longwave radiation leaving the ocean in the form of the $\sigma_{\text{sb}} T^4$ Stefan-Boltzmann Law, so that Q_{long} is typically positive, thus generally cooling the ocean surface.

18.3.5.2 Latent heat fluxes

Q_{latent} arises from phase changes whereby liquid seawater either evaporates, or it acts to melt frozen precipitation. When seawater evaporates, the latent heat lost by the ocean is determined by the latent heat of vaporization for fresh water

$$H^{\text{vapor}} = 2.5 \times 10^6 \text{ J kg}^{-1}, \quad (18.26)$$

so that

$$Q_{\text{evap}} = H^{\text{vapor}} Q_{\text{m}}^{\text{evap}} \quad (18.27)$$

where $Q_{\text{m}}^{\text{evap}}$ is the mass flux ($\text{kg m}^{-2} \text{ sec}^{-1}$) of fresh water leaving the ocean due to evaporation. A similar expression holds when seawater melts frozen precipitation (e.g., snow), in which case

$$H^{\text{fusion}} = 3.34 \times 10^5 \text{ J kg}^{-1}, \quad (18.28)$$

so that

$$Q_{\text{melt}} = H^{\text{fusion}} Q_{\text{m}}^{\text{frozen precip}}, \quad (18.29)$$

where $Q_{\text{m}}^{\text{frozen precip}}$ is the mass flux ($\text{kg m}^{-2} \text{ sec}^{-1}$) of frozen precipitation falling onto the ocean surface. Both Q_{evap} and Q_{melt} are positive, indicating that they act to cool the ocean.

18.3.5.3 Sensible heat fluxes

Q_{sens} is the sensible heat transfer proportional to the difference between atmosphere and ocean temperatures. Sensible heating generally acts to cool the ocean, particularly near western boundary currents such as the Gulf Stream, Kuroshio, and Agulhas.

18.3.5.4 Heating from frazil

As the temperature of seawater cools to the freezing point, sea ice is formed, initially through the production of frazil ice. Operationally in an ocean model, liquid water can be supercooled at any particular time step through surface fluxes and transport. An adjustment process is used in the models to heat the liquid water back to the freezing point, with this positive heat flux $Q_{\text{frazil}} > 0$ extracted from the ice model as frazil sea ice is formed.

18.3.6 Penetrative shortwave heating

The penetrative shortwave radiative heat flux $C_p Q_{\theta}^{\text{pen}} > 0$ arises from the net shortwave radiation entering through the ocean surface and absorbed by seawater. This heat flux does *not* arise from turbulent or advective processes, which makes it distinct from other heat and salt fluxes impacting the ocean through its upper boundary. This radiation is not generally deposited entirely within the ocean surface layer or the top ocean model grid cell. Instead, a fraction of this radiation can penetrate to beneath the surface ocean grid cell, with the fraction depending on the optical properties of seawater. Hence, we subtract a heat flux $C_p Q_{\theta}^{\text{pen}}(z = -\Delta z)$, which represents the radiative shortwave heat flux passing through the bottom of the surface ocean cell at $z = -\Delta z$. It is the difference,

$$\text{net shortwave heating of surface grid cell} = C_p \left(Q_{\theta}^{\text{pen}}(z = \eta) - Q_{\theta}^{\text{pen}}(z = -\Delta z) \right) \quad (18.30)$$

that stays in the surface grid cell. When considering the same budget for the surface ocean boundary layer, we are interested in the shortwave flux that penetrates through the bottom of the boundary layer at $z = -h$.

18.3.7 Buoyancy budget for a surface ocean model grid cell

We now bring the previous fluxes together to form the budget for buoyancy in a surface grid cell due to the impacts of surface fluxes. The resulting expression is then used to derive an expression for the buoyancy forcing that acts on the ocean surface boundary layer. Buoyancy (equation (18.19)) has a time tendency given by

$$-\left(\frac{\rho_0}{g}\right) \frac{\partial B}{\partial t} = \rho_{,\Theta} \frac{\partial \Theta}{\partial t} + \rho_{,S} \frac{\partial S}{\partial t}, \quad (18.31)$$

where we introduced the shorthand notation

$$\rho_{,\Theta} = \left(\frac{\partial \rho}{\partial \Theta} \right)_{S,p} \quad (18.32)$$

$$\rho_{,S} = \left(\frac{\partial \rho}{\partial S} \right)_{\Theta,p} \quad (18.33)$$

for the partial derivatives of density with respect to conservative temperature and salinity, respectively, each with pressure held constant. We wish to form an evolution equation for buoyancy at the ocean surface grid cell just due to the effects of surface forcing. For this purpose, multiply the temperature equation (18.20) by $\rho_{,\Theta}$ and add to the surface salinity equation (18.21) multiplied by $\rho_{,S}$

$$\rho_{,\Theta} (\rho dz \Theta)_{,t} + \rho_{,S} (\rho dz S)_{,t} = Q_m (\rho_{,\Theta} \Theta_m + \rho_{,S} S_m) + \rho_{,\Theta} (-Q_\theta^{\text{non-pen}} + \delta_k Q_\theta^{\text{pen}}) - \rho_{,S} Q_S, \quad (18.34)$$

where we introduced the shorthand

$$\delta_k Q_\theta^{\text{pen}} = Q_\theta^{\text{pen}}(z = \eta) - Q_\theta^{\text{pen}}(z = -\Delta z). \quad (18.35)$$

We now use the mass budget (18.22) and introduce the buoyancy tendency according to equation (18.31) to realize an expression for the time tendency of the surface ocean buoyancy

$$(\rho_0/g) \rho dz \left(\frac{\partial B}{\partial t} \right) = Q_m [\rho_{,\Theta} (\Theta - \Theta_m) + \rho_{,S} (S - S_m)] + \rho_{,\Theta} (Q_\theta^{\text{non-pen}} - \delta_k Q_\theta^{\text{pen}}) + \rho_{,S} Q_S. \quad (18.36)$$

Now introduce the thermal expansion and saline contraction coefficients

$$\alpha = -\frac{1}{\rho} \left(\frac{\partial \rho}{\partial \Theta} \right)_{S,p} \quad (18.37)$$

$$\beta = \frac{1}{\rho} \left(\frac{\partial \rho}{\partial S} \right)_{\Theta,p} \quad (18.38)$$

to render

$$dz \left(\frac{\partial B}{\partial t} \right) = \frac{g}{\rho_0} (Q_m [-\alpha (\Theta - \Theta_m) + \beta (S - S_m)] + \alpha (\delta_k Q_\theta^{\text{pen}} - Q_\theta^{\text{non-pen}}) + \beta Q_S). \quad (18.39)$$

18.3.8 Surface boundary terms contributing to ocean buoyancy evolution

We now summarize the various surface boundary terms appearing on the right hand side of the surface buoyancy budget (18.39).

18.3.8.1 Heat carried by water transport

Assuming a positive thermal expansion coefficient, $\alpha > 0$, the term $-Q_m \alpha (\Theta - \Theta_m)$ reduces ocean buoyancy when adding water $Q_m > 0$ to the ocean that is colder than the surface ocean temperature, $\Theta = \Theta_{k=1}$. The opposite occurs in regions of cold fresh waters, such as the Baltic, where $\alpha < 0$. In such cases, adding water to the ocean that is colder than the sea surface temperature increases seawater buoyancy. We now consider in turn the three cases evaporation, precipitation, and liquid river runoff and indicate how they are typically treated in climate models.

- It is quite accurate to assume that evaporating water leaves the ocean at the sea surface temperature, so that

$$\Theta^{\text{evap}} = \Theta_{k=1}, \quad (18.40)$$

in which case there is no change to ocean buoyancy upon transfer of evaporating water across the ocean surface. This is the approach taken by all ocean climate models.

- Precipitating liquid water need not fall on the ocean at the sea surface temperature, so that

$$\Theta^{\text{precip}} \neq \Theta_{k=1} \quad \text{real world.} \quad (18.41)$$

Kantha and Clayson (2000) (see their page 429) discuss this difference, and the associated transfer of heat across the ocean due to rain events, particularly in the West Pacific. However, we know of no climate modeling application in which the atmospheric model component carries information about the temperature of its condensed water, nor the heat content of that water. Hence, operationally all climate modeling applications assume that

$$\Theta^{\text{precip}} = \Theta_{k=1} \quad \text{climate models,} \quad (18.42)$$

in which case there is no change in ocean buoyancy upon transfer of precipitating liquid water across the ocean surface.

- Realistic river models carry the heat content of river water and pass this content to the ocean model at river mouths. Following from the discussion surrounding equation (18.23), we may thus write the river contribution to the buoyancy budget in the form

$$-Q_m \alpha (\Theta - \Theta_m) = \alpha \left(-Q_m \Theta + \frac{\mathcal{H}^{\text{liquid runoff}}}{C_p^{\text{liquid runoff}}} \right). \quad (18.43)$$

Depending on the heat content of liquid runoff relative to the sea surface, ocean buoyancy may increase or decrease when liquid runoff enters the ocean.

18.3.8.2 Salt carried by water transport

The haline contraction coefficient, β , is generally positive. Hence, the term $Q_m \beta (S - S_m)$ increases ocean buoyancy for those cases where the sea surface salinity, $S_{k=1}$, is greater than the salinity of the water transferred across the ocean surface. Most applications assume $S_m = 0$, such as for evaporation and precipitation

$$S^{\text{evap}} = 0 \quad (18.44)$$

$$S^{\text{precip}} = 0. \quad (18.45)$$

However, river models sometimes consider a nonzero salinity of the runoff, in which case

$$S^{\text{liquid runoff}} \neq 0. \quad (18.46)$$

18.3.8.3 Penetrative radiation

Shortwave radiation is absorbed by seawater as it penetrates from the surface into the upper ocean. Hence, $\delta_k Q_\theta^{\text{pen}} > 0$ so that radiation increases the grid cell buoyancy.

18.3.8.4 Non-penetrative heating

Longwave, latent, and sensible heating generally cool the upper ocean, and so lead to a decrease in ocean buoyancy. In contrast, frazil heating in sea ice regions increases buoyancy. The net effect from the non-penetrative heat fluxes, $Q_\theta^{\text{non-pen}}$, can be to either increase or decrease buoyancy.

18.3.8.5 Salt fluxes due to sea ice melt or formation

Salt is exchanged with the ocean when sea ice melts and forms, so that the term βQ_S can either increase (when salt is removed from the liquid ocean) or decrease (when salt is added to the liquid ocean) buoyancy.

18.3.9 Buoyancy forcing that acts on the OBL

The expression (18.39) for the buoyancy forcing from surface fluxes acting on a surface grid cell is now extended to an expression for the buoyancy forcing on the OBL. The only subtle point concerns the treatment of penetrative shortwave radiation. Rather than consider that radiation leaving the bottom of the surface cell at $z = -\Delta z$, we are now concerned with that leaving the bottom of the boundary layer at $z = -h$. We also multiply this penetrative flux by the thermal expansion coefficient at that depth, rather than the expansion coefficient in the ocean surface cell. In this way we write the buoyancy forcing acting on the boundary layer

$$B_f = \frac{g}{\rho_0} \left[Q_m [-\alpha (\Theta - \Theta_m) + \beta (S - S_m)] - \alpha Q_\theta^{\text{non-pen}} + \beta Q_S \right] + \left[(\alpha Q_\theta^{\text{pen}})_{z=\eta} - (\alpha Q_\theta^{\text{pen}})_{z=-h} \right]. \quad (18.47)$$

This expression for the net buoyancy forcing acting on the boundary layer can be written as the sum of two terms

$$B_f = -\overline{w b}^\eta + B_R. \quad (18.48)$$

The first term takes the form of a kinematic turbulent flux at the ocean surface

$$-\overline{w b}^\eta = \frac{g}{\rho_0} \left[Q_m [-\alpha (\Theta - \Theta_m) + \beta (S - S_m)] - \alpha Q_\theta^{\text{non-pen}} + \beta Q_S \right], \quad (18.49)$$

where the minus sign on the left hand side accounts for the assumption that $w > 0$ for upward velocity. The second term accounts for the penetrative radiation, which is neither a turbulent flux nor advective flux

$$B_R = (\alpha Q_\theta^{\text{pen}})_{z=\eta} - (\alpha Q_\theta^{\text{pen}})_{z=-h}. \quad (18.50)$$

The corresponding heat flux convergence onto the boundary layer is given by (see equation (A4) of [Large et al. \(1994\)](#))

$$Q_R = (Q_\theta^{\text{pen}})_{z=\eta} - (Q_\theta^{\text{pen}})_{z=-h}. \quad (18.51)$$

Notably, B_R , and hence B_f , are two-dimensional functions of the boundary forcing, even though they depend on the depth to which the penetrative radiation extends.

18.4 Surface layer and Monin-Obukhov similarity

The semi-empirical Monin-Obukhov similarity theory has proven quite useful in describing general features of boundary layer turbulence active in the atmospheric planetary boundary layer (see, e.g., Section 3.3 of [Kantha and Clayson, 2000](#)). One may thus choose to apply these ideas to the ocean planetary boundary layer, particularly since the atmospheric boundary layer is far better measured than the ocean, and there are certain features that are similar. However, before applying the Monin-Obukhov similarity theory to the ocean, we acknowledge some characteristics of the ocean surface boundary layer that distinguish it from atmospheric boundary layers.

- Surface ocean gravity waves can impact a nontrivial fraction of the ocean surface boundary layer, whereas such waves only impact a small fraction of atmospheric boundary layers.
- The surface ocean velocity is generally the largest velocity in the ocean. In contrast, the surface atmospheric velocity vanishes over land and is relatively small over the ocean.
- The surface ocean absorbs shortwave solar radiation, whereas the atmosphere is nearly transparent to radiation.

Despite these basic distinctions between planetary boundary layers in the atmosphere and ocean, [Large et al. \(1994\)](#) used the Monin-Obukhov similarity theory to introduce scales for turbulent fluctuations and to identify non-dimensional similarity functions in the ocean surface layer.

18.4.1 The surface layer

A molecular layer exists within roughly a millimetre of the upper ocean interface, with this layer dominated by molecular viscous and diffusive effects (Large, 1998). Since it is dominated by molecular viscous effects, this layer is not turbulent and thus leads to negligible mixing of tracer and momentum. It is the molecular layer that ultimately transfers properties between the ocean and atmosphere or ice, including momentum and buoyancy. The more this layer is “corrugated” through wave breaking and other turbulent action, the faster properties are transferred across the surface ocean interface.

The ocean *surface layer* (Figure 18.1) is a turbulent layer whose turbulent fluxes are roughly independent of distance from the upper boundary; i.e., the surface layer is nearly a *constant flux* layer. The surface layer starts just beneath the molecular viscous layer. Turbulence within the surface layer delivers properties to the molecular layer for transfer to the atmosphere or ice. Given that no ocean model resolves the molecular sublayer, the upper ocean interface at $z = \eta(x, y, t)$ in an ocean model operationally starts at the top of the surface layer.

18.4.2 Monin-Obukhov similarity theory

The surface turbulent layer is of fundamental importance for determining the rate that properties are transferred across the surface ocean interface. It thus plays a key role in how the ocean is forced. If we needed to model all the details of this layer, then the problem of coupled modeling would perhaps be intractable. Fortunately, the Monin-Obukhov similarity theory has proven to be quite useful in many contexts, particularly for the atmosphere boundary layer. Following Large et al. (1994), we consider its use for the ocean surface boundary layer.

Monin-Obukhov similarity theory assumes that the turbulent surface layer is a constant flux layer that starts just beneath any roughness elements, and certainly beneath the the molecular sublayer. In the absence of breaking surface waves, roughness elements arise from capillary waves that allow the wind to affect the otherwise smooth ocean surface, in which case the roughness length is on the order of centimetres. With breaking surface waves, the roughness length can increase to the order of a metre (e.g., see concluding section to Craig and Banner, 1994). Furthermore, the scalings from Monin-Obukhov are distinctly not correct with surface wave breaking (e.g., Craig and Banner, 1994; Terray et al., 1996). Surface gravity waves are ignored in the formulation of Large et al. (1994).

Even if the surface layer is not a constant flux layer, the following scalings are relevant so long as the surface fluxes remain the dominant parameters determining properties of this layer (Tennekes, 1973). Within the surface layer, the relevant dimensional quantities are the distance d from the surface interface at $z = \eta$, and the surface kinematic fluxes of momentum, tracer, scalars, and buoyancy

$$\overline{w\mathbf{u}}^\eta = \text{surface kinematic momentum flux} \quad (18.52)$$

$$\rho_o C_p \overline{w\theta}^\eta = \text{surface kinematic heat flux} \quad (18.53)$$

$$\overline{ws}^\eta = \text{surface kinematic scalar (e.g., salt) flux} \quad (18.54)$$

$$\overline{wb}^\eta = \text{surface kinematic buoyancy flux.} \quad (18.55)$$

We now introduce the following dimensional scales.

- **FRICTION VELOCITY:** From the surface kinematic momentum flux, we introduce the turbulent velocity scale, also known as the *friction velocity* scale

$$u_*^2 \equiv |\overline{w\mathbf{u}}^\eta|. \quad (18.56)$$

Use of the identity (18.18) provides a means to compute the surface friction velocity given the surface momentum stress

$$\rho_o u_*^2 = |\tau|. \quad (18.57)$$

- **TEMPERATURE SCALE:** From the surface kinematic heat flux and the surface kinematic momentum flux, we define a scale for the surface turbulent temperature fluctuations

$$\Theta_* = - \left(\frac{\overline{w\theta}^\eta}{\sqrt{|\overline{w\mathbf{u}}^\eta|}} \right) = - \left(\frac{\overline{w\theta}^\eta}{u_*} \right). \quad (18.58)$$

The sign is chosen so that turbulent fluxes leading to surface ocean cooling, $\overline{w\theta'} > 0$, correspond to a negative turbulent temperature scale, $\Theta_* < 0$.

- **SCALAR SCALE:** From the surface kinematic scalar flux and the surface kinematic momentum flux, we define a scale for the surface turbulent scalar fluctuations

$$S_* = -\left(\frac{\overline{ws'}}{u_*}\right). \quad (18.59)$$

- **BOUANCY SCALE:** From the surface kinematic buoyancy flux $-\overline{wb'}$ (equation (18.49)), and the penetrative buoyancy flux B_R (equation (18.50)), we define a scale for the surface turbulent buoyancy fluctuations

$$B_* = \left(\frac{B_f}{u_*}\right) = \left(\frac{-\overline{wb'} + B_R}{u_*}\right). \quad (18.60)$$

18.4.3 Similarity functions and length scale

The Monin-Obukhov similarity theory assumes the vertical gradient of any mean field, Λ , within the surface turbulent layer is a function of the scale Λ_* of its turbulent fluctuations, the buoyancy scale B_* , the velocity scale u_* , and the vertical distance from the upper interface, $d = -z + \eta$ (equation (18.10)). In this case, we write

$$\frac{\partial \Lambda}{\partial z} = \Psi(d, u_*, B_*, \Lambda_*), \quad (18.61)$$

where Ψ is an unknown function. Although no exact analytical expression exists for Ψ , Monin-Obukhov theory suggests that progress can be made by fitting data to the following form

$$\frac{\partial \Lambda}{\partial z} = \left(\frac{\Lambda_*}{\kappa d}\right) \phi_\Lambda(\zeta). \quad (18.62)$$

In this expression,

$$\kappa \approx 0.4 \quad (18.63)$$

is the von Karman constant, $\phi_\Lambda(\zeta)$ is a dimensionless *similarity function* or flux profile that is dependent only on the scaled distance

$$\zeta \equiv \frac{d}{L}, \quad (18.64)$$

and

$$L = \frac{u_*^2}{\kappa B_*} = \frac{u_*^3}{\kappa B_f} = \frac{|\tau/\rho_0|^{3/2}}{\kappa B_f} \quad (18.65)$$

is the Monin-Obukhov length scale determined by the ratio of the momentum forcing to buoyancy forcing.

The Monin-Obukhov length scale takes on the following values for the suite of available boundary forcing

$$L = \begin{cases} 0 & u_* = 0, B_* \neq 0 & \tau = 0, B_f \neq 0 & \text{zero winds} \\ \infty & u_* \neq 0, B_* = 0 & \tau \neq 0, B_f = 0 & \text{zero buoyancy forcing (neutral forcing)} \\ > 0 & u_* \neq 0, B_* > 0 & \tau \neq 0, B_f > 0 & \text{stabilizing buoyancy forcing} \\ < 0 & u_* \neq 0, B_* < 0 & \tau \neq 0, B_f < 0 & \text{destabilizing or convective buoyancy forcing.} \end{cases} \quad (18.66)$$

Notably, L is *not* the finite positive thickness of the surface turbulent layer (Figure 18.1), as evident since L can be negative or infinite. Instead, L is the depth scale at which buoyancy production of turbulent kinetic energy is of the same magnitude as shear production. For depths shallower than $L > 0$, shear production dominates due to the effects from mechanical forcing through momentum stress τ . The case $L = \infty$ is trivially dominated by shear production since there is no buoyancy forcing. For depths deeper than L , buoyancy production dominates the turbulence. The case of $L < 0$ (convection) is always dominated by buoyancy production.

The similarity function ϕ_Λ appearing in equation (18.62) satisfies the following limit case under neutral forcing (zero buoyancy forcing)

$$\phi_\Lambda(0) = 1 \quad \text{arising from } B_f = 0 \text{ so that } L = \infty \text{ and } \zeta = d/L = 0. \quad (18.67)$$

This limit reduces the more general Monin-Obukhov form for the vertical derivative (18.62) to the logarithmic Law of the Wall form

$$\frac{\partial \Lambda}{\partial z} = \left(\frac{\Lambda_*}{\kappa d} \right) \quad \text{neutral forcing so } \phi_\Lambda = 1. \quad (18.68)$$

In the general case of nonzero buoyancy forcing, we integrate the similarity form (18.62) to expose the logarithmic Law of the Wall for neutral forcing, plus a term present with nonzero buoyancy forcing. For this purpose, rewrite equation (18.62) in terms of the scaled Monin-Obukhov distance, ζ , to have

$$\frac{\partial \Lambda}{\partial \zeta} = - \left(\frac{\Lambda_*}{\kappa \zeta} \right) \phi_\Lambda(\zeta), \quad (18.69)$$

where we used the relation between vertical increments through

$$d\zeta = -L dz \quad (18.70)$$

using $d = -z + \eta$ (equation (18.10)). We now vertically integrate equation (18.69) to have

$$\Lambda(\zeta) = \Lambda(Z_\lambda/L) + \left(\frac{\Lambda_*}{L} \right) \int_{Z_\lambda/L}^{\zeta} \left(\frac{(1 - \phi_\Lambda) - 1}{\zeta'} \right) d\zeta'. \quad (18.71)$$

In this expression,

$$Z_\lambda = \text{roughness length} \quad (18.72)$$

introduced the roughness length associated with each fluctuating field. Within a distance Z_λ or less from the boundary at $z = \eta$, the kinematic fluxes are not expected to be constant due to the impacts from roughness elements. Hence, we expect the Monin-Obukhov similarity theory to breakdown when getting closer than the roughness length to the surface.

Integrating the right hand side of equation (18.71) from the roughness length to an arbitrary point within the surface layer renders¹

$$\Lambda(\zeta) = \Lambda(Z_\lambda/L) - \left(\frac{\Lambda_*}{L} \right) \ln(\zeta L/Z_\lambda) + \left(\frac{\Lambda_*}{L} \right) \int_{Z_\lambda/L}^{\zeta} \left(\frac{(1 - \phi_\Lambda)}{\zeta'} \right) d\zeta'. \quad (18.73)$$

As expected, the first term exposes the logarithmic Law of the Wall behaviour occurring for neutral forcing conditions ($\phi_\Lambda = 1$). Deviations from Law of the Wall for non-neutral forcing are embodied in the integral on the right hand side. Recall that values $\zeta < Z_\lambda/L$ are within the roughness elements or molecular sublayer, so the theory cannot be applied there.

Large et al. (1994) (see their page 365) use atmospheric boundary layer results from Tennekes (1973) to set the surface layer thickness to (see Figure 18.1)

$$\epsilon = 0.1 \quad \text{fraction of KPP boundary layer occupied by surface layer.} \quad (18.74)$$

Within the surface layer, atmospheric boundary layer studies indicate that turbulent fluxes are within 20% of their surface values when reaching a distance $d = \epsilon h$ from the upper ocean interface at $d = 0$. The value of $\epsilon = 0.1$ has never been observed in the ocean, but there is no reason to believe it is fundamentally incorrect. Hence, this is the value taken for the KPP scheme.

¹The result (18.73) disagrees with equation (4) in Large et al. (1994) by a minus sign, with the origin of the minus sign the relation (18.70) between infinitesimal changes in ζ and infinitesimal changes in z .

18.5 Specifying the KPP parameterization

We are now ready to determine the KPP boundary layer depth, h , the diffusivity, K_λ , and non-local transport, γ_λ , thus enabling a full parameterization of the turbulent flux $\overline{w\lambda}$ according to

$$\overline{w\lambda} = -K_\lambda \left(\frac{\partial \Lambda}{\partial z} - \gamma_\lambda \right), \quad (18.75)$$

where the diffusivity is given by equation (18.7), rewritten here as

$$K_\lambda(\sigma) = h w_\lambda(\sigma) G_\lambda(\sigma). \quad (18.76)$$

Recall that

$$\sigma = d/h \quad (18.77)$$

is the dimensionless distance from the upper surface normalized by the boundary layer thickness, with

$$d = -z + \eta \quad (18.78)$$

the dimensionful distance.

18.5.1 The turbulent vertical velocity scale w_λ

We now determine the turbulent vertical velocity scale w_λ appearing in equation (18.76).

18.5.1.1 Velocity scale with stable buoyancy forcing

Following page 370 of Large et al. (1994), we first specify the velocity scale within the Monin-Obukhov surface layer, where $\sigma = d/h < \epsilon = 0.1$. We also assume stable buoyancy forcing, so that the non-local term, γ_λ , vanishes. We later extend these results to the full boundary layer for arbitrary buoyancy forcing.

The similarity result (18.62) holds in the surface layer, in which

$$\frac{\partial \Lambda}{\partial z} = \left(\frac{\Lambda_*}{\kappa d} \right) \phi_\Lambda(\zeta). \quad (18.79)$$

We may eliminate the vertical gradient $\partial \Lambda / \partial z$ using the KPP parameterization (18.75) with a zero non-local term under stable buoyancy forcing

$$\phi_\Lambda = -\frac{\kappa d}{\Lambda_*} \left(\frac{\overline{w\lambda}}{K_\lambda} \right). \quad (18.80)$$

Substituting the turbulent scale $\Lambda_* = -\overline{w\lambda}^\eta / u_*$ from equation (18.59) yields

$$K_\lambda \phi_\Lambda = \kappa d u_* \left(\frac{\overline{w\lambda}}{\overline{w\lambda}^\eta} \right). \quad (18.81)$$

The KPP diffusivity expression (18.76) then renders

$$w_\lambda(\sigma) \sigma^{-1} G_\lambda(\sigma) = \left(\frac{\kappa u_*}{\phi_\Lambda(\sigma)} \right) \left(\frac{\overline{w\lambda}^\sigma}{\overline{w\lambda}^\eta} \right). \quad (18.82)$$

Recalling that $\sigma < \epsilon = 0.1$ in the surface layer yields the approximate linear relation

$$\sigma^{-1} G_\lambda(\sigma) \approx a_1 + a_2 \sigma, \quad (18.83)$$

where we used expression (18.13) for the structure function $G_\lambda(\sigma)$. Furthermore, within the surface layer, turbulent fluxes for any fluctuating field, $\overline{w\lambda}^\sigma$, are linearly proportional to their surface value, $\overline{w\lambda}^\eta$. We may thus use this result to specify a part of the structure function according to

$$a_1 + a_2 \sigma = \left(\frac{\overline{w\lambda}^\sigma}{\overline{w\lambda}^\eta} \right). \quad (18.84)$$

Note that as shown in Section 18.5.3, there is generally a dependence of a_2 on the field λ , whereas a_1 is unity for all fields. With the specification (18.84), we are led to an expression for the turbulent velocity scale within the surface layer

$$w_\lambda(\sigma) = \frac{\kappa u_*}{\phi_\Lambda(\sigma h/L)} \quad \text{for stable forcing } B_f > 0 \text{ and } 0 < \sigma < \epsilon. \quad (18.85)$$

Troen and Mahrt (1986) assume this expression is valid throughout the stably forced boundary layer for $0 < \sigma < 1$, and Large et al. (1994) also make that assumption.

18.5.1.2 Velocity scale with unstable buoyancy forcing

For unstable buoyancy forcing conditions, $B_f < 0$, the turbulent velocity scales within the surface layer are assumed to be the same as the stable velocity scale (18.85), again within the surface layer. For unstable forcing beneath the surface layer, $\epsilon < \sigma < 1$, Large et al. (1994) cap the velocity scale to that evaluated at the base of the surface layer at $\sigma = \epsilon$.

18.5.1.3 Summarizing properties of the turbulent velocity scale

The net result for all conditions is that the turbulent vertical velocity scale is given by

$$w_\lambda(\sigma) = \kappa u_* \begin{cases} \phi_\Lambda^{-1}(\sigma h/L) & \text{stable forcing } B_f > 0 & \text{OBL} & 0 < \sigma < 1 \\ \phi_\Lambda^{-1}(\sigma h/L) & \text{unstable forcing } B_f < 0 & \text{surface layer} & \sigma < \epsilon \\ \phi_\Lambda^{-1}(\epsilon h/L) & \text{unstable forcing } B_f < 0 & \text{OBL beneath surface layer} & \epsilon < \sigma < 1. \end{cases} \quad (18.86)$$

We now summarize various properties of the velocity scale, with these properties reflected in Figure 18.3.

- **STABLE FORCING:** The similarity functions ϕ_Λ and velocity scales w_λ satisfy the following properties under positive buoyancy forcing, $B_f > 0$.
 - The similarity functions are increased so that the turbulent velocity scales are reduced.
 - The similarity functions are the same for all scalars and momentum, so that the velocity scales w_λ are the same.
- **NEUTRAL FORCING:** with zero buoyancy forcing, $B_f = 0$, the similarity functions satisfy $\phi_\Lambda = 1$, so that $w_\lambda(\sigma) = \kappa u_*$.
- **UNSTABLE FORCING:** The similarity functions ϕ_Λ and velocity scales w_λ satisfy the following properties under negative buoyancy forcing, $B_f < 0$.
 - The similarity functions ϕ_Λ are reduced so that the turbulent velocity scales w_λ are enhanced.
 - The similarity functions for momentum are larger than those for scalars, so that the velocity scales for momentum are smaller than for scalars: $w_m < w_s$.
 - In the convective limit, for which $u_* \rightarrow 0$, the velocity scales behave according to

$$w_\lambda \sim w_* = (-B_f h)^{1/3}. \quad (18.87)$$

In order to satisfy this scaling, the similarity functions ϕ_Λ must have the form

$$\phi_\Lambda = (a_\lambda - c_\lambda \zeta)^{-1/3} \quad \text{convective conditions with } u_* \rightarrow 0, \quad (18.88)$$

where $\zeta = d/L \ll 0$, and the constants a_λ and c_λ are chosen to match the convective form (18.88) to less unstable forms.

We now use the expression (18.88) within the unstable surface layer ($\sigma < \epsilon$) form in (18.86) to render

$$\begin{aligned}
 w_\lambda &= \kappa (a_\lambda u_*^3 - c_\lambda u_*^3 \zeta)^{1/3} \\
 &= \kappa [a_\lambda u_*^3 - c_\lambda u_*^3 (h\sigma/L)]^{1/3} \\
 &= \kappa (a_\lambda u_*^3 - c_\lambda \sigma \kappa h B_f)^{1/3} \\
 &= \kappa (a_\lambda u_*^3 + c_\lambda \sigma \kappa w_*^3)^{1/3} \\
 &\rightarrow \kappa w_* (c_\lambda \sigma \kappa)^{1/3},
 \end{aligned} \tag{18.89}$$

where the final limit case is for the convective limit with $u_* \rightarrow 0$. Likewise, outside the surface layer ($\epsilon < \sigma < 1$) we have

$$w_\lambda = \kappa (a_\lambda u_*^3 + c_\lambda \epsilon \kappa w_*^3)^{1/3} \rightarrow \kappa w_* (c_\lambda \epsilon \kappa)^{1/3}, \tag{18.90}$$

where again the final limit case is for the convective limit with $u_* \rightarrow 0$.

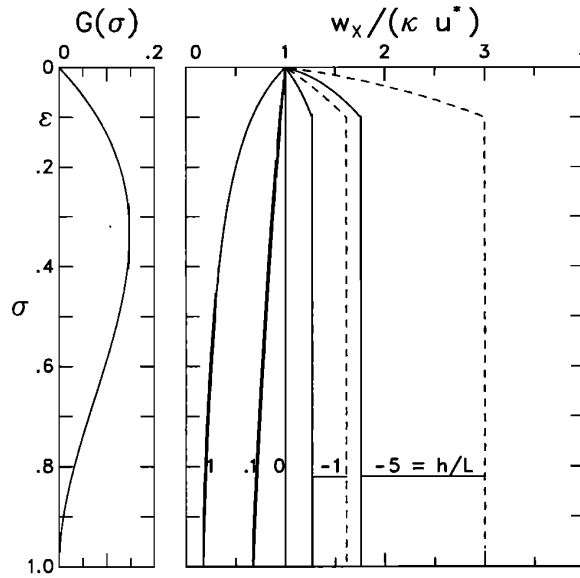


Figure 18.3: This is a reproduction of Figure 2 from [Large et al. \(1994\)](#). The vertical axis is the dimensionless vertical coordinate $\sigma = d/h$ within the KPP boundary layer $0 \leq \sigma \leq 1$. The left panel shows the vertical profile of the shape or structure function, $G_\lambda(\sigma)$, used to scale the vertical diffusivity via equation (18.76). The analytic form shown here is given by $G_\lambda(\sigma) = \sigma(1 - \sigma)^2$, which corresponds to the [Troen and Mahrt \(1986\)](#) form and which is independent of the quantity Λ being diffused. [Large et al. \(1994\)](#) chose a more general form, based on the need to match boundary layer diffusivities to interior diffusivities in which case the shape function becomes a function of λ . We detail this approach in Section 18.5.3. The right panel shows various examples of the normalized turbulent velocity scale w_λ (called w_x in [Large et al. \(1994\)](#)), with the examples differing by the value of the dimensionless ratio h/L between the boundary layer depth, h , and the Monin-Obukhov length scale L . For unstable buoyancy forcing, $L < 0$, the velocity scale for scalars, w_s (dashed lines), is greater than that for momentum, w_m (solid lines). For stable forcing, $L > 0$, and both scalar and momentum have the same turbulent velocity scales, $w_s = w_m$. In general, the turbulent velocity scale is enhanced with unstable surface buoyancy forcing, and reduced with stable buoyancy forcing.

18.5.2 Similarity functions ϕ_Λ

The vertical velocity scales are functions of the similarity functions ϕ_Λ , also called the dimensionless flux profiles. Appendix B of [Large et al. \(1994\)](#) present analytic forms for these functions, based on fits to available data, with their Figure B1 (reproduced here as Figure 18.4) providing a summary of the choices for the momentum function ϕ_m and the scalar function ϕ_s . Both functions agree for stable buoyancy forcing, and they depend linearly on the dimensionless Monin-Obukhov length $\zeta = d/L = \sigma h/L$.

18.5.2.1 The [Large et al. \(1994\)](#) choices for unstable buoyancy forcing

For unstable buoyancy forcing, where $L < 0$ and so $\zeta < 0$, there are two regimes. The scalar function ϕ_s is always less than the momentum function ϕ_m . Hence, for unstable forcing there is a larger turbulent velocity scale for the scalars than momentum, and thus a larger vertical diffusivity for scalars. The turbulent Prandtl number, Pr , is given by the ratio of the flux functions

$$Pr = K_m/K_s = w_m/w_s = \phi_m/\phi_s. \quad (18.91)$$

The choices made by [Large et al. \(1994\)](#) lead to a Prandtl number in the convective limit ($\zeta \rightarrow -\infty$) of

$$Pr \rightarrow (c_m/c_s)^{1/3} = 0.44, \quad (18.92)$$

where c_m and c_s are parameters in the similarity functions ϕ_m and ϕ_s , respectively.

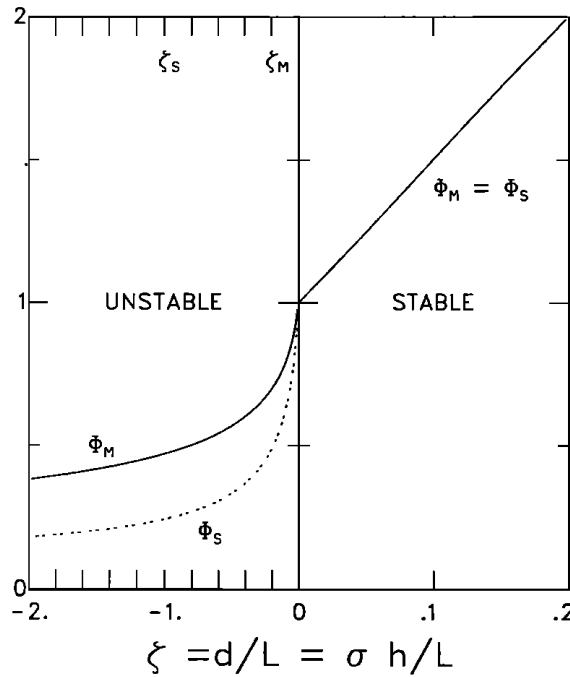


Figure 18.4: This is a reproduction of Figure B1 from [Large et al. \(1994\)](#). The vertical axis provides values for the dimensionless flux profiles, ϕ_Λ , for momentum and scalars, and the horizontal axis gives the dimensionless Monin-Obukhov length scale $\zeta = d/L = \sigma h/L$. There is a transition across the neutrally forced value of $\zeta = 0$. For stable buoyancy forcing ($\zeta > 0$), both functions are the same, $\phi_s = \phi_m$, and are linear functions of ζ . For unstable buoyancy forcing ($\zeta < 0$), the scalar function is less than momentum, $\phi_s < \phi_m$, with both functions falling off with a negative fractional power. The analytic forms for the functions are given by equations (B1) and (B2) in [Large et al. \(1994\)](#).

18.5.2.2 Alternative choices for unstable buoyancy forcing

Large et al. (1994) chose two regimes for the unstable buoyancy forced range, transitioning from different fractional exponents near $\zeta = 0$, to the same $-1/3$ power for larger negative ζ . The scalar function ϕ_s falls off faster near $\zeta = 0$, with a power $-1/2$, whereas the momentum function ϕ_m falls off with a $-1/4$ power. This initial distinct fractional power falloff sets the scale for the Prandtl number in this portion of ζ in the weakly unstable regime.

Having two regimes for the negative buoyancy forcing adds complexity to the algorithm. We thus consider how well the original two-regime forms for ϕ_m and ϕ_s can be fit using a single regime, using only the fractional power $-1/3$. Tests suggest that the following forms may be suitable

$$\phi_m(\zeta) = \begin{cases} 1 + 5\zeta & \zeta > 0 \\ (1 - 9\zeta)^{-1/3} & \zeta < 0 \end{cases} \quad (18.93)$$

$$\phi_s(\zeta) = \begin{cases} 1 + 5\zeta & \zeta > 0 \\ (1 - 60\zeta)^{-1/3} & \zeta < 0 \end{cases} \quad (18.94)$$

A comparison of the original forms from Large et al. (1994) to the alternative forms is shown in Figure 18.5. Also shown is the ratio of these two functions which yields the turbulent Prandtl number according to equation (18.91). The agreement between the original forms and the new forms is worse when considering the Prandtl number.

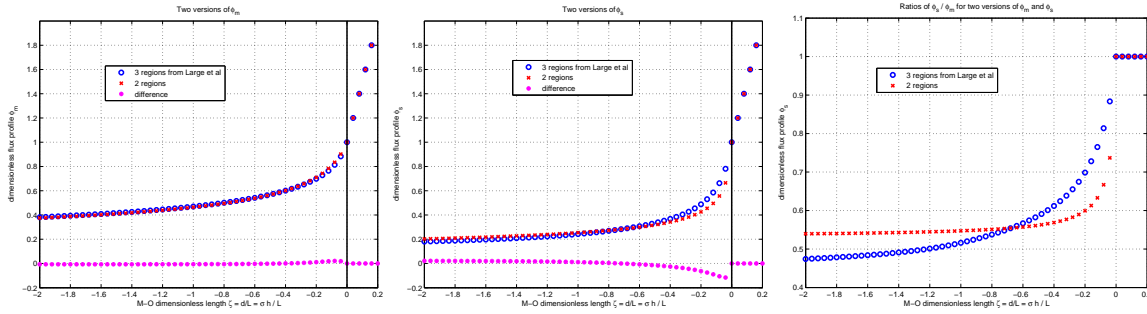


Figure 18.5: Shown here are alternative flux profiles given by equations (18.93) and (18.94), as well as their ratio ϕ_s/ϕ_m , with this ratio defining the turbulent Prandtl number, or the ratio of the vertical momentum viscosity to vertical tracer diffusivity.

18.5.3 The shape function $G_\lambda(\sigma)$

The vertical shape function $G_\lambda(\sigma)$ is given by the cubic polynomial

$$G_\lambda(\sigma) = a_0 + a_1 \sigma + a_2 \sigma^2 + a_3 \sigma^3. \quad (18.95)$$

As already noted when introducing this cubic expression (equation (18.13)), turbulent eddies do not cross the ocean surface at $\sigma = 0$, so the diffusivity should vanish at $\sigma = 0$. This constraint is satisfied by setting

$$a_0 = 0. \quad (18.96)$$

We now discuss further constraints to specify the remaining coefficients.

We start by rewriting the expression (18.84) that expresses the ratio of turbulent fluxes within the surface layer to those at the surface boundary

$$a_1 + a_2 \sigma = \left(\frac{\overline{w \lambda}^\sigma}{\overline{w \lambda}^\eta} \right) \quad \text{surface layer: } 0 \leq \sigma \leq \epsilon. \quad (18.97)$$

Satisfying this relation at the ocean surface, $\sigma = 0$, requires

$$a_1 = 1, \quad (18.98)$$

so that

$$1 + a_2 \sigma = \left(\frac{\overline{w \lambda}^\sigma}{\overline{w \lambda}^\eta} \right) \quad \text{surface layer: } 0 \leq \sigma \leq \epsilon. \quad (18.99)$$

Now define the ratio

$$\beta_\lambda = \left(\frac{\overline{w \lambda}^\epsilon}{\overline{w \lambda}^\eta} \right), \quad (18.100)$$

which is the ratio of the turbulent flux at the base of the surface layer, $\sigma = \epsilon$, to the flux at the upper ocean interface, $z = \eta$. For atmospheric boundary layers, [Troen and Mahrt \(1986\)](#) set

$$\beta_\lambda = 2\epsilon \quad \text{atmospheric boundary layers,} \quad (18.101)$$

with $\epsilon = 0.1$. [Troen and Mahrt \(1986\)](#) further assume both the shape function and its first derivative vanish at the base of the boundary layer, $\sigma = 1$. These assumptions lead to the cubic expression valid for all fluctuating fields λ

$$G(\sigma) = \sigma(1 - \sigma)^2 \quad \text{atmospheric boundary layers,} \quad (18.102)$$

with this function exhibited in the left panel of Figure 18.3.

[Large et al. \(1994\)](#) also assume the surface layer is 10% of the boundary layer, so that

$$\epsilon = 0.1 \quad \text{KPP scheme.} \quad (18.103)$$

However, they consider a more general approach for the remaining approach to deriving the shape function. The key reason to generalize the atmospheric approach of [Troen and Mahrt \(1986\)](#) is to admit the possibility of ocean boundary layer turbulence to be impacted by interior mixing, with this mixing parameterized by downgradient vertical diffusion. Such diffusion generally introduces distinct diffusivities for tracers (e.g., double diffusion) as well as for momentum (e.g., non-unit Prandtl number). For these reasons, [Large et al. \(1994\)](#) insist that both the diffusivity and its vertical derivative match across the base of the boundary layer at $\sigma = 1$. This matching condition leads to the constraints (18) given by [Large et al. \(1994\)](#), which in turn leads to shape functions that are dependent on the field being transported.

Matching both the shape function and its vertical derivative across the boundary layer base adds complexity to the KPP algorithm. Furthermore, it is unclear how accurate one can in fact satisfy both matching conditions on a finite grid with potentially coarse vertical grid spacing at the boundary layer base. To simplify the KPP algorithm, we drop the need to match the vertical derivative of the diffusivity. Instead, we assume continuity of the diffusivity with a vanishing derivative at the boundary layer base, $\sigma = 1$. Setting $\partial_\sigma G(\sigma) = 0$ at $\sigma = 1$ leads to the relation

$$3a_3 = -(1 + 2a_2). \quad (18.104)$$

Matching diffusivities at $\sigma = 1$ between the boundary layer and interior value leads to

$$a_2 = -2 + \left(\frac{3K_\lambda(h)}{hw_\lambda(h)} \right), \quad (18.105)$$

where the diffusivity $K_\lambda(h)$ is determined by parameterizations of interior mixing. Substituting this expression for a_2 into equation (18.104) for a_3 leads to

$$a_3 = 1 - \left(\frac{2K_\lambda(h)}{hw_\lambda(h)} \right). \quad (18.106)$$

Allowing for the interior mixing to influence the KPP boundary layer scheme suggests that the KPP calculation should be called *after* the various methods used to compute interior diffusivities.

18.5.4 The non-local transport γ_λ

We now consider the parameterization for the non-local transport (see Section 18.1.4) as suggested by Large et al. (1994). Again, the KPP parameterization takes the form (equation (18.3))

$$\overline{w\lambda} = -K_\lambda \left(\frac{\partial \Lambda}{\partial z} - \gamma_\lambda \right), \quad (18.107)$$

so that that non-local portion of the turbulent flux is parameterized according to

$$\overline{w\lambda}^{\text{non-local}} = K_\lambda \gamma_\lambda, \quad (18.108)$$

where K_λ takes the form in equation (18.76):

$$K_\lambda(\sigma) = h w_\lambda(\sigma) G_\lambda(\sigma). \quad (18.109)$$

For completeness, we repeat elements of the outline presented in Section 18.1.4.

18.5.4.1 General features of γ_λ with the KPP parameterization

- Smyth et al. (2002) consider a non-local term for momentum. Until their ideas have been fully tested in climate models, we follow recommendations from (Large et al., 1994), who set the non-local momentum transport to zero:

$$\gamma_\lambda = \begin{cases} 0 & \text{if } \lambda = (u, v, w) \text{ a velocity component} \\ \neq 0 & \text{nonzero if } \lambda = \theta, s \text{ or another tracer.} \end{cases} \quad (18.110)$$

- The non-local transport is non-zero only within the OBL:

$$\gamma_\lambda = \begin{cases} 0 & \text{if } \sigma > 1 \\ \neq 0 & \text{if } 0 \leq \sigma \leq 1. \end{cases} \quad (18.111)$$

- The non-local transport is non-zero only in the presence of destabilizing negative surface ocean buoyancy flux:

$$\gamma_\lambda = \begin{cases} 0 & \text{for } B_f > 0 \\ \neq 0 & \text{for } B_f < 0. \end{cases} \quad (18.112)$$

- The non-local transport for temperature and arbitrary scalars is given by the following form for destabilizing negative surface ocean buoyancy fluxes:

$$\gamma_\theta = C_s \left(\frac{\overline{w\theta}^\eta - Q_R/(\rho_0 C_p)}{h w_\theta(\sigma)} \right) \quad (18.113)$$

$$\gamma_s = C_s \left(\frac{\overline{ws}^\eta}{h w_s(\sigma)} \right), \quad (18.114)$$

where

$$C_s = C_* \kappa (c_s \kappa \epsilon)^{1/3}, \quad (18.115)$$

with

$$C_* = 10, \quad (18.116)$$

and Q_R is the heat flux from penetrative radiation given by equation (18.51).

Combining the parameterizations (18.113) and (18.114) for the non-local term γ_λ , with that for the vertical diffusivity K_λ in equation (18.109) renders the non-local flux parameterization in the form

$$\overline{w\theta}^{\text{non-local}} = K_\theta \gamma_\theta = G_\lambda(\sigma) C_s \left(\overline{w\theta}^\eta - Q_R/(\rho_0 C_p) \right) \quad (18.117)$$

$$\overline{ws}^{\text{non-local}} = K_s \gamma_s = G_s(\sigma) C_s \left(\overline{ws}^\eta \right). \quad (18.118)$$

Notice how explicit dependence on both the turbulent velocity scale, w_λ , and boundary layer depth, h , drop out from the parameterization of the non-local flux.

18.5.4.2 Potential problems with the parameterized non-local transport

Experience has shown that there are cases when the parameterized non-local flux, (18.117) of (18.118), can produce values larger than the surface flux. That is, one may realize cases when

$$G_\lambda(\sigma)C_s > 1 \quad \text{non-local flux greater than surface flux.} \quad (18.119)$$

This situation arises particularly near the boundary layer base, $\sigma = 1$, when the interior diffusivity is large. The matching conditions employed by Large et al. (1994) (Section 18.5.3) then lead to a very large value for the shape function $G(\sigma)$. In this case, one may be exposed to the production of extrema in the tracer field. In the presence of sea-ice, problems may arise particularly in fresh water regions such as the Baltic Sea where the thermal expansion coefficient is negative, $\alpha < 0$ (Martin Schmidt, personal communication).

The following modifications to the original Large et al. (1994) scheme have been found useful to reduce the potential for the non-local term to be problematic.

- **INTERIOR GRAVITATIONAL INSTABILITIES:** When the vertical stratification is unstable ($N^2 < 0$), vertical diffusivity is enhanced to remove the gravitational instability. Notably, it is *not* appropriate to enhance the diffusivity within the KPP boundary layer, beyond that already computed via the KPP scheme, even when $N^2 < 0$. On those occasions when the instabilities appear beneath the boundary layer, diffusivities are enhanced. If one insisted that such diffusivities should match those in the boundary layer, then the shape function $G(\sigma)$ would indeed become quite large in magnitude. Hence, NCAR recommends that one pull the “convective adjustment” portion of the mixing scheme outside of the KPP portion of the algorithm. That is, the interior convective instability diffusivities should *not* be matched to the KPP boundary layer diffusivities.
- **SIMPLER MATCHING:** As noted in Section 18.5.3, we propose to simplify the matching at the boundary layer base, so that only the diffusivities match across the boundary layer base, rather than also insisting on the derivative of the diffusivities as proposed by Large et al. (1994). The simplified matching condition leads to less problems computing discrete vertical derivatives of the diffusivities, and in turn produces more well regularized diffusivities and shape functions.

18.5.5 The bulk Richardson number and the OBL thickness h

Large et al. (1994) define the KPP boundary layer depth to be the first depth at which the bulk Richardson number, Ri_b , equals to a critical Richardson number, Ri_c . The bulk Richardson number is computed using bulk averaged buoyancy, B_r , and horizontal velocity, \mathbf{U}_r , over the surface layer, $0 \leq \sigma \leq \epsilon$, so that

$$Ri_b(d) = \frac{d[B_r - B(d)]}{|\mathbf{U}_r - \mathbf{U}(d)|^2 + U_t^2}. \quad (18.120)$$

Recall the buoyancy was defined by equation (18.19) as

$$B = g \left(\frac{\rho_0 - \rho}{\rho_0} \right), \quad (18.121)$$

where ρ is the *in-situ* density and ρ_0 is a constant reference density. In the denominator of the bulk Richardson number (18.120), Large et al. (1994) add the term U_t^2 , which is associated with parameterized unresolved vertical shears that may act to further reduce the bulk Richardson number.

The physics underlying the definition (18.120) is that boundary layer eddies with a surface layer buoyancy B_r and velocity \mathbf{U}_r may penetrate to the full boundary layer h , at which point their turbulence is suppressed by the reduced shear and increased buoyancy stratification. We now comment in turn on both the specification of the buoyancy in the numerator, and the unresolved shear in the denominator.

18.5.5.1 Local gravitational stability

The flux Richardson number is defined by

$$Ri_f = \frac{N^2}{|\partial_z \mathbf{U}|^2}, \quad (18.122)$$

where

$$N^2 = g \left(\alpha \frac{\partial \Theta}{\partial z} - \beta \frac{\partial S}{\partial z} \right) \quad (18.123)$$

is the buoyancy frequency. The denominator of the bulk Richardson number defined by equation (18.120) represents a finite difference of the velocity averaged over the surface layer and that at a distance d beneath the ocean surface. The numerator is also reminiscent of a finite difference representation of the buoyancy frequency, and so may represent a finite difference measure of gravitational stability. In the following, we consider how this measure of gravitational stability for a fluid column arises from the more traditional local methods in terms of local buoyancy frequency appearing in the flux Richardson number. In so doing, we provide an equivalence between the method proposed by Large et al. (1994), and another method based on vertically integrating the local buoyancy frequency. The alternative method may prove to be more useful when kernalizing the KPP scheme so to remove the need to recompute the equation of state when determining the boundary layer thickness.

To initiate the discussion, we start with some standard material on gravitational stability. Following McDougall (1987a), introduce the notion of a neutrally stable direction by considering an infinitesimal displacement $d\mathbf{x}$ of a fluid parcel. Under a general displacement, the *in situ* density is given to leading order by

$$\rho(\mathbf{x} + d\mathbf{x}) = \rho(\mathbf{x}) + d\rho(\mathbf{x}), \quad (18.124)$$

where

$$d\rho(\mathbf{x}) = \rho d\mathbf{x} \cdot \left(-\alpha \nabla \Theta + \beta \nabla S + \frac{1}{\rho c_{\text{sound}}^2} \nabla p \right), \quad (18.125)$$

with the sound speed defined by

$$c_{\text{sound}}^2 = \left(\frac{\partial p}{\partial \rho} \right)_{S, \Theta}, \quad (18.126)$$

The ambient density, $\rho(\mathbf{x} + d\mathbf{x})$, at the new point thus differs from density at the original point $\rho(\mathbf{x})$ by an amount $d\rho$ according to

$$\rho(\mathbf{x} + d\mathbf{x}) = \rho(\mathbf{x}) + \rho d\mathbf{x} \cdot \left(-\alpha \nabla \Theta + \beta \nabla S + \frac{1}{\rho c_{\text{sound}}^2} \nabla p \right). \quad (18.127)$$

Now instead of a general displacement that allows for temperature, salinity, and pressure to change, consider instead a displacement restricted to adiabatic and isohaline conditions (i.e., no heat or salt exchanged during the parcel displacement). These sorts of fictitious displacements are physically interesting since they occur in the absence of energy needed for mixing. The density change associated with an adiabatic and isohaline displacement is determined just by pressure changes arising from the displacement, so that

$$\rho(\mathbf{x} + d\mathbf{x})_{\text{adiabatic/isohaline}} = \rho(\mathbf{x}) + \rho d\mathbf{x} \cdot \left(\frac{1}{\rho c_{\text{sound}}^2} \nabla p \right). \quad (18.128)$$

The difference in density between a parcel undergoing an adiabatic and isohaline displacement, $\rho(\mathbf{x} + d\mathbf{x})_{\text{adiabatic/isohaline}}$, and the density of the ambient environment, $\rho(\mathbf{x} + d\mathbf{x})$, is thus given by

$$\rho(\mathbf{x} + d\mathbf{x}) - \rho(\mathbf{x} + d\mathbf{x})_{\text{adiabatic/isohaline}} = \rho d\mathbf{x} \cdot (-\alpha \nabla \Theta + \beta \nabla S). \quad (18.129)$$

If a parcel makes an adiabatic and isohaline excursion and finds itself in a region where the ambient density is unchanged, then there are no buoyancy forces to resist that displacement. Directions defined by such displacements are termed *neutral directions* (McDougall, 1987a).

For the upper ocean boundary layer, we are concerned with vertical displacements and the resistance from buoyancy stratification to such motions. In this case we have

$$\rho(z+dz) - \rho(z+dz)_{\text{adiabatic/isohaline}} = \rho dz \left[-\alpha \left(\frac{\partial \Theta}{\partial z} \right) + \beta \left(\frac{\partial S}{\partial z} \right) \right] = - \left(\frac{\rho dz}{g} \right) N^2 \quad \text{vertical displacements,} \quad (18.130)$$

where the second equality introduced the buoyancy frequency (18.123). To help further expose the physics of this result, consider two cases of vertically downward parcel displacements, $dz < 0$.

- **GRAVITATIONALLY STABLE:** $N^2 > 0$: In this case, a vertically downward displacement occurring without heat or salt exchange will produce a parcel density that is less than the ambient density: $\rho(z+dz) - \rho(z+dz)_{\text{adiabatic/isohaline}} > 0$. This particular adiabatic and isohaline displacement is hence resisted by buoyancy forces. The vertical density profile is thus gravitationally stable.
- **GRAVITATIONALLY UNSTABLE:** $N^2 < 0$: Now the downward adiabatic and isohaline displacement leads to a greater density than the ambient environment: $\rho(z+dz) - \rho(z+dz)_{\text{adiabatic/isohaline}} < 0$. Hence, this particular adiabatic and isohaline displacement is encouraged through buoyancy forces to deepen even further. The vertical density profile is thus gravitationally unstable.

18.5.5.2 Non-local gravitational stability

We now extend the previous ideas to the finite depth of a surface boundary layer. That is, we develop a means to compute the gravitational stability of seawater parcels that are a finite distance from one another. The question fundamentally concerns the sign of the difference $\rho(z+\Delta) - \rho(z+\Delta)_{\text{adiabatic/isohaline}}$, where z is an arbitrary finite position in the ocean and $\Delta > 0$ is a finite distance.

Extending the result (18.130) to finite displacements upward in a water column leads to

$$\rho(z+\Delta) - \rho(z+\Delta)_{\text{adiabatic/isohaline}} = -\frac{1}{g} \int_z^{z+\Delta} N^2 \rho dz'. \quad (18.131)$$

Correspondingly, downward displacements starting from the surface at $z = \eta$ lead to

$$\rho(-\Delta) - \rho(-\Delta)_{\text{adiabatic/isohaline}} = \frac{1}{g} \int_{-\Delta}^{\eta} N^2 \rho dz'. \quad (18.132)$$

To check the signs in equation (18.132), note that for cases of $N^2 > 0$ for a full water column, an adiabatic and isohaline downward displacement always results in a less dense parcel than the surrounding water, which is expected for a full column of gravitationally stable stratification. Operationally, the left hand side of equation (18.132) is computed according to the following evaluations of the equation of state:

$$\rho(z = -\Delta) = \rho[\Theta(z = -\Delta), S(z = -\Delta), p(z = -\Delta)] \quad (18.133)$$

$$\rho(z = -\Delta)_{\text{adiabatic/isohaline}} = \rho[\Theta(z = z_1), S(z = z_1), p(z = -\Delta)]. \quad (18.134)$$

That is, the adiabatic and isohaline density at $z = -\Delta$ is computed using the temperature and salinity of the origination depth $z = z_1$, but the local pressure at $z = -\Delta$.

18.5.5.3 The Ri_b calculation

The bulk Richardson number from Large et al. (1994) (equation (18.120)), is defined as the ratio of a buoyancy difference to a squared shear of horizontal velocity. We now propose that the buoyancy difference in the numerator should be replaced by the density difference (18.132), as we showed this density difference represents a measure of the gravitational stability of a finite depth water column. However, there is one slight modification to equation (18.132), with the temperature and salinity at the origination depth corresponding to values averaged over the surface layer Θ_r, S_r , so that

$$\rho(z = -\Delta)_{\text{adiabatic/isohaline}}^{\text{KPP}} = \rho[\Theta_r, S_r, p(z = -\Delta)]. \quad (18.135)$$

These results motivate the following definition for a bulk Richardson number

$$Ri_b^{\text{KPPa}}(z = -\Delta) = \left(\frac{g \Delta}{\rho_0} \right) \left(\frac{\rho(z = -\Delta) - \rho(z = -\Delta)_{\text{adiabatic/isohaline}}^{\text{KPP}}}{|\mathbf{U}_r - \mathbf{U}(d)|^2 + U_t^2} \right), \quad (18.136)$$

with this definition according to that from Large et al. (1994) in equation (18.120). There are useful algorithmic reasons to eschew recomputing the equation of state and determining a surface layer averaged

temperature and salinity profile. For that purpose, we propose the following alternative definition, based on the right hand side of equation (18.132)

$$Ri_b^{\text{KPPb}}(z = -\Delta) = \left(\frac{\Delta}{\rho_0} \right) \left(\frac{\int_{z=-\Delta}^{\eta} N^2 \rho dz'}{|\mathbf{U}_r - \mathbf{U}(d)|^2 + U_t^2} \right). \quad (18.137)$$

Since the definition Ri_b^{KPPb} does not require recomputation of the equation of state, it is more amenable to kernalization, with the kernal given values of the buoyancy frequency, N^2 , from the calling model. On the discrete grid, there will be differences between Ri_b^{KPPa} and Ri_b^{KPPb} , but these differences are at the level of numerical roundoff, and not physically meaningful.

18.5.5.4 Unresolved shear U_t

The shear, U_t/d , in the bulk Richardson number (18.120), or any of the alternatives such as (18.136) or (18.137), is meant to acknowledge the potential presence of unresolved shears that can impact on the boundary layer depth. Large et al. (1994) present an argument on page 372 for how to compute this shear based on other parameters available from the KPP parameterization. Additional ongoing work suggests modifications for unresolved Langmuir turbulence, such as the work of McWilliams and Sullivan (2001) and Sullivan and McWilliams (2010).

18.5.5.5 Restrictions on h unstable stable buoyancy forcing

Large et al. (1994) suggest on page 372 that for stable buoyancy forcing, $B_f > 0$, the boundary layer thickness, h , should be no larger than either the Monin-Obukhov length scale, L , or the Ekman length scale,

$$h_E = 0.7 u_* / f, \quad (18.138)$$

with f the Coriolis parameter. The following reasons are noted to motivate these two restrictions.

- **MONIN-OBUKHOV:** At depths deeper than L , buoyancy stratification suppresses the mechanically forced turbulence, thus cutting off the boundary layer.
- **EKMAN:** The Ekman depth is the extent of the boundary layer in neutral stratification ($N^2 = 0$). With stable buoyancy forcing, $B_f > 0$, we then expect the boundary layer depth to be less than the Ekman depth.

As noted in Large et al. (1994) and Large and Gent (1999), the restriction based on the Monin-Obukhov has been dropped in the NCAR implementation of KPP, as it does not lead to favorable effects. Dropping this constraint is also supported by the results from Shchepetkin (2005) and Lemarié et al. (2012b). Likewise, the constraint based on the Ekman depth is not used, as little sensitivity was seen with its use. Hence, there are no restrictions for the maximum boundary layer depth under stable forcing imposed by the NCAR implementation of KPP.

The key problem with the Monin-Obukhov length scale, L , relates to the question of how to include penetrative shortwave heating in the calculation of the buoyancy forcing, B_f (Section 18.3.9). Depending on the depth over which the penetrative heating is included (equation (18.50)), one can produce a positive Monin-Obukhov length (if including sufficient shortwave heating) or negative (if including less heating). Since there is no fundamental reason to choose a particular amount of the shortwave when considering the total buoyancy forcing, there is no compelling reason to enforce the L constraint on boundary layer thickness.

VERTICAL CONVECTIVE ADJUSTMENT SCHEMES

Contents

19.1	Introduction	283
19.2	Summary of the vertical adjustment options	283
19.3	Concerning a double application of vertical adjustment	284
19.4	Implicit vertical mixing	284
19.5	Convective adjustment	284
19.5.1	Comments on the convective adjustment schemes	284
19.5.2	Coding of full convection by M. Eby	285

The purpose of this chapter is to present the vertical convective adjustment schemes available in MOM. The following MOM modules are directly connected to the material in this chapter:

ocean_param/vertical/ocean_convect.F90.
ocean_param/vertical/ocean_vert_mix.F90.

19.1 Introduction

The hydrostatic approximation necessitates the use of a parameterization of vertical overturning processes. The original parameterization used by Bryan in the 1960's was motivated largely from ideas then used for modeling convection in stars (Bryan (1969)). Work by Marshall and collaborators (Klinger et al. (1996), Marshall et al. (1997)) have largely supported the basic ideas of vertical adjustment for purposes of large-scale ocean circulation.

The Cox (1984) implementation of convective adjustment (the "NCON" scheme) may leave columns unstable after completing the code's adjustment loop. Various full convective schemes have come on-line, with that from Rahmstorf (1993) implemented in MOM. An alternative to the traditional form of convective adjustment is to increase the vertical mixing coefficient to some large value (say $\geq 10\text{m}^2\text{s}^{-1}$) in order to quickly diffuse vertically unstable water columns. Indeed, it is this form recommended from the study of Klinger et al. (1996), and it is the approach commonly used in mixed layer schemes such as Pacanowski and Philander (1981) and Large et al. (1994).

19.2 Summary of the vertical adjustment options

The handling of gravitationally unstable water columns in MOM can happen in one of two basic ways.

- **Implicit vertical mixing:** By setting the namelist parameter `aidif = 1.0`, all vertical diffusion is handled implicitly. There are two approaches depending on the vertical mixing scheme used.
 1. When using the constant vertical mixing module, the vertical diffusivity is set to a maximum value determined by a namelist `diff_cbt_limit` upon reaching a gravitationally unstable situation. `diff_cbt_limit = 10.0 m2 sec-1` is a typical value.
 2. When using the Pacanowski and Philander, KPP, or Chen vertical mixing scheme, both the vertical diffusivity and vertical viscosity are set to the namelist settings `diff_cbt_limit` and `visc_cbu_limit` upon reaching a gravitationally unstable situation. `diff_cbt_limit = visc_cbu_limit = 10.0 m2 sec-1` are typical values.
- **Convective adjustment:** There are two convective adjustment schemes in MOM. Both schemes act only on tracers when mixing. The default is the *full convect* scheme of [Rahmstorf \(1993\)](#). The alternative scheme is the older one from [Cox \(1984\)](#) and is known colloquially as the “NCON scheme”. The NCON scheme has been implemented in MOM4.0 and later releases for legacy purposes. It is not recommended for new models, with preference given to the use of a large vertical diffusivity with `aidif = 1.0`.

19.3 Concerning a double application of vertical adjustment

Whether solving the vertical diffusion equation implicitly (`aidif = 1.0`) or explicitly (`aidif = 0.0`), it is possible to use convective adjustment. To avoid a double application of vertical adjustment, one should keep in mind the following points.

When using neutral diffusion, it is necessary to have `aidif = 1.0` for numerical stability. Hence, vertical diffusion will be computed implicitly in time. For those wishing to have vertical adjustment applied just via the convective adjustment scheme, then it will be necessary to set `diff_cbt_limit = kappa_h` and `visc_cbu_limit = kappa_m`. If wishing to adjust via large vertical diffusivities, then set `diff_cbt_limit` to a large value as described above, and set the namelist `convective_adjust = .false`.

19.4 Implicit vertical mixing

When the namelist `aidif` is set to unity, vertical mixing of momentum and tracers is time stepped implicitly. When `aidif = 0.0`, vertical mixing is time stepped explicitly. Intermediate values give a semi-implicit treatment, although at present MOM does not support semi-implicit treatments. An implicit treatment of vertical mixing allows unrestrained values of the vertical mixing coefficients. Details of the time implicit algorithm are given in Chapter 13.

An explicit treatment, especially with fine vertical grid resolution, places an unreasonable limitation on the size of the time step. The use of fine vertical resolution with sophisticated mixed layer and/or neutral physics schemes has prompted the near universal implicit treatment of vertical mixing in ocean climate models.

19.5 Convective adjustment

This section provides a description of the NCON and full convective adjustment schemes.

19.5.1 Comments on the convective adjustment schemes

As detailed in [Rahmstorf \(1993\)](#) and [Pacanowski and Griffies \(1999\)](#), the NCON convection scheme takes multiple passes through the water column, alternately looking for instability on odd and even model levels. When an instability is found, tracers are mixed, with their means (weighted by cell thickness) preserved. This process may induce further instability and therefore more than one pass through the water column

may be needed to remove all instability. The number of passes through the water column is controlled by the parameter *ncon*, hence the name of the scheme.

The NCON convection scheme has come under a lot of scrutiny. The discussion in [Rahmstorf \(1993\)](#), [Killworth \(1989\)](#) and [Marotzke \(1991\)](#), provide some elaboration and motivation to *not* employ the NCON scheme. Its presence in MOM4.0 is solely for legacy purposes so that modelers can attempt to reproduce older results using the new code.

The question inevitably arises of whether to mix or not mix momentum during a vertical adjustment process. When momentum is not mixed, it is thought that it is simply carried along through the effects on the density field. [Killworth \(1989\)](#) supports this idea, so long as the purpose is large-scale ocean modeling. Basically, through geostrophy, adjusting density appears sufficient. Also, the vertical thermal wind shears in simulated convection regions were found by Killworth to not be too strong. Hence, mixing momentum along with density did little to affect the overall solution. These ideas, however, appear less sound for equatorial oceanography, and so the mixing of both momentum and tracers might be more important in there.

19.5.2 Coding of full convection by M. Eby

In June 2000, Michael Eby (eby@uvic.ca) ported a coding of the Rahmstorf scheme to MOM3.1. His code is a bit more efficient and modern than the original Rahmstorf code. In particular, there are no more “goto” statements. This code was incorporated into MOM3.1 and then to MOM4. Tests reveal that the results from the Eby code and the original Rahmstorf code are nearly identical. The researcher can uncomment/comment out two lines highlighted in the code to get the exact same results as the original Rahmstorf scheme, if one so desires.

MIXING RELATED TO TIDAL ENERGY DISSIPATION

Contents

20.1	Formulation	287
20.2	Mixing from internal wave breaking	288
20.2.1	Simmons et al (2004) scheme	288
20.2.2	Some considerations for testing the implementation	290
20.2.2.1	Regularization of the diffusivity	290
20.2.2.2	Use of the scheme for all depths	291
20.2.2.3	Energetic balances	291
20.2.2.4	Further comments	291
20.3	Dianeutral diffusivities from bottom drag	292
20.3.1	Formulation and model implementation	292
20.3.2	Caveats about spuriously large diffusivities	293

The purpose of this chapter is to summarize the MOM implementation of the dianeutral parameterization of [Simmons et al. \(2004\)](#) and [Lee et al. \(2006\)](#). Both schemes are available in MOM. These schemes provide a physically based replacement for the vertical tracer diffusivity of [Bryan and Lewis \(1979\)](#). Throughout this chapter, we assume that the mixing of interest occurs with a unit Prandtl number¹, thus enhancing both the dianeutral tracer diffusivity and momentum viscosity by equal amounts. This issue was not discussed in the work of [Simmons et al. \(2004\)](#).

Hyun-Chul Lee and Harper Simmons provided valuable comments and suggestions for this chapter. The following MOM module is directly connected to the material in this chapter:

ocean_param/vertical/ocean_vert_tidal.F90

20.1 Formulation

Dianeutral mixing of tracer and momentum arises when energy dissipates at the small scales. There are two sources of energy dissipation considered here:

- breaking internal gravity waves, where the gravity wave energy source is from barotropic tidal energy scattered into internal tidal energy occurring when tides interact with rough bottom topography,

¹The Prandtl number is the ratio of viscosity to diffusivity.

- frictional bottom drag as tides encounter continental shelves (whose depths are generally 500m or less).

To resolve both of these dissipation processes explicitly in a numerical model requires grid resolution no coarser than meters in the vertical (throughout the water column), and 1-10 kilometers in the horizontal. This very fine resolution is not generally accessible to global climate models, in which case it is necessary to consider a parameterization.

Bottom drag is typically parameterized as

$$D_{\text{bottom drag}} = C_D \mathbf{u} |\mathbf{u}|, \quad (20.1)$$

where C_D is a dimensionless drag coefficient taken as 2.4×10^{-3} by Lee et al. (2006). As discussed by Lee et al. (2006), the velocity dominating this drag is associated with energy input to the ocean via the barotropic tides as they encounter continental shelves and other shallow ocean regions. The energy dissipation (W m^{-2}) associated with this bottom drag is given by

$$E_{\text{bottom drag}} = C_D \rho_o \langle \mathbf{u}^2 \rangle |\mathbf{u}| \quad (20.2)$$

where the angle bracket symbolizes a time or ensemble average. This energy dissipation represents energy taken out of the barotropic tide and into small scale dissipation within the ocean bottom boundary layer. We assume that the dissipated energy due to bottom drag contributes to enhanced diapycnal mixing locally, with a form for the diapycnal diffusivity described in Section 20.3.

A wave drag associated with breaking internal gravity waves is written by Jayne and St.Laurent (2001) as

$$D_{\text{wave drag}} = (1/2) N_b \kappa h^2 \mathbf{u}, \quad (20.3)$$

where N_b is the buoyancy frequency at the ocean bottom, and (κ, h) are wavenumber and amplitude scales for the topography. The product κh^2 has dimensions of length and thus defines a *roughness length*

$$L_{\text{rough}} = \kappa h^2 \quad (20.4)$$

to be specified according to statistics of the observed ocean bottom topography.

The energy dissipation (W m^{-2}) associated with breaking internal gravity waves is given by

$$E_{\text{wave drag}} = (\rho_o/2) N_b L_{\text{rough}} \langle \mathbf{u}^2 \rangle. \quad (20.5)$$

In the Jayne and St.Laurent (2001) paper, they emphasize that κ , which sets the roughness length through $L_{\text{rough}} = \kappa h^2$, is used as a tuning parameter, with the tide model tuned to give sea level values agreeing with observations. Then, the energy dissipation can be diagnosed from the tide model. As with the bottom drag, the wave drag energy dissipation represents energy taken out of the barotropic tides, with the energy here transferred into the baroclinic tides. Some of the energy transferred into the baroclinic tides dissipates locally due to local wave breaking, and this then leads to enhanced diapycnal mixing locally. The remaining baroclinic energy propagates away (i.e., it is nonlocal). The ratio of local to nonlocal energy is not well known, and is the focus of research.

20.2 Mixing from internal wave breaking

When mechanical energy is dissipated, it is associated with diapycnal mixing. The relation between energy dissipation and mixing is not known precisely, though some empirical formulations have proven useful.

20.2.1 Simmons etal (2004) scheme

For energy dissipation due to breaking internal gravity waves, we follow Simmons et al. (2004), who compute a tracer diffusivity²

$$\kappa_{\text{waves}} = \kappa_0 + \frac{q \Gamma E_{\text{waves}}(x, y) F(z)}{\rho N^2}, \quad (20.6)$$

²As stated at the start of this chapter, we assume a unit Prandtl number. This assumption means the vertical viscosity is enhanced along with the diffusivity when considering internal wave breaking. Simmons et al. (2004) do not discuss vertical viscosity in their study.

where $E_{\text{wave drag}}$ is the wave energy flux from scattered barotropic to baroclinic waves, given by equation (20.5). Vertical stratification as measured by the buoyancy frequency

$$\begin{aligned}\rho N^2 &= -g \left(\frac{\partial \rho}{\partial z} \right)_p \\ &= -g \left(\frac{\partial \rho}{\partial \theta} \frac{\partial \theta}{\partial z} + \frac{\partial \rho}{\partial S} \frac{\partial S}{\partial z} \right)\end{aligned}\quad (20.7)$$

acts to suppress vertical mixing, hence its presence in the denominator of equation (20.6). The energy flux in equation (20.5) is evaluated as follows.

- N_b is computed from the model's evolving buoyancy frequency at the top face of a bottom boundary layer (often just the bottom-most tracer cell). Note that the buoyancy frequency at the bottom face of the bottom-most cell is zero, by definition.
- The effective roughness length $L_{\text{rough}} = \kappa h^2$ requires an algorithm to compute h from observed bottom topography, and tide model to tune κ . However, in practice what can be done is to take h given some variance of topography within a grid cell, and then tune $E_{\text{wave drag}}$ to be roughly 1TW in ocean deeper than 1000m, with κ as the tuning parameter.

The dimensionless parameter Γ in equation (20.6) measures the efficiency that wave energy dissipation translates into diapycnal mixing. It is often chosen as

$$\Gamma = 0.2 \quad (20.8)$$

based on [Osborn \(1980\)](#). However, in regions of very weak stratification, the mixing efficiency tends to zero according to

$$\Gamma = 0.2 \left(\frac{N^2}{N^2 + \Omega^2} \right) \quad (20.9)$$

where

$$\begin{aligned}\Omega &= \left(\frac{2\pi + 2\pi/365.24}{86400\text{s}} \right) \\ &= \left(\frac{\pi}{43082} \right) \text{s}^{-1} \\ &= 7.2921 \times 10^{-5} \text{s}^{-1}.\end{aligned}\quad (20.10)$$

is the angular rotation rate of the earth about its axis and about the sun. This modified mixing efficiency reduces the regions where spuriously large values of diffusivity may occur, especially next to the bottom, where low values of N^2 may appear. There is little physical reason to believe the huge diffusivities diagnosed from regions with $N^2 < \Omega^2$.

Another dimensionless parameter, q , is used to measure the amount of energy dissipated locally, and thus contributes to local diapycnal mixing. [Simmons et al. \(2004\)](#) chose

$$q = 1/3 \quad (20.11)$$

based on the work of [St.Laurent et al. \(2002\)](#). The remaining 2/3 of the energy is assumed to propagate away and be dissipated nonlocally. The nonlocal dissipation of internal tidal energy, as well as the dissipation of internal energy from other sources (e.g., wind energy), are accounted for in an *ad hoc* manner via the background diffusivity κ_0 (and background viscosity). A value within the range

$$\kappa_0 = (0.1 - 0.2) \times 10^{-4} \text{m}^2 \text{s}^{-1} \quad (20.12)$$

is recommended based on the measurements of [Ledwell et al. \(1993\)](#). Note that this value does not account for mixing in a planetary boundary layer, such as that discussed by [Large et al. \(1994\)](#).

Setting $q = 1/3$ globally is strictly incorrect. The actual value is related to the modal content of the excited internal tide, which is related to the roughness spectrum of topography. The redder the mode/roughness spectrum, the lower q . For example, Hawaii has been modelled as a knife-edge by (St.Laurent et al., 2003). This topography excites predominantly low modes, and these modes are stable, propagate quickly, and have long interaction times. That is, they propagate to the far field. Klymak et al. (2005) argue that $q = 0.1$ for Hawaii from the Hawaiian Ocean Mixing Experiment (HOME) data. For the mid-Atlantic ridge, the use of $q = 1/3$, as in Simmons et al. (2004), may be more suitable.

The bottom intensified vertical profile, or *deposition function*, $F(z)$ is taken as

$$\begin{aligned} F &= \frac{e^{-(D-h)/\zeta}}{\zeta (1 - e^{-D/\zeta})} \\ &= \frac{e^{h/\zeta}}{\zeta (e^{D/\zeta} - 1)}. \end{aligned} \quad (20.13)$$

In this expression,

$$D = H + \eta \quad (20.14)$$

is the time dependent thickness of water between the free surface at $z = \eta$ and the ocean bottom at $z = -H$, and

$$h = -z + \eta \quad (20.15)$$

is the time dependent distance from the free surface to a point within the water column.³ The chosen form of the deposition function is motivated by the microstructure measurements of St.Laurent et al. (2001) in the abyssal Brazil Basin, and the continental slope measurements of Moum et al. (2002). This profile respects the observation that mixing from breaking internal gravity waves, generated by scattered barotropic tidal energy, is exponentially trapped within a distance ζ from the bottom. An *ad hoc* decay scale of

$$\zeta = 500\text{m} \quad (20.16)$$

is suggested by Simmons et al. (2004) for use with internal gravity wave breaking in the abyssal ocean.

20.2.2 Some considerations for testing the implementation

We present some comments and details regarding the implementation and testing of the Simmons et al. (2004) scheme.

20.2.2.1 Regularization of the diffusivity

The diffusivities resulting from this parameterization can reach levels upwards of the maximum around $20 \times 10^{-4} \text{m}^2 \text{s}^{-1}$ seen in the Polzin et al. (1997) results. Due to numerical resolution issues, the scheme can in practice produce even larger values. We need to consider the physical relevance of these large values. The following lists some options that the modeller may wish to exercise.

- We may choose to limit the diffusivity to be no larger than a maximum value, defaulted to $50 \times 10^{-4} \text{m}^2 \text{s}^{-1}$ in MOM.
- Based on observations, the mechanical energy input from wave drag (equation (20.5)) should not exceed roughly 0.1W m^{-2} at a grid point (Bob Hallberg, personal communication 2008). Depending on details of the bottom roughness and tide velocity amplitude, a typical model implementation may easily exceed this bound. Hence, it may be necessary to cap the mechanical energy input to be no larger than a set bound.
- Use of the stratification dependent mixing efficiency (20.9) provides a physically based means to regularize the regions where N^2 can get extremely small.

³We emphasize that with a free surface, D and h are generally time dependent. Furthermore, with general vertical coordinates, h is time dependent for all grid cells.

20.2.2.2 Use of the scheme for all depths

[Simmons et al. \(2004\)](#) do not apply their scheme in waters with ocean bottom shallower than 1000m, whereas [Jayne \(2009\)](#) applies the scheme for all depths. MOM has a namelist that allows for setting a cutoff depth. For the continental shelves, the scheme of [Lee et al. \(2006\)](#) described in Section 20.3 dominates. Hence, in principle, there is nothing wrong with using the [Simmons et al. \(2004\)](#) scheme all the way to shallow waters. So one may wish to naively use $q = 1/3$ without a 1000m depth cutoff. Likewise, $\zeta = 500\text{m}$ globally may be a reasonable choice. The structure function will do the right thing and integrate to unity, whether or not the ocean depth H is greater or less than ζ .

20.2.2.3 Energetic balances

One of the main reasons to formulate diffusivities based on energy input is that energy is exchanged in a conservative manner. This conservation then leads to self-consistency tests for the model implementation. In particular, the work done against stratification by diapycnal diffusion is given by

$$\mathcal{P} \equiv \int dV \kappa_{\text{diapycnal}} \rho N^2. \quad (20.17)$$

Use of equation (20.6) for the diffusivity with a constant mixing efficiency $\Gamma = 0.2$ yields

$$\begin{aligned} \mathcal{P} &= \int dV (\kappa_{\text{wave drag}} - \kappa_0) \rho N^2 \\ &= q\Gamma \int dx dy E_{\text{wave drag}}(x, y) \end{aligned} \quad (20.18)$$

assuming $q\Gamma$ constant. Note that to reach this result, we set $\int F(z) dz = 1$, which is a constraint that is maintained by the model implementation. Equation (20.18) says that the energy deposited in the ocean interior that works against stratification originates from that scattered from the ocean bottom. For the general case of $q\Gamma$ spatially dependent, we have the balance

$$\begin{aligned} \mathcal{P} &= \int dV (\kappa_{\text{wave drag}} - \kappa_0) \rho N^2 \\ &= \int dx dy dz q\Gamma E_{\text{wave drag}}(x, y) F(z), \end{aligned} \quad (20.19)$$

which again is a statement of energy conservation between wave dissipation and mixing of density.

Although equation (20.19) is a trivial identity following from the definition of the closure, it is not trivial to maintain in the ocean model. The main reason is that we work with diffusivities when integrating the equations of an ocean model, and these diffusivities are often subjected to basic numerical consistency criteria, such as the following.

- We may wish to have the diffusivities monotonically decay upwards in the column. Given the N^{-2} dependence of the diffusivity in equation (20.6), monotonicity is not necessary. Without an added monotonicity constraint, the simulation can be subject to spurious instabilities in which intermediate depths destratify, then producing larger diffusivities, and further reducing the stratification. [Jayne \(2009\)](#) discovered this behaviour in his simulations.
- The diffusivities should be bounded by a reasonable number, such as $50 - 100 \text{ cm}^2 \text{ sec}^{-1}$.

Imposing constraints such as these on the diffusivity corrupts the identity (20.18). In general, the constraints remove energy from the interior, so that in practice $\int dV (\kappa_{\text{wave drag}} - \kappa_0) \rho N^2 < \int dx dy dz q\Gamma E_{\text{wave drag}}(x, y)$.

20.2.2.4 Further comments

Here are some further points to consider when setting some of the namelists for this scheme.

- One means to ensure that the diffusivities are within a reasonable bound, without capping them after their computation, is to artificially restrict the stratification used in the calculation to be no less than a certain number. [Simmons et al. \(2004\)](#) chose the floor value $N^2 \geq 10^{-8} \text{ s}^{-2}$. There is a great deal of sensitivity to the floor value used. GFDL practice is to keep the floor value quite low so that $N_{\min}^2 < \Omega^2$.
- If the maximum diffusivity realized by the scheme is allowed to be very large, say much greater than as $1000 \text{ cm}^2 \text{ sec}^{-1}$, then the near bottom stratification can become very small. In this case, $E_{\text{wave drag}}$ can dip below the canonical 1TW value. This process resembles a negative feedback in some manner, though it has not been explored extensively.

20.3 Dianeutral diffusivities from bottom drag

The [Lee et al. \(2006\)](#) scheme provides a means to parameterize mixing from barotropic tides interacting with the continental shelf regions. The purpose of this section is to detail the scheme and present some discussion of its implementation in MOM.

20.3.1 Formulation and model implementation

Contrary to the energetic approach of [Simmons et al. \(2004\)](#), the [Lee et al. \(2006\)](#) scheme does not consider energetic arguments for determining the diffusivity associated with barotropic tides dissipated by the bottom boundary layer. Instead, it follows the older ideas of [Munk and Anderson \(1948\)](#), whereby a dianeutral diffusivity is given by

$$\kappa_{\text{drag}} = \kappa_{\text{max}} (1 + \sigma \text{ Ri})^{-p} \exp^{-(D-h)/z_{\text{tide}}}, \quad (20.20)$$

where the dimensionless parameters σ and p have the default values

$$\sigma = 3 \quad (20.21)$$

$$p = 1/4. \quad (20.22)$$

The Richardson number is given by

$$\text{Ri} = \frac{N^2}{|\partial_z \mathbf{u}|^2}. \quad (20.23)$$

Small Richardson numbers (e.g., regions of low stratification or strong vertical shear) will give larger vertical diffusivities, with the maximum diffusivity set by κ_{max} . Following [Lee et al. \(2006\)](#), we set the default for the maximum diffusivity arising from bottom drag dissipation as

$$\kappa_{\text{max}} = 5 \times 10^{-3} \text{ m}^2 \text{ s}^{-1}. \quad (20.24)$$

Since we do not generally resolve the bottom boundary layer in global models, we must approximate the vertical shear to compute the Richardson number, and here we use the form

$$2|\partial_z \mathbf{u}|^2 = \left(\frac{\tilde{U}_{\text{tide}}}{D-h} \right)^2, \quad (20.25)$$

with the scaled tidal speed \tilde{U}_{tide} given by

$$\tilde{U}_{\text{tide}} = U_{\text{tide}} \sqrt{Cd} / \kappa_{\text{von Karman}}. \quad (20.26)$$

Here, Cd is the bottom drag coefficient, taken as $Cd = 2.4 \times 10^{-3}$ by [Lee et al. \(2006\)](#),

$$\kappa_{\text{von Karman}} = 0.41 \quad (20.27)$$

is the von Karman constant, and U_{tide} is the tidal speed taken from a barotropic tidal model. These speeds are largest in the shallow regions.

20.3.2 Caveats about spuriously large diffusivities

The exponential decay appearing in equation (20.20) is not part of the original [Lee et al. \(2006\)](#) scheme, nor was it part of the MOM4.0 and MOM4p1 implementations. However, it is an essential element added for the MOM implementation as of 2012 that ensures diffusivities drop off exponentially when moving away from the ocean bottom. Absent this exponential decay, regions of small Richardson number, leading to large κ_{drag} , can move upwards in a column, as revealed by the diffusivities in Figure 20.1. The chosen exponential decay length scale is given by

$$z_{\text{tide}} = \tilde{U}_{\text{tide}} \frac{\tau_{\text{tide}}}{2\pi} \quad (20.28)$$

where

$$\tau_{\text{tide}} = 12 \times 3600 \text{ s}, \quad (20.29)$$

corresponding to the M2 tide period. Another means for removing the spurious diffusivities from the [Lee et al. \(2006\)](#) scheme is to enable the scheme *only* in continental shelf regions, which is where it is physically appropriate. Such is the default for the MOM implementation.

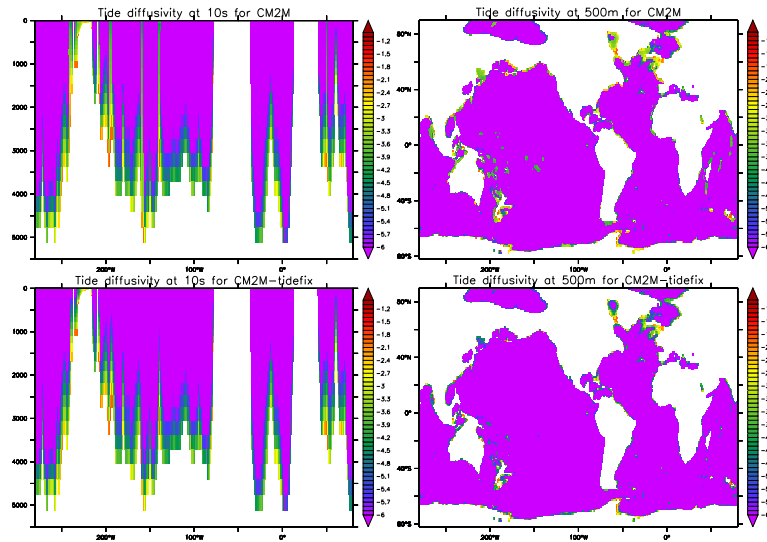


Figure 20.1: Shown here are the diffusivities resulting from the sum of the [Simmons et al. \(2004\)](#) and [Lee et al. \(2006\)](#) schemes. The ocean component is taken from a coupled climate simulation forming the basis for the ESM2M earth system model of [Dunne et al. \(2012\)](#), which uses a grid configured as in the CM2.1 model documented by [Delworth et al. \(2006\)](#), [Griffies et al. \(2005\)](#) and [Gnanadesikan et al. \(2006\)](#), but with updated numerics and physical parameterizations. Note in particular the nontrivial values in certain “hot-spots” in the Pacific for the top row of figures. These regions arise from spurious treatment of the original [Lee et al. \(2006\)](#) scheme, where the absence of an exponential decay exposes the scheme to a positive feedback, whereby large diffusivities move upward in the column from regions where there is a nontrivial barotropic tide. The bottom row of figures results from applying the exponential decay in equation (20.20), which removes the spurious diffusivities. The result is a far more sensible diffusivity profile that is bottom intensified and does not penetrate through the pycnocline from deep ocean regions.

MIXING RELATED TO SPECIFIED MINIMUM DISSIPATION

Contents

21.1	Formulation	295
----------------------	---------------------------------------	-----

The purpose of this chapter is to summarize an option in MOM to specify the vertical tracer diffusivities based on setting a floor to the power dissipation. The following MOM module is directly connected to the material in this chapter:

ocean_param/vertical/ocean_vert_mix.F90

21.1 Formulation

Vertical tracer diffusion is associated with a dissipation of power. Assuming temperature and salinity have the same vertical diffusivities leads to the expression for power dissipation (W m^{-3})

$$\begin{aligned}
\epsilon &= \rho \kappa N^2 \\
&= -\kappa g \left(\frac{\partial \rho}{\partial z} \right)_p \\
&= -\kappa g \left(\frac{\partial \rho}{\partial \theta} \frac{\partial \theta}{\partial z} + \frac{\partial \rho}{\partial S} \frac{\partial S}{\partial z} \right).
\end{aligned} \tag{21.1}$$

In these equations, κ is the vertical tracer diffusivity, g is the gravitational acceleration, and the vertical density derivative is locally referenced. When the temperature and salinity diffusivities differ, as occurs with double diffusion, we compute the power dissipation via

$$\epsilon = -\kappa_{\text{temp}} g \left(\frac{\partial \rho}{\partial \theta} \frac{\partial \theta}{\partial z} \right) - \kappa_{\text{salt}} g \left(\frac{\partial \rho}{\partial S} \frac{\partial S}{\partial z} \right). \tag{21.2}$$

We now compute a floor to the dissipation according to

$$\epsilon_{\text{floor}} = \epsilon_{\text{min}} + B|N|, \tag{21.3}$$

where

$$\epsilon_{\text{min}} \sim 10^{-6} \text{ W m}^{-3} \tag{21.4}$$

is a specified minimum power dissipation (set according to a namelist),

$$B \sim 1.5 \times 10^{-4} \text{ J m}^{-3} \quad (21.5)$$

is another namelist parameter, and $|N|$ is the absolute value of the buoyancy frequency. The $B|N|$ contribution to dissipation is motivated by the stratification dependent diffusivity proposed by [Gargett \(1984\)](#). We establish a floor to the vertical diffusivity according to

$$\begin{aligned} \kappa_{\text{floor}} &= \frac{\epsilon_{\text{floor}} \Gamma^{\text{regularized}}}{\rho N^2} \\ &\approx \frac{0.2 \epsilon_{\text{floor}}}{\rho_o (N^2 + \Omega^2)}. \end{aligned} \quad (21.6)$$

In this equation,

$$\Gamma^{\text{regularized}} = \frac{0.2 N^2}{N^2 + \Omega^2} \quad (21.7)$$

is a regularized mixing efficiency, and

$$\Omega = 7.2921 \times 10^{-5} \text{ s}^{-1} \quad (21.8)$$

is the angular rotation rate of the earth about its axis and around the sun. The tracer diffusivity for temperature, salinity, and passive tracers is not allowed to be smaller than κ_{floor} .

Chapter 22

PARAMETERIZATION OF FORM DRAG

Contents

22.1	Regarding the TEM approach	297
22.2	What is available in MOM	298

This chapter is a placeholder for documentation of the form drag parameterizations available in MOM. At present, MOM has implemented the schemes proposed by [Greatbatch and Lamb \(1990\)](#), [Aiki et al. \(2004\)](#), and [Ferreira and Marshall \(2006\)](#). The following MOM module is directly connected to the material in this chapter:

ocean_param/vertical/ocean_form_drag.F90.

Notably, these schemes are experimental and have not been thoroughly used at GFDL.

22.1 Regarding the TEM approach

We comment here on the relevance of implementing the [Gent and McWilliams \(1990\)](#) scheme via the tracer equation, as traditionally done in MOM as motivated by [Gent et al. \(1995\)](#) and [Griffies \(1998\)](#), versus the alternative, which adds a vertical stress to the horizontal momentum equation, as recently implemented in a global model by [Ferreira and Marshall \(2006\)](#). When adding a stress to the momentum equation, the prognostic velocity variable is interpreted as the residual mean, or effective velocity, rather than the traditional Eulerian mean velocity. This *transformed Eulerian mean* (TEM) interpretation is quite elegant, since it is the residual mean velocity that advects tracers in a coarsely resolved (i.e., no mesoscale eddies) z-model, *not* the Eulerian mean velocity. The elegance is maintained so long as one need not compute the Eulerian mean velocity.

For many purposes, we do not require the Eulerian mean velocity, so the TEM momentum equation provides all the variables required to run an ocean model. However, the following considerations point to a need for the Eulerian mean velocity in cases of realistic ocean climate modeling.

- In computing the air-sea stress in realistic coupled climate models, it is important for many purposes to include the velocity of the ocean currents according to the discussion in [Pacanowski \(1987\)](#). The relevant currents for this calculation are the Eulerian mean currents, not the residual mean.
- When computing the Richardson number commonly required for mixed layer parameterizations, we require the vertical shear of the Eulerian mean velocity, not the vertical shear of the residual mean velocity.

There are two options that one may consider. First, one may choose to ignore the difference between the Eulerian mean and the residual mean velocity. Alternatively, one may choose to diagnose the Eulerian mean

from the residual mean. The Eulerian mean velocity is available within a TEM ocean model, given the prognostic residual mean velocity plus a prescribed mesoscale eddy closure to compute the eddy induced velocity. Its calculation requires derivatives of the quasi-Stokes streamfunction to obtain the eddy-induced velocity, and one further derivative to compute the vertical shear. As discussed by [Griffies \(1998\)](#), this calculation can produce a rather noisy eddy induced velocity, especially near boundaries. Furthermore, there is no general principle guiding us in formulating a particular choice of discretization for the eddy induced velocity from a streamfunction. A noisy eddy induced velocity produces a noisy diagnosed Eulerian mean velocity, which then puts noise in the air-sea stress as well as the Richardson number. These numerical sources of noise are unacceptable for realistic climate modeling, and represent a practical barrier to making use of the TEM approach.

22.2 What is available in MOM

In addition to the practical limitations noted above, there are further theoretical concerns with the TEM approach, as its formulation generally assumes a small Rossby number. Nonetheless, the idea is compelling and as such it may be suitable for study, especially in idealized settings such as that of [Ferreira and Marshall \(2006\)](#) and [Zhao and Vallis \(2008\)](#). Additionally, the scheme from [Aiki et al. \(2004\)](#) is quite novel, and may present some interesting features of use for climate modeling. It is for these reasons that the form drag approach has been implemented in MOM. Given the absence of extensive experience with the scheme at GFDL, we recommend that it be used only by those who are focused on research related to the use of these schemes.

Subgrid scale parameterizations for lateral processes

The purpose of this part of the manual is to describe certain of the subgrid scale (SGS) parameterizations of physical processes used in MOM, with focus here on lateral processes.

Chapter 23

Neutral Physics

This chapter gives a description of a new implementation of the neutral physics component of MOM, with the implementation coded by Tim.Leslie@gmail.com. The following MOM modules are directly connected to the material in this chapter:

```
ocean_param/neutral/ocean_nphysics.diff.F90
ocean_param/neutral/ocean_nphysics.flux.F90
ocean_param/neutral/ocean_nphysics.new.F90
ocean_param/neutral/ocean_nphysics.skew.F90
ocean_param/neutral/ocean_nphysics.tensor.F90
ocean_param/neutral/ocean_nphysics.util.new.F90
```

This new implementation breaks bitwise compatibility with the original code. However, it addresses some issues while simplifying and clarifying much of the code. The original code is contained in the modules

```
ocean_param/neutral/ocean_nphysicsA.F90
ocean_param/neutral/ocean_nphysicsB.F90
ocean_param/neutral/ocean_nphysicsC.F90
ocean_param/neutral/ocean_nphysics.F90
ocean_param/neutral/ocean_nphysics.util.F90
```

The original code is retained for legacy purposes, and as a means to carefully move forward with the new code through testing various options.

This chapter was written by

tim.leslie@gmail.com

with help from

Stephen.Griffies@noaa.gov

There are many unfinished pieces to this chapter, reflecting the ongoing development, testing, and documentation of the new neutral physics code.

Caveat

Given the early stages of testing for the new implementation of neutral physics, we recommend that the general user employ the original version of neutral physics, with these original options part of the MOM test cases.

Contents

23.1	Introduction	303
23.1.1	Neutral diffusion and neutral skewness	303
23.1.1.1	Advection-diffusion equations	304
23.1.1.2	Neutral Diffusion Parameterization	305
23.1.1.3	Skew Diffusion Parameterization	305
23.1.2	Overview of this chapter	306
23.2	Notation	306
23.2.1	Grid Points	306
23.2.2	Tensors	306
23.2.3	Cell Faces	307
23.2.4	Spanning Pairs	307
23.2.5	Triads	307
23.2.6	Harmonic Mean	308
23.3	Discretization	309
23.3.1	Functional	309
23.3.2	Fluxes	312
23.3.3	Skew flux discretization	314
23.4	Implementation	315
23.4.1	Fluxes	315
23.4.1.1		316
23.4.1.2		316
23.4.1.3	Tracer Limiting	316
23.4.1.4		317
23.4.1.5		319
23.4.1.6	Density Weighted Quarter-cell Volumes	319
23.5	Diffusion and Skew-Diffusion Tensors	320
23.5.1	Depth Taper	320
23.5.2	Neutral Diffusion Tensor	321
23.5.2.1	Tapers	322
23.5.3	Skew-Diffusion Tensor	323
23.5.3.1	Preliminaries	323
23.5.3.2	GM Skew-Diffusion Tensor	324
23.5.3.3	Baroclinic Modes	325
23.5.3.4	Boundary Value Problem	326
23.5.4	Vertical Boundary Tensor Values	328
23.6	Tracer Gradients	328
23.7	Quantities related to density gradients	329
23.7.1	Gradient of locally referenced potential density	329
23.7.1.1	Optional adjustments of the vertical density derivative	330
23.7.1.2	Diagnostics	330
23.7.2	Neutral Slopes	331
23.7.2.1	Neutral Slope Vector	331
23.7.2.2	Neutral Slope Magnitude	331
23.7.2.3	Diagnostics	332
23.7.3	Buoyancy	332
23.7.4	Baroclinic Gravity Wave Speed	332
23.7.5	Rossby Radius	332
23.7.6	Neutral Boundary Layers	333

23.7.7 Summary	333
23.8 Specification of the diffusivity	333
23.8.1 Namelist Parameters	334
23.8.2 Fixed GM Diffusivity	335
23.8.3 Flow Dependent GM Diffusivity	335
23.8.3.1 Selecting a GM Diffusivity Closure	335
23.8.3.2 Diffusivity Postprocessing	335
23.8.3.3 Grid Scaling	336
23.8.4 MICOM	336
23.8.5 Buoyancy Scaling	336
23.8.6 Rate Length Squared	337
23.8.6.1 Growth Rate	337
23.8.6.2 Length	340
23.8.6.3 Baroclinic Zone Width	341
23.8.7 Redi Diffusivity	342
23.9 Summary of the notation	342
23.9.1 General	343
23.9.2 Tracer Gradients	343
23.9.3 Density Calculations	344
23.9.4 Neutral Boundary Layer	344
23.9.5 Diffusivity	345
23.9.6 Tensors	345
23.9.7 Fluxes	346

23.1 Introduction

This chapter details the neutral physics parameterization as implemented in MOM. The two processes described are *neutral diffusion*, which diffuses tracers along neutral directions, and *skew diffusion* or *skew-sion*, which stirs tracers. These processes aim to parameterize the mixing and stirring effects of mesoscale eddies. Both neutral and skew diffusion can be mathematically described in terms of a generalised transport tensor, which in turn allows us to unify many aspects of the calculation (Griffies, 1998).

The MOM implementation of neutral physics is limited to the use of quasi-horizontal coordinates z , z^* , p , and p^* (see Chapter 5). As discussed in Section 4.2.3, when working with these coordinates, there is no modification to the algorithm relative to that of the original geopotential implementation. However, terrain following coordinates, which are not quasi-horizontal, require extra considerations, such as those detailed in Lemarié et al. (2012b,a).

23.1.1 Neutral diffusion and neutral skewsion

Tracer prognostics within MOM are computed in terms of the non-Boussinesq thickness weighted tracer concentration, $dz\rho T$. This quantity evolves according to a range of physical processes. We can therefore write the tracer evolution equation as

$$\frac{\partial}{\partial t}(dz\rho T) = \sum_{\text{processes}} \frac{\partial}{\partial t}(dz\rho T) \quad (23.1)$$

where we sum over all the physical processes. When computing the neutral physics processes, we choose to write the tracer evolution as

$$\frac{\partial}{\partial t}(dz\rho T) = dz \frac{\partial}{\partial t}(\rho T)^{\text{neutral}} + \sum_{\text{non-neutral}} \frac{\partial}{\partial t}(dz\rho T). \quad (23.2)$$

This chapter focuses on the first term in this equation, which we can write as $dz(\rho T)_{,t}^{\text{neutral}}$. We will parameterize the neutral processes with an advection-diffusion equations, so

$$(\rho T)_{,t}^{\text{neutral}} = (\rho T)_{,t}^{\text{adv}} + (\rho T)_{,t}^{\text{diff}} \quad (23.3)$$

$$= -\nabla \cdot (\mathbf{v} \rho T - \nabla \cdot (\rho \mathbf{K} \nabla T)), \quad (23.4)$$

where \mathbf{K} is a positive-semidefinite, symmetric diffusion tensor and

$$\nabla \cdot (\rho \mathbf{v}) = 0. \quad (23.5)$$

We can write this equation in a number of equivalent forms. These different forms will help guide the discretization used below.

23.1.1.1 Advection-diffusion equations

As shown in [Griffies \(1998\)](#), we can write both the advective and diffusive terms as the divergence of fluxes which gives

$$(\rho T)_{,t}^{\text{neutral}} = -\nabla \cdot \rho (\mathbf{F}^{\text{adv}} + \mathbf{F}^{\text{diff}}), \quad (23.6)$$

where

$$\mathbf{F}^{\text{adv}} = \mathbf{v} T \quad (23.7)$$

$$\mathbf{F}^{\text{diff}} = -\mathbf{K} \nabla T \quad (23.8)$$

are tracer concentration fluxes. Since $\rho \mathbf{v}$ is divergence free, it can be written as the curl of a stream function plus the divergence of an arbitrary gauge field,

$$\rho \mathbf{v} = \nabla \times (\rho \Psi + \nabla \phi). \quad (23.9)$$

We ignore the gauge function in the following, though our choice for Ψ implicitly assumes a choice. The corresponding advective tracer flux is given by

$$\rho \mathbf{F}^{\text{adv}} = (\nabla \times \rho \Psi) T \quad (23.10)$$

$$= -\nabla T \times \rho \Psi + \nabla \times (\rho \Psi T). \quad (23.11)$$

It follows that the advective portion of the tracer equation can be written as the convergence of a skew flux

$$(\rho T)_{,t}^{\text{adv}} = -\nabla \cdot \rho \mathbf{F}^{\text{skew}} \quad (23.12)$$

where the skew flux is defined as

$$\rho \mathbf{F}^{\text{skew}} = \rho \Psi \times \nabla T. \quad (23.13)$$

We can write equation 23.3 as a generalised transport equation,

$$(\rho T)_{,t}^{\text{neutral}} = -\partial_m (\rho F^m) \quad (23.14)$$

$$F^m = J^{mn} T_{,n}. \quad (23.15)$$

J^{mn} is a transport tensor, and we have used tensor notation with repeated indices summed over their range, and a comma denotes a partial derivative (see [Griffies \(2004\)](#) Chapter 20). The transport tensor can be decomposed into symmetric and anti-symmetric terms, to give

$$J^{mn} = K^{mn} + A^{mn} \quad (23.16)$$

where

$$K^{mn} = K^{nm} \quad (23.17)$$

$$A^{mn} = -A^{nm}. \quad (23.18)$$

The symmetric tensor is identified as the neutral diffusion tensor \mathbf{K} in equation 23.4. The antisymmetric tensor A corresponds to the operator $\Psi \times$ in equation 23.13. We call A the skew diffusion tensor, and it can be written as

$$A = \begin{pmatrix} 0 & -\Psi_z & \Psi_y \\ \Psi_z & 0 & -\Psi_x \\ -\Psi_y & \Psi_x & 0 \end{pmatrix}. \quad (23.19)$$

We can now define our neutral physics parameterization in terms of \mathbf{K} and Ψ .

23.1.1.2 Neutral Diffusion Parameterization

The neutral diffusion tensor parameterizes mixing from eddies according to a diffusion acting along neutral directions. The small slope approximated form of this tensor [Gent and McWilliams \(1990\)](#); [Griffies et al. \(1998\)](#) is given by

$$K = A_R \begin{pmatrix} 1 & 0 & S_x \\ 0 & 1 & S_y \\ S_x & S_y & |\mathbf{S}|^2 \end{pmatrix}, \quad (23.20)$$

where

$$\mathbf{S} = (S_x, S_y, 0) \quad (23.21)$$

is a horizontal vector measuring the slope of the neutral direction relative to the horizontal. We compute this slope vector according to the following expression

$$\mathbf{S} = - \left(\frac{\rho_{,\Theta} \nabla_s \Theta + \rho_{,S} \nabla_s S}{\rho_{,\Theta} \partial_z \Theta + \rho_{,S} \partial_z S} \right), \quad (23.22)$$

where

$$\rho_{,\Theta} = \frac{\partial \rho}{\partial \Theta} \quad (23.23)$$

$$\rho_{,S} = \frac{\partial \rho}{\partial S} \quad (23.24)$$

are density partial derivatives with respect to conservative temperature and salinity. The lateral gradient operator, ∇_s , is taken along surfaces of constant vertical coordinates, s . The only vertical coordinates in MOM that are supported for neutral physics are the quasi-horizontal coordinates z , z^* , p , and p^* (see Chapter 5 for a discussion of vertical coordinates). Hence, it is very accurate to compute the neutral slope vector by taking lateral gradients along the constant vertical coordinate surfaces. The diffusivity A_R sets the overall magnitude of the neutral diffusion, with this diffusivity sometimes called the Redi diffusivity, reflecting the work of [Redi \(1982\)](#) who formulated the kinematics of neutral diffusion. A discussion of neutral diffusion, with details of how the MOM discretization is formulated, is given by [Griffies et al. \(1998\)](#) and [Griffies \(2004\)](#).

23.1.1.3 Skew Diffusion Parameterization

The skew diffusion tensor parameterizes mesoscale eddy stirring, and it is given by

$$A = \begin{pmatrix} 0 & 0 & \Upsilon_x \\ 0 & 0 & \Upsilon_y \\ -\Upsilon_x & -\Upsilon_y & 0 \end{pmatrix}, \quad (23.25)$$

where Υ is the parameterized transport vector. Comparing this with equation 23.19 we note that

$$\Upsilon = \hat{\mathbf{z}} \times \Psi. \quad (23.26)$$

We are able to make this choice due to the existence of the gauge term $\nabla\phi$ in equation 23.9. Most parameterizations of this transport vector are based on the Gent-McWilliams (GM) parameterization [Gent and McWilliams \(1990\)](#); [Griffies \(1998\)](#),

$$\mathbf{r}^{\text{GM}} = -\kappa \mathbf{S}, \quad (23.27)$$

which leads to the skew tensor

$$A^{\text{GM}} = \kappa \begin{pmatrix} 0 & 0 & -S_x \\ 0 & 0 & -S_y \\ S_x & S_y & 0 \end{pmatrix}. \quad (23.28)$$

However, [Ferrari et al. \(2010\)](#) propose an alternative to the traditional GM parameterization, with this alternative also implemented in MOM. Indeed, it is the recommended approach, as the [Ferrari et al. \(2010\)](#) method provides for an elegant treatment of weakly stratified regions, such as the surface and bottom boundary layers.

23.1.2 Overview of this chapter

The remainder of this chapter details the discretization and parameterization of the equations described above. Section 23.6 details the calculation of T_n for each of the tracers in the model. In Section 23.7 the calculation of the neutral slope vector \mathbf{S} is described, along with some other required quantities which depend directly on the density gradients. The parameterization of the diffusivities A_R and κ are described in Section 23.8. Having calculated these values, Section 23.5 outlines the calculation of the two tensors, K and A . Finally, Section 23.3.2 describes the calculation of the flux vector \mathbf{F} for each tracer, which allows us to calculate the density-thickness weighted tendencies defined in Equations (23.3) and (23.4). Additionally, Appendix A provides a summary of the values described in each section, along with the corresponding variable names in the implementation.

23.2 Notation

The implementation of the neutral physics module requires the description of variables that are components of a rank two tensor located on a two dimensional sub-grid at a point in three dimensional space. Furthermore, these variables may be arranged in one of four configurations. This arrangement leads to the requirement of some relatively advanced indexing notation, which we will describe here. We will also describe how the indexing notation corresponds to variable naming conventions within the implementation.

23.2.1 Grid Points

Most variables are associated with a particular point in space. Within the model, these points correspond to the tracer cell locations. Note that velocity cells are not used in this module, which means the treatment of neutral physics is identical for B-grid and C-grid ocean models. Three dimensional fields have their spatial location marked with the subscripts i, j, k , while two dimensional fields use i, j . These subscripts correspond to array indices within the code. For example, the variable A might be represented by the array A in the code, and the value $A_{i,j,k}$ would correspond to $A(i, j, k)$.

23.2.2 Tensors

The individual components of a rank two tensor are denoted by a pair of subscripts mn , where m is the row and n is the column in the matrix representation of the tensor. For example a tensor A has components

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix}. \quad (23.29)$$

To represent the value of a component at a grid point, parentheses are applied before the grid subscript, giving e.g. $(A_{23})_{i,j,k}$. In the code, separate variables are used for each component, with the component label usually included in the name, e.g. $A_{23}(i, j, k)$.

23.2.3 Cell Faces

Some variables, such as the fluxes, are defined at the faces of the tracer cell, which is the appropriate location when considering a finite volume interpretation of the tracer equation. By convention, we define values at the eastern, northern and bottom faces in each of the directions respectively. Cell face variables are thus indicated with a superscript of E , N and b respectively. So for example, the flux through the eastern face of the i, j, k tracer cell is written $F_{i,j,k}^E$. In the code, such variables are denoted by appending `_xte`, `_ytn` and `_ztb` respectively to the variable names. So the flux $F_{i,j,k}^E$ is represented by the variable `flux_xte(i, j, k)`.

23.2.4 Spanning Pairs

Some variables, such as gradients and cell widths, span multiple cells. A variable can span cells horizontally to the left or right, or vertically up or down. These variables naturally occur as pairs, either horizontally left-right or vertically up-down. To represent these variables, an additional superscript is added, which takes the value zero or one (left/up = 0, right/down = 1). This index has the name ip , jq or kr for the x , y or z directions respectively. For example, a variable that spans horizontally to the left and right in the x direction is called $A_{i,j,k}^{(ip)}$. The corresponding variable in the code is a four dimensional array and the variable name has a suffix of `_x`, `_y` or `_z` as appropriate. The variable $A_{i,j,k}^{(ip)}$ is therefore represented as `A_x(i, j, k, ip)`.

23.2.5 Triads

When combining two variables that span different directions, there are four possible combinations to consider, i.e. left-up, right-down, etc. These combinations use a three point stencil, and hence are called *triads*. Figure 23.1 shows an example of a set of four triads with a common central point. When arranged in this fashion the group is called a *centred triad group*. For a given point, the triads are indexed with a pair of numbers as shown in the figure. When written as a variable, a parenthetical pair of superscripts are used for these indexes. In the $x - z$ plane they have the generic indices (ip, kr) while in the $y - z$ plane indices (jq, kr) are used. A variable in the $x - z$ plane is therefore written as $A_{i,j,k}^{(ip,kr)}$. In the code, such a variable is represented with a five dimensional array, with the last two dimensions representing the ip and kr indices. Furthermore, to indicate the plane being used, a suffix of `_xz` or `_yz` is appended to the variable. Therefore, our variable $A_{i,j,k}^{(ip,kr)}$ is written in code as `A_xz(i, j, k, ip, kr)`.

As with regular variables, we will need to have groups of triads centred on the faces of tracer cells. Figure 23.2 shows an example of such *horizontal* and *vertical face centred triad groups*. As with regular variables, a face centred triad variable has an extra superscript of E , N or b . In the code, an extra suffix of either `_h` or `_v` is appended to indicate a horizontal or vertical face centred triad group. Hence, the horizontal face centred triad group variable A , defined in the $x - y$ plane, has indices $A_{i,j,k}^{E(ip,kr)}$ and is coded as `A_hxz(i, j, k, ip, kr)`.

A variable defined over the centered triad group may be needed over the face centred triad group. Mathematically we can indicate this shift implicitly by using a different indexing scheme. However, in the code implementation, data must be copied from one array to another to accommodate the different indexing scheme. The module `ocean_nphysics_util_new` provides the subroutines `stencil_centre_to_vert` and `stencil_centre_to_horiz` to perform the appropriate copying actions. We do this shift mathematically by

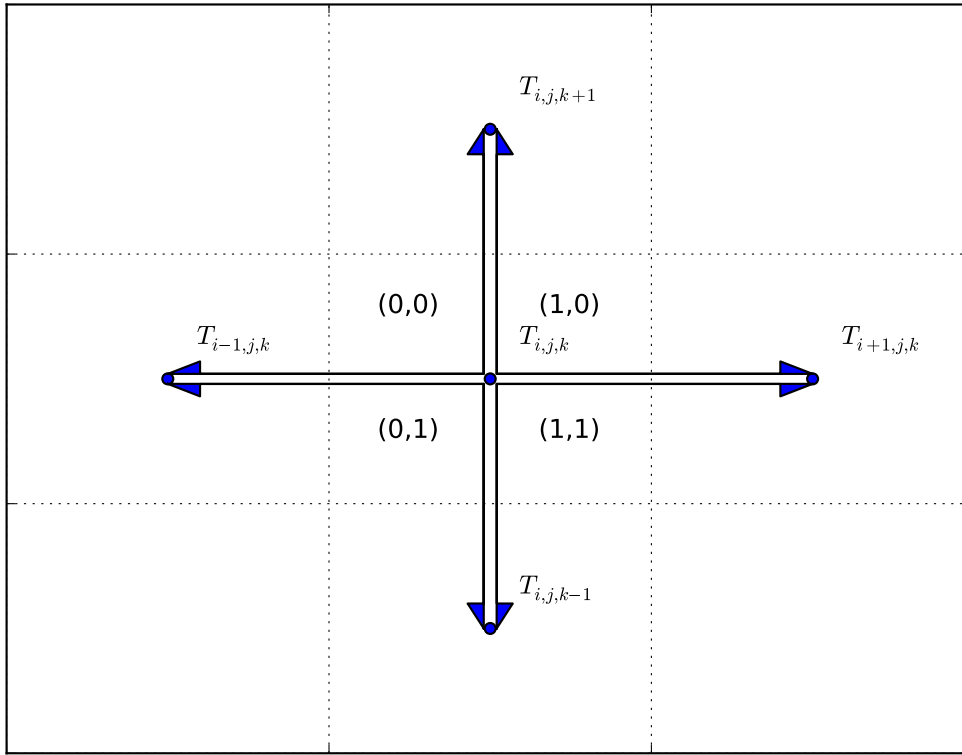


Figure 23.1: Stencils and indices for a centered triad group.

the functions $C2V$ and $C2H$ respectively, so the transformation of an array is written as

$$A_{i,j,k}^{E(ip,kr)} = C2H\left(A_{i,j,k}^{(ip,kr)}\right) \quad (23.30)$$

$$A_{i,j,k}^{N(jq,kr)} = C2H\left(A_{i,j,k}^{(jq,kr)}\right) \quad (23.31)$$

$$A_{i,j,k}^{b(ip,kr)} = C2V\left(A_{i,j,k}^{(ip,kr)}\right) \quad (23.32)$$

$$A_{i,j,k}^{b(jq,kr)} = C2V\left(A_{i,j,k}^{(jq,kr)}\right). \quad (23.33)$$

23.2.6 Harmonic Mean

In a number of places a harmonic mean function is used to place a smooth but hard cap on the size of certain values. This function is defined as

$$H(a,b) = \frac{2ab}{a+b}. \quad (23.34)$$

It has the useful property that

$$\lim_{a \rightarrow \infty} H(a,b) = 2b \quad (23.35)$$

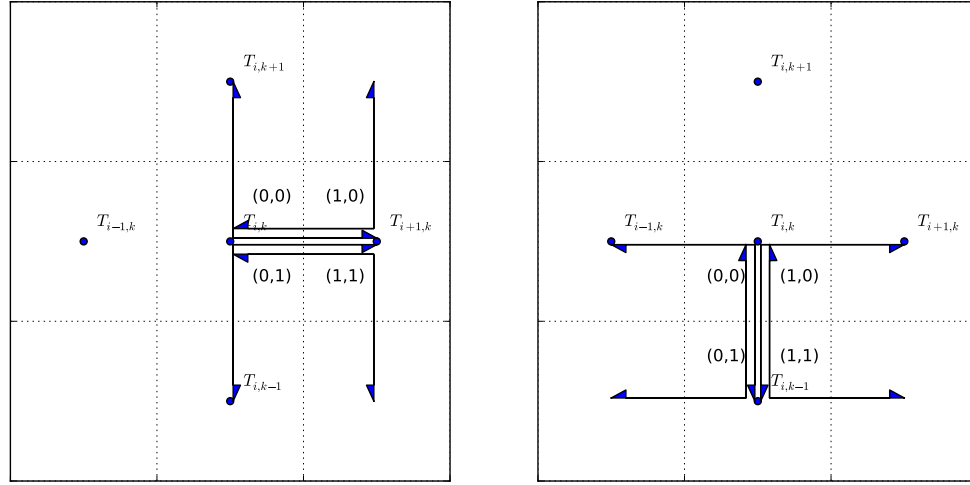


Figure 23.2: Stencils and indices for horizontal and vertical face centered triad groups.

which means that in the transformation $a \rightarrow H(a, b)$, a is capped above by the value $2b$, but approaches this value smoothly.

23.3 Discretization

This section describes a discretization of the term $-\nabla \cdot \rho \mathbf{F}^{\text{diff}}$ in equation 23.6. The discretization of the skew flux follows by direct analogy. A finite volume approach is used, however certain choices are made to maintain key properties of the continuous equations.

The finite volume approximation can be written as

$$-\nabla \cdot \rho \mathbf{F} \approx -\frac{1}{V} \int_V \nabla \cdot \rho \mathbf{F} dV \quad (23.36)$$

$$= -\frac{1}{V} \oint_S \rho \mathbf{F} \cdot \hat{\mathbf{n}} dS, \quad (23.37)$$

where the first integral is over the volume of a single tracer cells, and the second is over the faces of the tracer cell, via the divergence theorem.

The above equation gives us a starting point for our discretization, however does not fully specify how best to compute the fluxes at each face. To guide our choice of flux discretization, we have two key requirements which must be satisfied:

- Tracer variance must not increase.
- There must be zero flux of locally referenced potential density.

Proving that these conditions are satisfied by a given finite volume discretization is not easy. Using a different formulation, based on functional calculus, a discretization can be developed which does meet the requirements. This new discretization can be shown to be equivalent to a finite volume discretization, which is how it is implemented.

23.3.1 Functional

The following discussion of the functional approach borrows heavily from Section 16.1 of Griffies (2004). Certain results from that work are stated here without proof.

We begin with the observation that the evolution of tracer variance can be written as

$$\mathcal{V}_t = \frac{2\rho_0}{M} \int dV \nabla T \cdot \mathbf{F}. \quad (23.38)$$

We next define the functional

$$\mathcal{F} = \int dV \mathcal{L} \quad (23.39)$$

$$= \frac{1}{2} \int dV \mathbf{F} \cdot \nabla T \quad (23.40)$$

$$= \frac{M}{2\rho_0} \mathcal{V}_t. \quad (23.41)$$

Therefore, if the functional is positive, then the tracer variance is non-increasing. Furthermore, we can write the neutral diffusion equation at a given point as

$$(\rho T)_{,t}^{\text{diff}}(\mathbf{x}) = -\nabla \cdot \rho \mathbf{F} \quad (23.42)$$

$$= (d\mathbf{x})^{-1} \frac{\partial \mathcal{F}}{\partial T(\mathbf{x})}. \quad (23.43)$$

As such, we would like to look at ways of discretizing the right hand side of these equation, while maintaining our key requirements. We begin by writing equation 23.43 in discrete form:

$$(d\mathbf{x})^{-1} \frac{\partial \mathcal{F}}{\partial T(\mathbf{x})} \approx \frac{1}{V_{i,j,k}} \frac{\partial \mathcal{F}}{\partial T_{i,j,k}} \quad (23.44)$$

If we consider for now the flux expressed in terms of the diffusion tensor, we can write the functional as

$$\mathcal{F} = -\frac{1}{2} \int \rho dV \nabla T \mathbf{K} \nabla T \quad (23.45)$$

$$= -\frac{1}{2} \int \rho dV A_R ((T_{,x} - T_{,z} S_x)^2 + (T_{,y} - T_{,z} S_y)^2) \quad (23.46)$$

$$= \mathcal{F}^{x-z} + \mathcal{F}^{y-z} \quad (23.47)$$

$$\mathcal{F}^{x-z} = -\frac{1}{2} \int \rho dV A_R (T_{,x} - T_{,z} S_x)^2. \quad (23.48)$$

We will focus on the \mathcal{F}^{x-z} term, on the understanding that the \mathcal{F}^{y-z} can be handled symmetrically. We now discretize the functional by breaking up each tracer cell into quarter cells and writing the continuous equation in discrete form.

$$\mathcal{F}^{x-z} = -\frac{1}{2} \sum_{i,j,k} \sum_{ip,kr} \rho_{i,j,k} V_{i,j,k}^{(ip,kr)} (A_R)_{i,j,k} \left((\delta_x T)_{i,j,k}^{(ip)} - (\delta_z T)_{i,j,k}^{(kr)} (S_x)_{i,j,k}^{(ip,kr)} \right)^2 \quad (23.49)$$

$$= -\frac{1}{2} \sum_{i,j,k} \sum_{ip,kr} C_{i,j,k}^{(ip,kr)} \left(L_{i,j,k}^{(ip,kr)} \right)^2 \quad (23.50)$$

where

$$C_{i,j,k}^{(ip,kr)} = \rho_{i,j,k} V_{i,j,k}^{(ip,kr)} (A_R)_{i,j,k} \quad (23.51)$$

$$L_{i,j,k}^{(ip,kr)} = (\delta_x T)_{i,j,k}^{(ip)} - (\delta_z T)_{i,j,k}^{(kr)} (S_x)_{i,j,k}^{(ip,kr)}. \quad (23.52)$$

It is worth noting that this discretization maintains the sign-definite nature of the functional. It is this property which ensures our tracer variance is non-increasing.

We now take the derivative of the functional with respect to the tracer. Care must be taken at this step with the grid indices. The functional is a single value, resulting from an integral over the entire domain.

We are now taking derivatives at each point. We use the indices (i', j', k') to represent the point at which the derivative is taken, and (i, j, k) in the summation for the functional.

$$\frac{\delta \mathcal{F}^{x-z}}{\delta T_{i',j',k'}} = - \sum_{i,j,k} \sum_{ip,kr} \left(C_{i,j,k}^{(ip,kr)} L_{i,j,k}^{(ip,kr)} \frac{\delta L_{i,j,k}^{(ip,kr)}}{\delta T_{i',j',k'}} \right) \quad (23.53)$$

$$= - \sum_{i,j,k} \sum_{ip,kr} \left(C L_{i,j,k}^{(ip,kr)} \frac{\delta L_{i,j,k}^{(ip,kr)}}{\delta T_{i',j',k'}} \right) \quad (23.54)$$

To make this simpler we will write

$$L_{i,j,k}^{(ip,kr)} = (\delta_x T)_{i,j,k}^{(ip)} - (\delta_z T)_{i,j,k}^{(kr)} (S_x)_{i,j,k}^{(ip,kr)} \quad (23.55)$$

$$= L_{i,j,k}^{x(ip)} + L_{i,j,k}^{z(ip,kr)} \quad (23.56)$$

and handle each term separately, so

$$\frac{\delta \mathcal{F}^{(x-z)_x}}{\delta T_{i',j',k'}} = - \sum_{i,j,k} \sum_{ip,kr} \left(C L_{i,j,k}^{(ip,kr)} \frac{\delta L_{i,j,k}^{x(ip)}}{\delta T_{i',j',k'}} \right) \quad (23.57)$$

$$= - \sum_{i,j,k} \sum_{ip,kr} \left(C L_{i,j,k}^{(ip,kr)} \frac{\delta (\delta_x T)_{i,j,k}^{(ip)}}{\delta T_{i',j',k'}} \right) \quad (23.58)$$

$$= - \sum_{kr} \left[\frac{C L_{i'-1,j',k'}^{(1,kr)}}{(\Delta x)_{i'-1,j',k'}^{(1)}} + \frac{C L_{i',j',k'}^{(0,kr)}}{(\Delta x)_{i',j',k'}^{(0)}} \right] \\ - \sum_{kr} \left[- \frac{C L_{i',j',k'}^{(1,kr)}}{(\Delta x)_{i',j',k'}^{(1)}} - \frac{C L_{i'+1,j',k'}^{(0,kr)}}{(\Delta x)_{i'+1,j',k'}^{(0)}} \right] \quad (23.59)$$

$$= - (G_{i',j',k'}^E - G_{i'-1,j',k'}^E) \quad (23.60)$$

where

$$G_{i,j,k}^E = - \sum_{kr} \left(\frac{C L_{i,j,k}^{(1,kr)}}{(\Delta x)_{i,j,k}^{(1)}} + \frac{C L_{i+1,j,k}^{(0,kr)}}{(\Delta x)_{i+1,j,k}^{(0)}} \right). \quad (23.61)$$

We note that in equation 23.59 the global summation disappears, as only those points horizontally adjacent to (or on) (i', j', k') have non-zero derivatives. Likewise,

$$\frac{\delta \mathcal{F}^{(x-z)_z}}{\delta T_{i',j',k'}} = - \sum_{i,j,k} \sum_{ip,kr} \left(C L_{i,j,k}^{(ip,kr)} \frac{\delta L_{i,j,k}^{z(ip,kr)}}{\delta T_{i',j',k'}} \right) \quad (23.62)$$

$$= \sum_{i,j,k} \sum_{ip,kr} \left(C L_{i,j,k}^{(ip,kr)} (S_x)_{i,j,k}^{(ip,kr)} \frac{\delta (\delta_z T)_{i,j,k}^{(kr)}}{\delta T_{i',j',k'}} \right) \quad (23.63)$$

$$= \sum_{ip} \left[\frac{C L_{i',j',k'-1}^{(ip,1)} (S_x)_{i',j',k'-1}^{(ip,1)}}{(\Delta z)_{i',j',k'-1}^{(1)}} + \frac{C L_{i',j',k'}^{(ip,0)} (S_x)_{i',j',k'}^{(ip,0)}}{(\Delta z)_{i',j',k'}^{(0)}} \right] \quad (23.64)$$

$$+ \sum_{ip} \left[- \frac{C L_{i',j',k'}^{(ip,1)} (S_x)_{i',j',k'}^{(ip,1)}}{(\Delta z)_{i',j',k'}^{(1)}} - \frac{C L_{i',j',k'+1}^{(ip,0)} (S_x)_{i',j',k'+1}^{(ip,0)}}{(\Delta z)_{i',j',k'+1}^{(0)}} \right] \quad (23.65)$$

$$= - (G_{i',j',k'}^{b(x)} - G_{i',j',k'-1}^{b(x)}). \quad (23.66)$$

where

$$G_{i,j,k}^{b(x)} = - \sum_{ip} \left(\frac{CL_{i,j,k}^{(ip,1)}(S_x)_{i,j,k}^{(ip,1)}}{(\Delta z)_{i,j,k}^{(1)}} + \frac{CL_{i,j,k+1}^{(ip,0)}(S_x)_{i,j,k+1}^{(ip,0)}}{(\Delta z)_{i,j,k+1}^{(0)}} \right). \quad (23.67)$$

Invoking the symmetry mentioned above, we can now write our final discretization as

$$\frac{1}{V_{i,j,k}} \frac{\delta \mathcal{F}}{\delta T_{i,j,k}} = \frac{1}{V_{i,j,k}} \frac{\delta}{\delta T_{i,j,k}} \left(\mathcal{F}^{(x-z)_x} + \mathcal{F}^{(y-z)_y} + \mathcal{F}^{(x-z)_z} + \mathcal{F}^{(y-z)_z} \right) \quad (23.68)$$

$$= - \frac{1}{V_{i,j,k}} \left((G_{i,j,k}^E - G_{i-1,j,k}^E) + (G_{i,j,k}^N - G_{i,j-1,k}^N) + (G_{i,j,k}^{b(x)} - G_{i,j,k-1}^{b(x)}) + (G_{i,j,k}^{b(y)} - G_{i,j,k-1}^{b(y)}) \right) \quad (23.69)$$

$$= - \frac{1}{V_{i,j,k}} \left((G_{i,j,k}^E - G_{i-1,j,k}^E) + (G_{i,j,k}^N - G_{i,j-1,k}^N) + (G_{i,j,k}^b - G_{i,j,k-1}^b) \right) \quad (23.70)$$

We notice that we have a G term defined at each face of the tracer cell. Each G term has the dimension of a density times a flux times a surface area. We have thus derived exactly the finite volume flux formulation we were after, however by deriving it from the functional approach, we can ensure our requirements are satisfied.

23.3.2 Fluxes

We now have a discretization for our fluxes, which we can look at in detail. We begin with the x flux.

$$G_{i,j,k}^E = - \sum_{kr} \left(\frac{CL_{i,j,k}^{(1,kr)}}{(\Delta x)_{i,j,k}^{(1)}} + \frac{CL_{i+1,j,k}^{(0,kr)}}{(\Delta x)_{i+1,j,k}^{(0)}} \right) \quad (23.71)$$

$$= - \sum_{ip,kr} \left(\frac{CL_{i,j,k}^{E(ip,kr)}}{(\Delta x)_{i,j,k}^E} \right) \quad (23.72)$$

$$= - \frac{1}{(\Delta x)_{i,j,k}^E} \sum_{ip,kr} \rho_{i,j,k}^{E(ip,kr)} V_{i,j,k}^{E(ip,kr)} (A_R)_{i,j,k}^{E(ip,kr)} \left((\delta_x T)_{i,j,k}^E + (\delta_z T)_{i,j,k}^{E(ip,kr)} (S_x)_{i,j,k}^{E(ip,kr)} \right) \quad (23.73)$$

$$= - \frac{1}{(\Delta x)_{i,j,k}^E} \left[(\delta_x T)_{i,j,k}^E \sum_{ip,kr} (\rho V A_R)_{i,j,k}^{E(ip,kr)} + \sum_{ip,kr} (\rho V A_R)_{i,j,k}^{E(ip,kr)} (S_x)_{i,j,k}^{E(ip,kr)} (\delta_z T)_{i,j,k}^{E(ip,kr)} \right] \quad (23.74)$$

$$= - \frac{1}{(\Delta x)_{i,j,k}^E} \left[(\delta_x T)_{i,j,k}^E (\rho V A_R)_{i,j,k}^E + \sum_{ip,kr} (\rho V A_R S_x)_{i,j,k}^{E(ip,kr)} (\delta_z T)_{i,j,k}^{E(ip,kr)} \right]. \quad (23.75)$$

By symmetry we also have

$$G_{i,j,k}^N = - \frac{1}{(\Delta y)_{i,j,k}^N} \left[(\delta_y T)_{i,j,k}^N (\rho V A_R)_{i,j,k}^N + \sum_{jq,kr} (\rho V A_R S_y)_{i,j,k}^{N(jq,kr)} (\delta_z T)_{i,j,k}^{N(jq,kr)} \right]. \quad (23.76)$$

The x component of the z flux can be written

$$G_{i,j,k}^{b(x)} = - \sum_{ip} \left(\frac{CL_{i,j,k}^{(ip,1)} (S_x)_{i,j,k}^{(ip,1)}}{(\Delta z)_{i,j,k}^{(1)}} + \frac{CL_{i,j,k+1}^{(ip,0)} (S_x)_{i,j,k+1}^{(ip,0)}}{(\Delta z)_{i,j,k+1}^{(0)}} \right) \quad (23.77)$$

$$= - \sum_{ip,kr} \left(\frac{CL_{i,j,k}^{b(ip,kr)} (S_x)_{i,j,k}^{b(ip,kr)}}{(\Delta z)_{i,j,k}^b} \right) \quad (23.78)$$

$$= - \frac{1}{(\Delta z)_{i,j,k}^b} \sum_{ip,kr} \rho_{i,j,k}^{b(ip,kr)} V_{i,j,k}^{b(ip,kr)} (A_R)_{i,j,k}^{b(ip,kr)} \left((\delta_x T)_{i,j,k}^{b(ip,kr)} + (\delta_z T)_{i,j,k}^b (S_x)_{i,j,k}^{b(ip,kr)} \right) (S_x)_{i,j,k}^{b(ip,kr)} \quad (23.79)$$

$$= - \frac{1}{(\Delta z)_{i,j,k}^b} \sum_{ip,kr} (\rho V A_R)_{i,j,k}^{b(ip,kr)} \left((\delta_x T)_{i,j,k}^{b(ip,kr)} (S_x)_{i,j,k}^{b(ip,kr)} + (\delta_z T)_{i,j,k}^b (S_x)_{i,j,k}^{b(ip,kr)} \right)^2 \quad (23.80)$$

$$= - \frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{ip,kr} (\rho V A_R S_x^2)_{i,j,k}^{b(ip,kr)} + \sum_{ip,kr} (\rho V A_R S_x)_{i,j,k}^{b(ip,kr)} (\delta_x T)_{i,j,k}^{b(ip,kr)} \right] \quad (23.81)$$

$$= - \frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{kr} (\rho V A_R S_x^2)_{i,j,k}^{b(kr)} + \sum_{ip,kr} (\rho V A_R S_x)_{i,j,k}^{b(ip,kr)} (\delta_x T)_{i,j,k}^{b(ip,kr)} \right]. \quad (23.82)$$

Again, symmetry gives

$$G_{i,j,k}^{b(y)} = - \frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{kr} (\rho V A_R S_y^2)_{i,j,k}^{b(kr)} + \sum_{jq,kr} (\rho V A_R S_y)_{i,j,k}^{b(jq,kr)} (\delta_y T)_{i,j,k}^{b(jq,kr)} \right]. \quad (23.83)$$

Combining the two z flux terms we get

$$G_{i,j,k}^b = G_{i,j,k}^{b(x)} + G_{i,j,k}^{b(y)} \quad (23.84)$$

$$= - \frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{kr} (\rho V A_R S_x^2)_{i,j,k}^{b(kr)} + \sum_{ip,kr} (\rho V A_R S_x)_{i,j,k}^{b(ip,kr)} (\delta_x T)_{i,j,k}^{b(ip,kr)} \right. \\ \left. + (\delta_z T)_{i,j,k}^b \sum_{kr} (\rho V A_R S_y^2)_{i,j,k}^{b(kr)} + \sum_{jq,kr} (\rho V A_R S_y)_{i,j,k}^{b(jq,kr)} (\delta_y T)_{i,j,k}^{b(jq,kr)} \right] \quad (23.85)$$

$$= - \frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{kr} \left((\rho V A_R S_x^2)_{i,j,k}^{b(kr)} + (\rho V A_R S_y^2)_{i,j,k}^{b(kr)} \right) \right. \\ \left. + \sum_{ip,kr} (\rho V A_R S_x)_{i,j,k}^{b(ip,kr)} (\delta_x T)_{i,j,k}^{b(ip,kr)} + \sum_{jq,kr} (\rho V A_R S_y)_{i,j,k}^{b(jq,kr)} (\delta_y T)_{i,j,k}^{b(jq,kr)} \right] \quad (23.86)$$

$$= - \frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{kr} (\rho V A_R S^2)_{i,j,k}^{b(kr)} \right. \\ \left. + \sum_{ip,kr} (\rho V A_R S_x)_{i,j,k}^{b(ip,kr)} (\delta_x T)_{i,j,k}^{b(ip,kr)} + \sum_{jq,kr} (\rho V A_R S_y)_{i,j,k}^{b(jq,kr)} (\delta_y T)_{i,j,k}^{b(jq,kr)} \right] \quad (23.87)$$

Summarising all fluxes, we have

$$G_{i,j,k}^E = -\frac{1}{(\Delta x)_{i,j,k}^E} \left[(\delta_x T)_{i,j,k}^E (\rho V A_R)^E_{i,j,k} + \sum_{ip,kr} (\rho V A_R S_x)^{E(ip,kr)}_{i,j,k} (\delta_z T)_{i,j,k}^{E(ip,kr)} \right] \quad (23.88)$$

$$G_{i,j,k}^N = -\frac{1}{(\Delta y)_{i,j,k}^N} \left[(\delta_y T)_{i,j,k}^N (\rho V A_R)^N_{i,j,k} + \sum_{jq,kr} (\rho V A_R S_y)^{N(jq,kr)}_{i,j,k} (\delta_z T)_{i,j,k}^{N(jq,kr)} \right] \quad (23.89)$$

$$G_{i,j,k}^b = -\frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{kr} (\rho V A_R S^2)^{b(kr)}_{i,j,k} + \sum_{ip,kr} (\rho V A_R S_x)^{b(ip,kr)}_{i,j,k} (\delta_x T)_{i,j,k}^{b(ip,kr)} + \sum_{jq,kr} (\rho V A_R S_y)^{b(jq,kr)}_{i,j,k} (\delta_y T)_{i,j,k}^{b(jq,kr)} \right]. \quad (23.90)$$

Putting all these results together gives us the final discretized density weighted tracer tendency due to diffusion,

$$(dz(\rho T)_{,t}^{\text{diff}})_{i,j,k} = dz_{i,j,k} \frac{1}{V_{i,j,k}} \frac{\partial \mathcal{F}}{\partial T_{i,j,k}} \quad (23.91)$$

$$= \frac{1}{A_{i,j,k}^{x-y}} \left((G_{i,j,k}^E - G_{i-1,j,k}^E) + (G_{i,j,k}^N - G_{i,j-1,k}^N) + (G_{i,j,k}^b - G_{i,j,k-1}^b) \right). \quad (23.92)$$

23.3.3 Skew flux discretization

We now need to discretize the skew flux term. This can be done by appealing to analogy with the diffusive term. To distinguish between diffusive and skew terms we will employ d and s superscripts respectively. We will look to write the skew-flux term as

$$(dz(\rho T)_{,t}^{\text{skew}})_{i,j,k} = \frac{1}{A_{i,j,k}^{x-y}} \left(((G^s)_{i,j,k}^E - (G^s)_{i-1,j,k}^E) + ((G^s)_{i,j,k}^N - (G^s)_{i,j-1,k}^N) + ((G^s)_{i,j,k}^b - (G^s)_{i,j,k-1}^b) \right). \quad (23.93)$$

The diffusive flux terms can be written with respect to the components of the diffusion tensor \mathbf{K} ,

$$(G^d)_{i,j,k}^E = -\frac{1}{(\Delta x)_{i,j,k}^E} \left[(\delta_x T)_{i,j,k}^E (\rho V K_{11})_{i,j,k}^E + \sum_{ip,kr} (\rho V K_{13})_{i,j,k}^{E(ip,kr)} (\delta_z T)_{i,j,k}^{E(ip,kr)} \right] \quad (23.94)$$

$$(G^d)_{i,j,k}^N = -\frac{1}{(\Delta y)_{i,j,k}^N} \left[(\delta_y T)_{i,j,k}^N (\rho V K_{22})_{i,j,k}^N + \sum_{jq,kr} (\rho V K_{23})_{i,j,k}^{N(jq,kr)} (\delta_z T)_{i,j,k}^{N(jq,kr)} \right] \quad (23.95)$$

$$(G^d)_{i,j,k}^b = -\frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T)_{i,j,k}^b \sum_{kr} (\rho V K_{33})_{i,j,k}^{b(kr)} + \sum_{ip,kr} (\rho V K_{31})_{i,j,k}^{b(ip,kr)} (\delta_x T)_{i,j,k}^{b(ip,kr)} + \sum_{jq,kr} (\rho V K_{32})_{i,j,k}^{b(jq,kr)} (\delta_y T)_{i,j,k}^{b(jq,kr)} \right]. \quad (23.96)$$

Since the skew diffusion tensor only has the 13, 23, 31 and 32 terms as non-zero, the skew flux terms can be written as

$$(G^s)^E_{i,j,k} = -\frac{1}{(\Delta x)^E_{i,j,k}} \left[\sum_{ip,kr} (\rho V A_{13})^{E(ip,kr)}_{i,j,k} (\delta_z T)^{E(ip,kr)}_{i,j,k} \right] \quad (23.97)$$

$$(G^s)^N_{i,j,k} = -\frac{1}{(\Delta y)^N_{i,j,k}} \left[\sum_{jq,kr} (\rho V A_{23})^{N(jq,kr)}_{i,j,k} (\delta_z T)^{N(jq,kr)}_{i,j,k} \right] \quad (23.98)$$

$$(G^s)^b_{i,j,k} = -\frac{1}{(\Delta z)^b_{i,j,k}} \left[\sum_{ip,kr} (\rho V A_{31})^{b(ip,kr)}_{i,j,k} (\delta_x T)^{b(ip,kr)}_{i,j,k} + \sum_{jq,kr} (\rho V A_{32})^{b(jq,kr)}_{i,j,k} (\delta_y T)^{b(jq,kr)}_{i,j,k} \right], \quad (23.99)$$

or in terms of Υ ,

$$(G^s)^E_{i,j,k} = -\frac{1}{(\Delta x)^E_{i,j,k}} \left[\sum_{ip,kr} (\rho V \Upsilon_x)^{E(ip,kr)}_{i,j,k} (\delta_z T)^{E(ip,kr)}_{i,j,k} \right] \quad (23.100)$$

$$(G^s)^N_{i,j,k} = -\frac{1}{(\Delta y)^N_{i,j,k}} \left[\sum_{jq,kr} (\rho V \Upsilon_y)^{N(jq,kr)}_{i,j,k} (\delta_z T)^{N(jq,kr)}_{i,j,k} \right] \quad (23.101)$$

$$(G^s)^b_{i,j,k} = \frac{1}{(\Delta z)^b_{i,j,k}} \left[\sum_{ip,kr} (\rho V \Upsilon_x)^{b(ip,kr)}_{i,j,k} (\delta_x T)^{b(ip,kr)}_{i,j,k} + \sum_{jq,kr} (\rho V \Upsilon_y)^{b(jq,kr)}_{i,j,k} (\delta_y T)^{b(jq,kr)}_{i,j,k} \right], \quad (23.102)$$

We have now derived the discrete formulation which forms the basis for the neutral physics modules within MOM, based on the parameterizations of Redi and GM. The discretization does not yet include details of how each of the terms in the given equations are computed. Various implementation details need to be discussed to gain a full understanding of the handling of neutral physics within MOM.

23.4 Implementation

Having derived a discretization, we now turn to the implementation of this formulation within MOM. The implementation is most easily understood from a top-down perspective, starting with the final result, and progressively working down into the finer details.

The subroutine `neutral_physics_new` computes the value of $dz(\rho T_n)_{i,t}^{\text{neutral}}$ for each tracer T_n . This value is stored in the array `total_th_tendency(:, :, :, n)`. Once calculated, this value is then added to `T_prog(n)%th_tendency`.

23.4.1 Fluxes

The value of the tracer tendencies are computed in the subroutine `flux_calculations`, in the module `ocean.nphysics_flux.F90`.

23.4.1.1

Within `flux_calculations`, the subroutine `update_tendencies` performs the calculations

$$\begin{aligned} (dz(\rho T_n)_{,t}^{\text{diff}})_{i,j,k} &= \frac{1}{A_{i,j,k}^{x-y}} \left(((G_n^d)^E_{i,j,k} - (G_n^d)^E_{i-1,j,k}) \right. \\ &\quad \left. + ((G_n^d)^N_{i,j,k} - (G_n^d)^N_{i,j-1,k}) + ((G_n^d)^b_{i,j,k} - (G_n^d)^b_{i,j,k-1}) \right) \end{aligned} \quad (23.103)$$

$$\begin{aligned} (dz(\rho T_n)_{,t}^{\text{skew}})_{i,j,k} &= \frac{1}{A_{i,j,k}^{x-y}} \left(((G_n^s)^E_{i,j,k} - (G_n^s)^E_{i-1,j,k}) \right. \\ &\quad \left. + ((G_n^s)^N_{i,j,k} - (G_n^s)^N_{i,j-1,k}) + ((G_n^s)^b_{i,j,k} - (G_n^s)^b_{i,j,k-1}) \right) \end{aligned} \quad (23.104)$$

$$(dz(\rho T_n)_{,t}^{\text{neutral}})_{i,j,k} = (dz(\rho T_n)_{,t}^{\text{diff}})_{i,j,k} + (dz(\rho T_n)_{,t}^{\text{skew}})_{i,j,k}. \quad (23.105)$$

The diffusive and skew-flux tendencies for each tracer can be diagnosed by the diagnostics `diff_th_tendency(n)` and `skew_th_tendency(n)` respectively.

23.4.1.2

The flux terms are computed in the subroutine `compute_fluxes`. These are computed according to the equations

$$(G_n^d)^E_{i,j,k} = -\frac{1}{(\Delta x)_{i,j,k}^E} \left[(\delta_x T_n)^E_{i,j,k} (\rho V K_{11})_{i,j,k}^E + \sum_{ip,kr} (\rho V K_{13})_{i,j,k}^{E(ip,kr)} (\delta_z T_n)^{E(ip,kr)}_{i,j,k} \right] \quad (23.106)$$

$$(G_n^d)^N_{i,j,k} = -\frac{1}{(\Delta y)_{i,j,k}^N} \left[(\delta_y T_n)^N_{i,j,k} (\rho V K_{22})_{i,j,k}^N + \sum_{jq,kr} (\rho V K_{23})_{i,j,k}^{N(jq,kr)} (\delta_z T_n)^{N(jq,kr)}_{i,j,k} \right] \quad (23.107)$$

$$\begin{aligned} (G_n^d)^b_{i,j,k} &= -\frac{1}{(\Delta z)_{i,j,k}^b} \left[(\delta_z T_n)^b_{i,j,k} (\rho V K_{33}^{\text{exp}})_{i,j,k}^b \right. \\ &\quad \left. + \sum_{ip,kr} (\rho V K_{31})_{i,j,k}^{b(ip,kr)} (\delta_x T_n)^{b(ip,kr)}_{i,j,k} + \sum_{jq,kr} (\rho V K_{32})_{i,j,k}^{b(jq,kr)} (\delta_y T_n)^{b(jq,kr)}_{i,j,k} \right] \end{aligned} \quad (23.108)$$

for the diffusive flux, and

$$(G_n^s)^E_{i,j,k} = -\frac{1}{(\Delta x)_{i,j,k}^E} \left[\sum_{ip,kr} (\rho V \Upsilon_x)_{i,j,k}^{E(ip,kr)} (\delta_z T_n)^{E(ip,kr)}_{i,j,k} \right] \quad (23.109)$$

$$(G_n^s)^N_{i,j,k} = -\frac{1}{(\Delta y)_{i,j,k}^N} \left[\sum_{jq,kr} (\rho V \Upsilon_y)_{i,j,k}^{N(jq,kr)} (\delta_z T_n)^{N(jq,kr)}_{i,j,k} \right] \quad (23.110)$$

$$(G_n^s)^b_{i,j,k} = \frac{1}{(\Delta z)_{i,j,k}^b} \left[\sum_{ip,kr} (\rho V \Upsilon_x)_{i,j,k}^{b(ip,kr)} (\delta_x T_n)^{b(ip,kr)}_{i,j,k} + \sum_{jq,kr} (\rho V \Upsilon_y)_{i,j,k}^{b(jq,kr)} (\delta_y T_n)^{b(jq,kr)}_{i,j,k} \right] \quad (23.111)$$

for the skew flux.

23.4.1.3 Tracer Limiting

In regions of large tracer gradient, it may be desirable to resort to a purely horizontal diffusion, with no vertical or skew terms. If the namelist parameter `neutral_physics_limit == .true` then the following procedure is applied for each each tracer. At those points where the mask `T_prog(n)%tmask_limit(i,j,k)`

== 1.0, the following transformations are applied,

$$(G_n^d)_{i,j,k}^E \rightarrow -\frac{1}{\Delta x_{i,j}^E} (\rho V K_{11})_{i,j,k}^E (\delta_x T_n)_{i,j,k}^E \quad (23.112)$$

$$(G_n^d)_{i,j,k}^N \rightarrow -\frac{1}{\Delta y_{i,j}^N} (\rho V K_{22})_{i,j,k}^N (\delta_y T_n)_{i,j,k}^N \quad (23.113)$$

$$(G_n^d)_{i,j,k}^b \rightarrow 0.0 \quad (23.114)$$

$$(G_n^s)_{i,j,k}^E \rightarrow 0.0 \quad (23.115)$$

$$(G_n^s)_{i,j,k}^N \rightarrow 0.0 \quad (23.116)$$

$$(G_n^s)_{i,j,k}^b \rightarrow 0.0 \quad (23.117)$$

$$(\rho K_{33}^{\text{imp}})_{i,j,k}^n \rightarrow 0.0. \quad (23.118)$$

The calculation of the mask is performed in the module `ocean_tracer`, while the transformation described above is performed in the subroutine `apply_tracer_limits`.

After the tracer limiting process has been applied, the fluxes are diagnosed. The quantities diagnosed include a minus sign, i.e. $-(G_n^d)_{i,j,k}^E$, to adhere to a legacy convention within MOM. The diagnostics used are

```
diff_flux_x_xte(n)
diff_flux_y_ytn(n)
diff_flux_z_ztb(n)
skew_flux_x_xte(n)
skew_flux_y_ytn(n)
skew_flux_z_ztb(n).
```

23.4.1.4

The terms incorporating the tensor components (e.g. $(\rho V K_{11})$) have units of mass times diffusivity. These terms are calculated in the subroutine `compute_mass_diff`. The diffusive terms are calculated as

$$(\rho V K_{11})_{i,j,k}^E = \sum_{ip,kr} C2H \left((\rho V_x)_{i,j,k}^{(ip,kr)} (A_R)_{i,j,k} \right) \quad (23.119)$$

$$(\rho V K_{22})_{i,j,k}^N = \sum_{jq,kr} C2H \left((\rho V_y)_{i,j,k}^{(jq,kr)} (A_R)_{i,j,k} \right) \quad (23.120)$$

$$(\rho V K_{13})_{i,j,k}^{E(ip,kr)} = C2H \left((\rho V K_{13})_{i,j,k}^{(ip,kr)} \right) \quad (23.121)$$

$$(\rho V K_{23})_{i,j,k}^{N(jq,kr)} = C2H \left((\rho V K_{23})_{i,j,k}^{(jq,kr)} \right) \quad (23.122)$$

$$(\rho V K_{31})_{i,j,k}^{b(ip,kr)} = C2V \left((\rho V K_{13})_{i,j,k}^{(ip,kr)} \right) \quad (23.123)$$

$$(\rho V K_{32})_{i,j,k}^{b(jq,kr)} = C2V \left((\rho V K_{23})_{i,j,k}^{(jq,kr)} \right) \quad (23.124)$$

$$(\rho V K_{13})_{i,j,k}^{(ip,kr)} = (\rho V_x)_{i,j,k}^{(ip,kr)} (K_{13})_{i,j,k}^{(ip,kr)} \quad (23.125)$$

$$(\rho V K_{23})_{i,j,k}^{(jq,kr)} = (\rho V_y)_{i,j,k}^{(jq,kr)} (K_{23})_{i,j,k}^{(jq,kr)} \quad (23.126)$$

$$(\rho V K_{33})_{i,j,k}^{(kr)} = \sum_{ip} (\rho V_x)_{i,j,k}^{(ip,kr)} (K_{33x})_{i,j,k}^{(ip,kr)} + \sum_{jq} (\rho V_y)_{i,j,k}^{(jq,kr)} (K_{33y})_{i,j,k}^{(jq,kr)}. \quad (23.127)$$

The skew terms are calculated as

$$(\rho V \Upsilon_x)_{i,j,k}^{E(ip,kr)} = C2H \left((\rho V \Upsilon_x)_{i,j,k}^{(ip,kr)} \right) \quad (23.128)$$

$$(\rho V \Upsilon_y)_{i,j,k}^{N(jq,kr)} = C2H \left((\rho V \Upsilon_y)_{i,j,k}^{(jq,kr)} \right) \quad (23.129)$$

$$(\rho V \Upsilon_x)_{i,j,k}^{b(ip,kr)} = C2V \left((\rho V \Upsilon_x)_{i,j,k}^{(ip,kr)} \right) \quad (23.130)$$

$$(\rho V \Upsilon_y)_{i,j,k}^{b(jq,kr)} = C2V \left((\rho V \Upsilon_y)_{i,j,k}^{(jq,kr)} \right) \quad (23.131)$$

$$(\rho V \Upsilon_x)_{i,j,k}^{(ip,kr)} = (\rho V_x)_{i,j,k}^{(ip,kr)} (\Upsilon_x)_{i,j,k}^{(ip,kr)} \quad (23.132)$$

$$(\rho V \Upsilon_y)_{i,j,k}^{(jq,kr)} = (\rho V_y)_{i,j,k}^{(jq,kr)} (\Upsilon_y)_{i,j,k}^{(jq,kr)}. \quad (23.133)$$

Three dimensional versions of each of these quantities are diagnosed. The diagnostics `symm_mass_diff_11_xte` and `symm_mass_diff_22_ytn` diagnose the quantities in equations 23.119 and 23.120. The diagnostics

`symm_mass_diff_13_h_xz`

`symm_mass_diff_23_h_yz`

`symm_mass_diff_31_v_xz`

`symm_mass_diff_32_v_yz`

diagnose the terms

$$\sum_{ip,kr} (\rho V K_{13})_{i,j,k}^{E(ip,kr)} \quad (23.134)$$

$$\sum_{jq,kr} (\rho V K_{23})_{i,j,k}^{N(jq,kr)} \quad (23.135)$$

$$\sum_{ip,kr} (\rho V K_{31})_{i,j,k}^{b(ip,kr)} \quad (23.136)$$

$$\sum_{jq,kr} (\rho V K_{32})_{i,j,k}^{b(jq,kr)} \quad (23.137)$$

respectively. Likewise, the diagnostics

`symm_mass_skew_13_h_xz`

`symm_mass_skew_23_h_yz`

`symm_mass_skew_31_v_xz`

`symm_mass_skew_32_v_yz`

diagnose the terms

$$\sum_{ip,kr} (\rho V \Upsilon_x)_{i,j,k}^{E(ip,kr)} \quad (23.138)$$

$$\sum_{jq,kr} (\rho V \Upsilon_y)_{i,j,k}^{N(jq,kr)} \quad (23.139)$$

$$\sum_{ip,kr} (\rho V \Upsilon_x)_{i,j,k}^{b(ip,kr)} \quad (23.140)$$

$$\sum_{jq,kr} (\rho V \Upsilon_y)_{i,j,k}^{b(jq,kr)}. \quad (23.141)$$

23.4.1.5

The $\rho V K_{33}$ term must be handled carefully. As discussed in Griffies (2004) Section 16.8.3, the vertical diffusion from the 33-term is broken into an explicit and implicit component (Griffies (2004) Eqn 16.104),

$$K_{33} = K_{33}^{\text{exp}} + K_{33}^{\text{imp}}. \quad (23.142)$$

The diffusivity is handled explicitly up to a critical threshold, above which the remainder is treated implicitly. The critical diffusivity is given by (Griffies (2004) Eqn 16.105)

$$K_{33}^{\text{crit}} = \frac{(\Delta z)^2}{2\Delta t_{\text{forward}}} \quad (23.143)$$

The implementation of this decomposition is done in terms of mass weighted diffusivities, leading to a critical value of

$$(\rho V K_{33}^{\text{crit}})_{i,j,k}^{(kr)} = \frac{(\mu_k^{(kr)} \Delta z_{i,j,k})^2}{2\Delta \tau} \left(\mu_k^{(kr)} (\rho \Delta z)_{i,j,k} \Delta A_{i,j} \right). \quad (23.144)$$

The explicit component is then defined as

$$(\rho V K_{33}^{\text{exp}})_{i,j,k}^{(kr)} = \begin{cases} (\rho V K_{33}^{\text{crit}})_{i,j,k}^{(kr)} & \text{if } (\rho V K_{33})_{i,j,k}^{(kr)} > (\rho V K_{33}^{\text{crit}})_{i,j,k}^{(kr)} \\ (\rho V K_{33})_{i,j,k}^{(kr)} & \text{otherwise.} \end{cases} \quad (23.145)$$

The implicit component is required at the tracer cell point, and so is given by

$$(\rho V K_{33}^{\text{imp}})_{i,j,k} = \sum_{kr} \left((\rho V K_{33})_{i,j,k}^{(kr)} - (\rho V K_{33}^{\text{exp}})_{i,j,k}^{(kr)} \right). \quad (23.146)$$

Each tracer carries this value (as later steps apply tracer specific masks), with the volume term removed, giving

$$(\rho K_{33}^{\text{imp}})_{i,j,k}^n = \frac{(\rho V K_{33}^{\text{imp}})_{i,j,k}}{\Delta z_{i,j,k} \Delta A_{i,j}}. \quad (23.147)$$

This value is subsequently used by the module `ocean_vert_mix` to calculate the implicit component of vertical diffusion.

Finally, the explicit component is calculated at the bottom of the tracer cell,

$$(\rho V K_{33}^{\text{exp}})_b^{i,j,k} = (\rho V K_{33}^{\text{exp}})_{i,j,k}^{(1)} + (\rho V K_{33}^{\text{exp}})_{i,j,k+1}^{(0)}. \quad (23.148)$$

There is a namelist parameter which can be used to turn off the above decomposition into explicit and implicit parts. If `diffusion_all_explicit == .true.`, then Equation (23.145) becomes

$$(\rho V K_{33}^{\text{exp}})_{i,j,k}^{(kr)} = (\rho V K_{33})_{i,j,k}^{(kr)}, \quad (23.149)$$

which ensures that the implicit component is zero. These calculations are performed in the subroutine `compute_33_terms`. The values of ρK_{33}^{imp} and $(\rho V K_{33}^{\text{exp}})_b$ are diagnosed via the names `m33_implicit` and `m33_explicit_ztb`.

23.4.1.6 Density Weighted Quarter-cell Volumes

The geometric factor required is a density weighted volume for the quarter-cell associated with the triad over which the diffusivity is defined. In the direction perpendicular to the triad plane the tracer cell width is taken, while in the triad plane half cell widths are taken. For example, in the $x - z$ plane, the density weighted quarter-cell volume would be the product of the density, the width of the cell in the y direction and the half widths in the x and z directions.

An issue arises here in that the vertical thickness of adjacent cells may not be equal. As such, the effective contact area of the two cells may not be equal when looked at from either side. It stands to reason that the diffusive flux from each side of the cell faces should be calculated through the same area from each side. As such, the minimum half-cell thickness between adjacent tracer cells is used in the calculation of quarter-cell volumes. Further justification for this choice can be found in [Griffies \(2004\)](#) Chapter 16.4.3.

The half-cell thicknesses at the eastern and northern faces are given by

$$(\Delta z')_{i,j,k}^{E(kr)} = \min\left(\mu_k^{(kr)} \Delta z_{i,j,k}, \mu_k^{(kr)} \Delta z_{i+1,j,k}\right) \quad (23.150)$$

$$(\Delta z')_{i,j,k}^{N(kr)} = \min\left(\mu_k^{(kr)} \Delta z_{i,j,k}, \mu_k^{(kr)} \Delta z_{i,j+1,k}\right). \quad (23.151)$$

This leads to density weighted quarter-cell thicknesses defined as

$$(\rho V_x)_{i,j,k}^{(ip,kr)} = \rho_{i,j,k} (\Delta z')_{i-1+ip,j,k}^{E(kr)} \Delta x_{i,j}^{(ip)} \Delta y_{i,j} \quad (23.152)$$

$$(\rho V_y)_{i,j,k}^{(jq,kr)} = \rho_{i,j,k} (\Delta z')_{i,j-1+jq,k}^{N(kr)} \Delta x_{i,j} \Delta y_{i,j}^{(jq)} \quad (23.153)$$

where¹

$$\rho_{i,j,k} = \frac{(\rho \Delta z)_{i,j,k}}{\Delta z_{i,j,k}}. \quad (23.154)$$

These calculations are performed in the subroutine `geometric_terms`.

23.5 Diffusion and Skew-Diffusion Tensors

We will now look at the parameterizations which are used to determine the values of the tensors K and A . As well as computing the expressions in equations (23.20) and (23.25), the parameterization must also account for regions of steep neutral slope, which generally, but not always, occur within boundary layers. In these regions, the basic equations require some modification due both to modifications of the physical processes, and the need to maintain numerical regularity. The following computations are carried out in the module `ocean_nphysics_tensor`.

23.5.1 Depth Taper

As discussed in [Griffies \(2004\)](#) Section 15.3.2, as the surface is approached, the transport due to neutral physics must be modified. To do this, a taper function T is calculated at each tracer point, which takes values between zero and one. This function is then multiplied by the appropriate components of the diffusion tensors to ensure that the total transport behaves in a physically sensible manner.

The tapering is applied in regions where the vertical displacement due to the undulating density surface is greater than the depth ([Griffies \(2004\)](#) Section 15.3.3). The vertical displacement is approximated by $\xi = \lambda_1 |S|$. As per [Large et al. \(1997\)](#), the Rossby radius used in this calculation is restricted to lie within the range $[\lambda_{\max}, \lambda_{\min}]$. This choice leads to the capped Rossby radius

$$\lambda'_1 = \max(\lambda_{\min}, \min(\lambda_{\max}, \lambda_1)). \quad (23.155)$$

The limits are controlled by the namelist parameters `rossby_radius_min` and `rossby_radius_max` and have default values of 15km and 100km respectively.

As well as considering the vertical displacement distance ξ , we also take into account the surface boundary layer thickness, D_{BL} , as calculated in the subroutine `vert_mix_coeff` in the `ocean_vert_mix` module. Finally, the namelist parameter `turb_blayer_min` provides a minimum depth over which to taper the

¹This computation is an implementation detail which eliminates the need for the `Density` variable in the neutral physics module.

neutral transport. Taking the deepest value of each of these depths, calculated at each tracer point, gives us the adjusted displacement depth

$$\xi'_{i,j,k} = \max\left((\lambda'_1)_{i,j} |\mathbf{S}|_{i,j,k}, (D_{BL})_{i,j}, \text{turb_blayer_min}\right). \quad (23.156)$$

In regions shallower than ξ' the taper goes as a sine function of the depth ratio r (Griffies (2004) equation (15.24))

$$T_{\text{sine}} = \frac{1}{2} \left(1 + \sin \left(\pi \left(r - \frac{1}{2} \right) \right) \right), \quad (23.157)$$

where (Griffies (2004) equation (15.22))

$$r = \frac{D}{\xi'}, \quad (23.158)$$

as depicted in Figure 23.3.

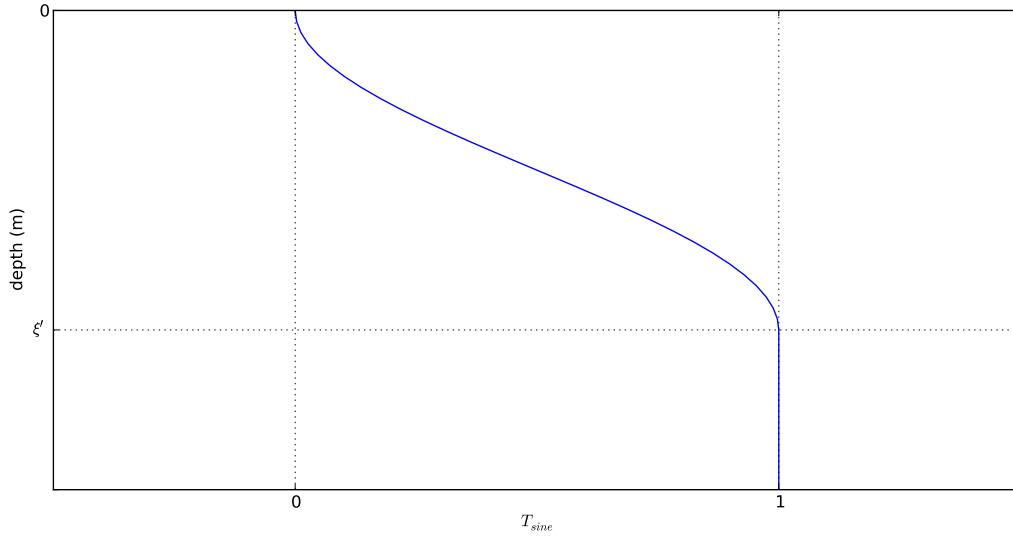


Figure 23.3: The sine taper function.

This function is discretized as

$$r_{i,j,k} = \frac{D_{i,j,k}}{\xi'_{i,j,k}} \quad (23.159)$$

$$(T_{\text{sine}})_{i,j,k} = \begin{cases} \frac{1}{2} \left(1 + \sin \left(\pi \left(r_{i,j,k} - \frac{1}{2} \right) \right) \right) & \text{if } r_{i,j,k} \leq 1 \\ 1 & \text{otherwise,} \end{cases} \quad (23.160)$$

where $(T_{\text{sine}})_{i,j,k}$ is evaluated at the tracer points. This taper function is calculated in the subroutine `sine_taper` and is used in calculating both the diffusion tensor and the skew tensor. The depth taper can be diagnosed via the name `depth_taper` while the adjusted displacement depth can be diagnosed via the name `displacement_depth`. The tapering is used by default, however it can be turned off (i.e. T_{sine} set to 1 everywhere) by setting the namelist parameter `neutral_sine_taper` to `.false..`

23.5.2 Neutral Diffusion Tensor

The tensor used to parameterize neutral diffusion is the small slope approximation of the Redi Diffusion Tensor (Redi (1982), Gent and McWilliams (1990), Griffies (2004) equation (14.19)) with the ϵ term dropped

from the (3,3) component,

$$K = A_R \begin{pmatrix} 1 & 0 & S_x \\ 0 & 1 & S_y \\ S_x & S_y & S_x^2 + S_y^2 \end{pmatrix} \quad (23.161)$$

To maintain physical and numerical consistency, the components of this tensor must be tapered near the surface (as per Section 23.5.1) and also in regions of steep neutral slope. A neutral taper function T_N is calculated as a combination of T_{sine} and another taper function based on neutral slope. Those components involving a slope term are multiplied by the neutral taper T_N . This leads to the tensor

$$K_R = A_R \begin{pmatrix} 1 & 0 & T_N S_x \\ 0 & 1 & T_N S_y \\ T_N S_x & T_N S_y & T_N (S_x^2 + S_y^2) \end{pmatrix}. \quad (23.162)$$

Taking note of the symmetry of the system, the components of the matrix can be written as

$$K = \begin{pmatrix} A_R & 0 & K_{13} \\ 0 & A_R & K_{23} \\ K_{13} & K_{23} & K_{33x} + K_{33y} \end{pmatrix}, \quad (23.163)$$

which leads to the discrete equations

$$(K_{13})_{i,j,k}^{(ip,kr)} = (A_R)_{i,j,k} (T_N)_{i,j,k}^{(kr)} (S_x)_{i,j,k}^{(ip,kr)} \quad (23.164)$$

$$(K_{23})_{i,j,k}^{(jq,kr)} = (A_R)_{i,j,k} (T_N)_{i,j,k}^{(kr)} (S_y)_{i,j,k}^{(jq,kr)} \quad (23.165)$$

$$(K_{33x})_{i,j,k}^{(ip,kr)} = (A_R)_{i,j,k} (T_N)_{i,j,k}^{(kr)} \left((S_x)_{i,j,k}^{(ip,kr)} \right)^2 \quad (23.166)$$

$$(K_{33y})_{i,j,k}^{(jq,kr)} = (A_R)_{i,j,k} (T_N)_{i,j,k}^{(kr)} \left((S_y)_{i,j,k}^{(jq,kr)} \right)^2. \quad (23.167)$$

The components of this tensor are computed in the subroutine `diffusion_tensor`. The tensor values are diagnosed by averaging over the triads, and are saved via the diagnostic table entries `symm_tensor33x`, `symm_tensor33y`, `symm_tensor13` and `symm_tensor23`.

23.5.2.1 Tapers

The tapers used in the above calculations must take into account both regions near the surface and also regions of steep neutral slope (Griffies (2004) Section 15.1). Section 23.5.1 described the algorithm for tapering based on depth. We now look at the slope based taper.

Exponential Tapering An exponential taper can be used to quickly and smoothly transition between 1.0 and 0.0 when the slope is above some critical value S_{\max} (Danabasoglu and McWilliams (1995) Eqn A.7a). The equation for the taper is given by

$$T_{\tanh} = \frac{1}{2} \left(1 + \tanh \left(\frac{S_{\max} - |S|}{\Delta S} \right) \right), \quad (23.168)$$

where ΔS takes the value of the namelist parameter `swidth` and represents the half-width of the transition region between the maximum and minimum values of the taper, as depicted in Figure 23.4. The discretization of this equation is

$$(T_{\tanh})_{i,j,k}^{(kr)} = \frac{1}{2} \left(1 + \tanh \left(\frac{S_{\max} - |S|_{i,j,k}^{(kr)}}{\Delta S} \right) \right). \quad (23.169)$$

As noted in Griffies (2004) Section 15.1.4, this equation can be applied at all points, rather than having to first test for those points above S_{\max} .

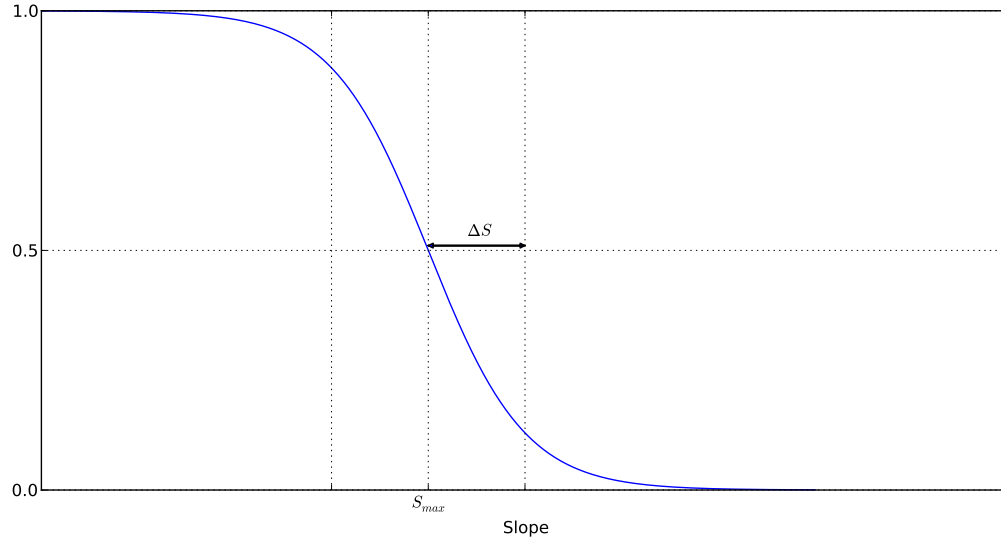


Figure 23.4: the tanh taper function.

Neutral taper The neutral tapering function uses the product of the depth-based taper T_{sine} and the slope-based taper T_{tanh} .

$$(T_N)_{i,j,k}^{(kr)} = (T_{\text{tanh}})_{i,j,k}^{(kr)} (T_{\text{sine}})_{i,j,k}. \quad (23.170)$$

23.5.3 Skew-Diffusion Tensor

The stirring of tracers by mesoscale eddies is modeled by an anti-symmetric skew tensor, which combines with the symmetric diffusion tensor to give the complete neutral transport tensor. The skew tensor can be expressed in terms of the eddy-induced transport, Υ , to give (Ferrari et al. (2010) equation (52))

$$A = \begin{pmatrix} 0 & 0 & \Upsilon_x \\ 0 & 0 & \Upsilon_y \\ -\Upsilon_x & -\Upsilon_y & 0 \end{pmatrix}. \quad (23.171)$$

The calculation of the transport, Υ , can be done in a number of ways. The module `ocean_nphysics_skew` provides three different methods, which are described below. Each method computes a discretization of Υ over the centred triad grouping, giving $\Upsilon = \left((\Upsilon_x)_{i,j,k}^{(ip,kr)}, (\Upsilon_y)_{i,j,k}^{(jq,kr)} \right)$. By averaging over the triads, these values are diagnosable via the names `upsilonx` and `upsilony`.

23.5.3.1 Preliminaries

The following quantities are used in the calculations below, and their discretizations are given here for completeness.

$$(\kappa S_x)_{i,j,k}^{(ip,kr)} = \kappa_{i,j,k} (S_x)_{i,j,k}^{(ip,kr)} \quad (23.172)$$

$$(\kappa S_y)_{i,j,k}^{(jq,kr)} = \kappa_{i,j,k} (S_y)_{i,j,k}^{(jq,kr)} \quad (23.173)$$

$$(N^2 \Delta z)_{i,j,k}^{(kr)} = (N^2)_{i,j,k}^{(kr)} (\Delta z)_{i,j,k}^{(kr)} \quad (23.174)$$

$$(N^2 S_x \kappa \Delta z)_{i,j,k}^{(ip,kr)} = (N^2 \Delta z)_{i,j,k}^{(kr)} (\kappa S_x)_{i,j,k}^{(ip,kr)} \quad (23.175)$$

$$(N^2 S_y \kappa \Delta z)_{i,j,k}^{(jq,kr)} = (N^2 \Delta z)_{i,j,k}^{(kr)} (\kappa S_y)_{i,j,k}^{(jq,kr)} \quad (23.176)$$

It should be noted that the slope used here relates the density surface to the horizontal surface of constant s , however the development of the GM tensor below is originally based on a slope vector relative to surfaces of constant geopotential. As discussed in Section 4.2.3, using the generalized vertical coordinate slope is usually a reasonable approximation to the geopotential slope, as long as the generalised vertical coordinate is relatively horizontal. In situations where this is not the case, such as sigma models, the approximation may lead to spurious results. As stated in Section 23.1, neutral physics in MOM has not been implemented for the terrain following vertical coordinate option.

23.5.3.2 GM Skew-Diffusion Tensor

The first technique for computing eddy-induced tracer-transport is based on the parameterization of Gent and McWilliams (1990) and Gent et al. (1995). The implementation is based largely on that described in Griffies (2004). This method is enabled by setting the namelist parameter `gm_transport = .true..`

The skew-diffusion tensor which corresponds to the GM parameterization of eddy-induced tracer transports is defined as (Griffies (1998) equation (14).)

$$A = \kappa \begin{pmatrix} 0 & 0 & -S_x \\ 0 & 0 & -S_y \\ S_x & S_y & 0 \end{pmatrix}, \quad (23.177)$$

which leads to the individual components being defined as (Ferrari et al. (2010) equation (9).)

$$\Upsilon^{GM} = -\kappa \mathbf{S}. \quad (23.178)$$

As discussed in Griffies (2004), Section 15.3.4.3, the tensor must be tapered to zero at the surface. For those points within the surface boundary layer, as defined in Section 23.7.6, a depth based linear tapering scheme is applied. This reduces $(\kappa \mathbf{S})$ from $(\kappa \mathbf{S})_{\max}$ at the base of the boundary layer to zero at the surface. The value of $(\kappa \mathbf{S})_{\max}$ is defined as

$$(\kappa \mathbf{S})_{\max} = \frac{S_{\max}}{|\mathbf{S}'|} (\kappa \mathbf{S})', \quad (23.179)$$

where primed variables indicate a value evaluated at the base of the boundary layer, i.e. at (i, j, k) for $k = k_{\text{surf},i,j}$ ². Including the linear taper, we obtain the following discretization for points (i, j, k) with $k \leq k_{\text{surf},i,j}$:

$$(\Upsilon_x)_{i,j,k}^{(ip,kr)} = -\frac{D_{i,j,k}}{D'_{i,j}} \frac{S_{\max}}{|\mathbf{S}'|_{i,j}} (\kappa S_x)'_{i,j} \quad (23.180)$$

$$(\Upsilon_y)_{i,j,k}^{(jq,kr)} = -\frac{D_{i,j,k}}{D'_{i,j}} \frac{S_{\max}}{|\mathbf{S}'|_{i,j}} (\kappa S_y)'_{i,j}. \quad (23.181)$$

In those regions below the boundary layer, the depth based sine taper described in Section 23.5.1 is applied, giving

$$(\Upsilon_x)_{i,j,k}^{(ip,kr)} = -(T_{\text{sine}})_{i,j,k} (\kappa S_x)_{i,j,k}^{(ip,kr)} \quad (23.182)$$

$$(\Upsilon_y)_{i,j,k}^{(jq,kr)} = -(T_{\text{sine}})_{i,j,k} (\kappa S_y)_{i,j,k}^{(jq,kr)}. \quad (23.183)$$

This calculation is performed in the subroutine `gm_tensor`. The value of D' is diagnosed via the name `depth_blayer_base` while the maximum slope vector is diagnosed via the names `agm_slope_x_max` and `agm_slope_y_max`.

²This definition ensures that $|(\kappa \mathbf{S})_{\max}| = \kappa' S_{\max}$ while maintaining the same orientation as \mathbf{S}' . This definition is based on an interpretation of the text in Sections 15.2 and 15.3.4.3 of Griffies (2004).

23.5.3.3 Baroclinic Modes

The second technique involves writing the GM transport as a sum of baroclinic modes and taking only the low mode-number terms. This method is described in Section 3.1 of [Ferrari et al. \(2010\)](#). This method requires that the diffusivity κ is depth independent, which is checked when the module is initially loaded, based on the namelist parameters selected in the diffusivity module.

The baroclinic modes are defined as ([Ferrari et al. \(2010\)](#) equation (59))

$$\mathcal{S}_m(z) = \mathcal{S}_m^0 \sin\left(\frac{1}{c_m} \int_{-H}^z N(z') dz'\right) \quad (23.184)$$

$$= \mathcal{S}_m^0 \sin(\theta_m), \quad (23.185)$$

subject to the normalization ([Ferrari et al. \(2010\)](#) equation (58c))

$$\int_{-H}^0 \mathcal{S}_m^2 N^2 dz = g \quad (23.186)$$

$$\mathcal{S}_m^0 = \sqrt{\frac{g}{\int_{-H}^0 \sin^2(\theta_m) N^2 dz}}. \quad (23.187)$$

The GM transport can be now written as ([Ferrari et al. \(2010\)](#) equations (11) and (14))

$$\Upsilon^{GM} = \sum_{m=1}^{\infty} \mathcal{S}_m(z) \Upsilon_m \quad (23.188)$$

$$= -\frac{1}{\rho_0} \sum_{m=1}^{\infty} \mathcal{S}_m(z) \left(\int_{-H}^0 \mathcal{S}_m \kappa \nabla_z \rho dz' \right) \quad (23.189)$$

which gives coefficients of

$$\Upsilon_m = -\frac{1}{\rho_0} \left(\int_{-H}^0 \mathcal{S}_m \kappa \nabla_z \rho dz' \right) \quad (23.190)$$

$$= -\frac{1}{g} \int_{-H}^0 \mathcal{S}_m \kappa N^2 dz. \quad (23.191)$$

By considering only the baroclinic modes for $m \leq M$, we can find the low baroclinic mode dominated transport

$$\Upsilon^{BC} = \sum_{m=1}^M \mathcal{S}_m(z) \Upsilon_m. \quad (23.192)$$

Implementation The subroutine `compute_transport_modes` implements this method. The baroclinic modes are calculated for $m = [1, M]$, where M is set by the namelist parameter `number_bc_modes`, using the following discretizations

$$c_{i,j} = \max(c_{\min}, (c_1)_{i,j}) \quad (23.193)$$

$$\theta_{i,j,k}^{(kr)} = \frac{1}{c_{i,j}} \sum_k \sum_{kr} (N \Delta z)_{i,j,k}^{(kr)} \quad (23.194)$$

$$(\sin(\theta_m))_{i,j,k}^{(kr)} = \sin\left(m \theta_{i,j,k}^{(kr)}\right) \quad (23.195)$$

$$(S_m^0)_{i,j} = \sqrt{\frac{g}{\sum_k \sum_{kr} (\sin(\theta_m))_{i,j,k}^{(kr)} (N^2 \Delta z)_{i,j,k}^{(kr)}}} \quad (23.196)$$

$$(S_m)_{i,j,k}^{(kr)} = (S_m^0)_{i,j} (\sin(\theta_m))_{i,j,k}^{(kr)}. \quad (23.197)$$

The value of c_{\min} is set by the namelist parameter `min.bc.speed`. The tracer point averaged values of c , θ and S_1 are diagnosed via the names `bc.speed`, `sine_arg` and `bc.mode_1`. Having calculated the baroclinic modes, the coefficients can then be determined, giving

$$(\Upsilon_m^{(x)})_{i,j}^{(ip)} = -\frac{1}{g} \sum_k \sum_{kr} (S_m)_{i,j,k}^{(kr)} (N^2 S_x \kappa \Delta z)_{i,j,k}^{(ip,kr)} \quad (23.198)$$

$$(\Upsilon_m^{(y)})_{i,j}^{(jq)} = -\frac{1}{g} \sum_k \sum_{kr} (S_m)_{i,j,k}^{(kr)} (N^2 S_y \kappa \Delta z)_{i,j,k}^{(jq,kr)}. \quad (23.199)$$

Finally, the transport is calculated by taking the sum over the baroclinic modes,

$$(\Upsilon_x)_{i,j,k}^{(ip,kr)} = \sum_{m=1}^M (S_m)_{i,j,k}^{(kr)} (\Upsilon_m^{(x)})_{i,j}^{(ip)} \quad (23.200)$$

$$(\Upsilon_y)_{i,j,k}^{(jq,kr)} = \sum_{m=1}^M (S_m)_{i,j,k}^{(kr)} (\Upsilon_m^{(y)})_{i,j}^{(jq)}. \quad (23.201)$$

23.5.3.4 Boundary Value Problem

As a third option, the transport vector can be found as the solution to the one dimensional vertical boundary-value problem (Ferrari et al. (2010) Eqn 16a,b)

$$\left(c_1^2 \frac{d^2}{dz^2} - N^2 \right) \Upsilon(z) = \frac{g}{\rho_0} \kappa \nabla_z \rho \quad (23.202)$$

$$= -\frac{N^2}{\rho_{,z}} \kappa \nabla_z \rho \quad (23.203)$$

$$= N^2 \kappa \mathbf{S} \quad (23.204)$$

$$\Upsilon(\eta) = 0 \quad (23.205)$$

$$\Upsilon(-H) = 0. \quad (23.206)$$

The second order differential operator can be vertically discretized as

$$\frac{d^2 \Upsilon}{dz^2} = \frac{1}{\Delta z} \left(\frac{d^-}{dz} \Upsilon - \frac{d^+}{dz} \Upsilon \right) \quad (23.207)$$

$$= \frac{1}{\Delta z} \left(\frac{\Upsilon^- - \Upsilon^0}{\Delta z^-} - \frac{\Upsilon^0 - \Upsilon^+}{\Delta z^+} \right) \quad (23.208)$$

$$= \frac{1}{\Delta z} \left(\frac{\Upsilon^-}{\Delta z^-} - \Upsilon^0 \left(\frac{1}{\Delta z^-} + \frac{1}{\Delta z^+} \right) + \frac{\Upsilon^+}{\Delta z^+} \right). \quad (23.209)$$

The left hand side of equation (23.202) thus becomes.

$$\left(c_1^2 \frac{d^2}{dz^2} - N^2 \right) \Upsilon = \frac{c_1^2}{\Delta z} \left(\frac{\Upsilon^-}{\Delta z^-} - \Upsilon^0 \left(\frac{1}{\Delta z^-} + \frac{1}{\Delta z^+} \right) + \frac{\Upsilon^+}{\Delta z^+} \right) - N^2 \Upsilon^0 \quad (23.210)$$

$$= \frac{c_1^2 \Upsilon^-}{\Delta z \Delta z^-} - \Upsilon^0 \left(N^2 + \frac{c_1^2}{\Delta z \Delta z^-} + \frac{c_1^2}{\Delta z \Delta z^+} \right) + \frac{c_1^2 \Upsilon^+}{\Delta z \Delta z^+}. \quad (23.211)$$

Finally, we can write down the vertically discretized boundary value problem as

$$\frac{c_1^2 \Upsilon^-}{\Delta z \Delta z^-} - \Upsilon^0 \left(N^2 + \frac{c_1^2}{\Delta z \Delta z^-} + \frac{c_1^2}{\Delta z \Delta z^+} \right) + \frac{c_1^2 \Upsilon^+}{\Delta z \Delta z^+} = N^2 \kappa \mathbf{S} \quad (23.212)$$

$$\frac{c_1^2 \Upsilon^-}{\Delta z^-} - \Upsilon^0 \left(N^2 \Delta z + \frac{c_1^2}{\Delta z^-} + \frac{c_1^2}{\Delta z^+} \right) + \frac{c_1^2 \Upsilon^+}{\Delta z^+} = N^2 \kappa \mathbf{S} \Delta z, \quad (23.213)$$

which can be written as

$$a\Upsilon^- + b\Upsilon^0 + c\Upsilon^+ = \mathbf{r} \quad (23.214)$$

where

$$a = \frac{c_1^2}{\Delta z^-} \quad (23.215)$$

$$b = -\left(N^2 \Delta z + \frac{c_1^2}{\Delta z^-} + \frac{c_1^2}{\Delta z^+}\right) \quad (23.216)$$

$$c = \frac{c_1^2}{\Delta z^+} \quad (23.217)$$

$$\mathbf{r} = N^2 \kappa \mathbf{S} \Delta z.$$

The boundary conditions given by equations (23.205) and (23.206) exist at the vertical faces of tracer cells. As such, the following calculations are all performed on values at the bottom face of the tracer cells. The right hand side term, \mathbf{r} is discretized as

$$(r_x)_{i,j,k}^{b(ip)} = (N^2 S_x \kappa \Delta z)_{i,j,k}^{(ip,1)} + (N^2 S_x \kappa \Delta z)_{i,j,k+1}^{(ip,0)} \quad (23.218)$$

while the left hand side coefficients are given by

$$a_{i,j,k} = \frac{(c_1)_{i,j}}{\Delta z_{i,j,k}} \quad (23.219)$$

$$c_{i,j,k} = \frac{(c_1)_{i,j}}{\Delta z_{i,j,k+1}} \quad (23.220)$$

$$b_{i,j,k} = -\left(\left((N \Delta z)_{i,j,k}^{(1)} + (N \Delta z)_{i,j,k+1}^{(0)}\right) + a_{i,j,k} + c_{i,j,k}\right) \quad (23.221)$$

Now, since $\Upsilon_0 = \Upsilon_{kmt} = 0$, the discretized one dimensional problem for a given column can be written (omitting the i, j subscript) as

$$\begin{pmatrix} b_1 & c_1 & & & & 0 \\ a_2 & b_2 & c_2 & & & \\ & \ddots & \ddots & \ddots & & \vdots \\ & & a_k & b_k & c_k & \\ \vdots & & & \ddots & \ddots & \\ 0 & \dots & & a_{kmt-2} & b_{kmt-2} & c_{kmt-2} \\ & & & a_{kmt-1} & b_{kmt-1} & \end{pmatrix} \begin{pmatrix} \Upsilon_1 \\ \Upsilon_2 \\ \vdots \\ \Upsilon_k \\ \vdots \\ \Upsilon_{kmt-2} \\ \Upsilon_{kmt-1} \end{pmatrix} = \begin{pmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \\ \vdots \\ \mathbf{r}_k \\ \vdots \\ \mathbf{r}_{kmt-2} \\ \mathbf{r}_{kmt-1} \end{pmatrix}. \quad (23.222)$$

This result gives us the following matrix equation for the transport vector,

$$A\Upsilon = \mathbf{r} \quad (23.223)$$

$$\Upsilon = A^{-1} \mathbf{r}. \quad (23.224)$$

When discretized, this result leads to values for Υ evaluated at the base of the tracer cells.

$$(\Upsilon_x)_{i,j,k}^{b(ip)} = \left((A_{i,j})^{-1} \cdot (r_x)_{i,j}^{b(ip)} \right)_k \quad (23.225)$$

$$(\Upsilon_y)_{i,j,k}^{b(jq)} = \left((A_{i,j})^{-1} \cdot (r_y)_{i,j}^{b(jq)} \right)_k \quad (23.226)$$

For each column we need to solve a tridiagonal matrix problem using a single matrix and four different right hand side vectors. This is done using a modified version of the `invtri` algorithm described in [Press et al. \(1992\)](#), which is implemented as the subroutine `invtri_bvp`. The modified algorithm vectorizes the process across all columns and also saves use from inverting the same matrix multiple times for each column.

23.5.4 Vertical Boundary Tensor Values

The tapering processes applied above reduce the off-diagonal and (3, 3) terms of the tensors towards zero at the surface. Another option available to the user is to set the diffusivity tensors to be explicitly horizontal at both the surface and bottom of the ocean. If the namelist parameter `tmask_neutral_on == .true.` then the values of the tensors at the surface are modified to be

$$(K_{33x})_{i,j,1}^{(ip,kr)} \rightarrow 0 \quad (23.227)$$

$$(K_{33y})_{i,j,1}^{(jq,kr)} \rightarrow 0 \quad (23.228)$$

$$(K_{13})_{i,j,1}^{(ip,kr)} \rightarrow 0 \quad (23.229)$$

$$(K_{23})_{i,j,1}^{(jq,kr)} \rightarrow 0 \quad (23.230)$$

$$(\Upsilon_x)_{i,j,1}^{(ip,kr)} \rightarrow 0 \quad (23.231)$$

$$(\Upsilon_y)_{i,j,1}^{(jq,kr)} \rightarrow 0 \quad (23.232)$$

and at the ocean bottom are modified to be

$$(K_{33x})_{i,j,kmt}^{(ip,kr)} \rightarrow 0 \quad (23.233)$$

$$(K_{33y})_{i,j,kmt}^{(jq,kr)} \rightarrow 0 \quad (23.234)$$

$$(K_{13})_{i,j,kmt}^{(ip,kr)} \rightarrow 0 \quad (23.235)$$

$$(K_{23})_{i,j,kmt}^{(jq,kr)} \rightarrow 0 \quad (23.236)$$

$$(\Upsilon_x)_{i,j,kmt}^{(ip,kr)} \rightarrow 0 \quad (23.237)$$

$$(\Upsilon_y)_{i,j,kmt}^{(jq,kr)} \rightarrow 0. \quad (23.238)$$

23.6 Tracer Gradients

The gradients of tracer concentrations are fundamental to neutral physics calculations, as they define the direction and magnitude of the physical processes being parameterized. The tracer flux is linearly related to the tracer gradient, while the gradients of conservative temperature and salinity are also required to calculate the neutral slopes (equation (23.22)).

Finite difference approximations of the tracer gradient components are defined by the following equations (Griffies (2004) equation (16.58)),

$$\nabla T^n = (\nabla_s + \hat{\mathbf{z}} \partial_z) T^n \quad (23.239)$$

$$\rightarrow (\delta_x + \delta_y + \delta_z) T_{i,j,k}^n \quad (23.240)$$

$$(\delta_x T_n)_{i,j,k}^E = \frac{T_{i+1,j,k}^n - T_{i,j,k}^n}{\Delta x_{i,j}^E} \quad (23.241)$$

$$(\delta_y T_n)_{i,j,k}^N = \frac{T_{i,j+1,k}^n - T_{i,j,k}^n}{\Delta y_{i,j}^N} \quad (23.242)$$

$$(\delta_z T_n)_{i,j,k}^b = \frac{T_{i,j,k}^n - T_{i,j,k+1}^n}{\Delta z_{i,j,k}^b}, \quad (23.243)$$

with vertical boundary conditions

$$(\delta_z T_n)_{i,j,0}^b = 0 \quad (23.244)$$

$$(\delta_z T_n)_{i,j,nk}^b = 0. \quad (23.245)$$

The gradients are calculated at all points on the grid for all tracers. The z gradient has an extra level defined at $k = 0$ which serves to simplify notation in subsequent calculations. To account for points where tracers are not defined (such as over land), each derivative is masked using the grid's tracer mask, based on those points used in each calculation.

$$(\delta_x T_n)_{i,j,k}^E \rightarrow M_{i,j,k} M_{i+1,j,k} (\delta_z T_n)_{i,j,k}^E \quad (23.246)$$

$$(\delta_y T_n)_{i,j,k}^N \rightarrow M_{i,j,k} M_{i,j+1,k} (\delta_z T_n)_{i,j,k}^N \quad (23.247)$$

$$(\delta_z T_n)_{i,j,k}^b \rightarrow M_{i,j,k} M_{i,j,k+1} (\delta_z T_n)_{i,j,k}^b. \quad (23.248)$$

This masking ensures that the tracer derivatives are defined only across tracer cell faces when the tracer exists on both sides of the cell face. The tracer gradients are computed in the subroutine `tracer_gradients`. The horizontal derivatives are calculated using the `FDX_T` and `FDY_T` functions. These functions come from the `ocean_operators` module.

Note that we may consider two calculations of the lateral tracer gradients. The first takes derivatives along surfaces of constant vertical coordinate (constant k -level). This method follows the discretization given above. The second method linearly maps tracers to a constant depth level before taking derivatives, providing a horizontal derivative relative to a constant z -level. This approach can introduce spurious extrema and so is not recommended, except perhaps for terrain following σ -models. As neutral physics has not been implemented in MOM for sigma models (see [Lemarié et al. \(2012b,a\)](#) for a discussion), we do not support the second derivative calculation in MOM.

23.7 Quantities related to density gradients

In this section, we detail the calculation of the density gradient, with this calculation used to compute various density related quantities.

23.7.1 Gradient of locally referenced potential density

Components of the locally referenced potential density gradient $\nabla\rho = (\nabla_s + \hat{\mathbf{z}}\partial_z)\rho$ are written as ([Griffies \(2004\)](#) equation 13.36)

$$\frac{\partial\rho}{\partial x} = \frac{\partial\Theta}{\partial x} \left(\frac{\partial\rho}{\partial\Theta} \right)_{s,p} + \frac{\partial S}{\partial x} \left(\frac{\partial\rho}{\partial S} \right)_{\Theta,p} \quad (23.249)$$

$$\frac{\partial\rho}{\partial y} = \frac{\partial\Theta}{\partial y} \left(\frac{\partial\rho}{\partial\Theta} \right)_{s,p} + \frac{\partial S}{\partial y} \left(\frac{\partial\rho}{\partial S} \right)_{\Theta,p} \quad (23.250)$$

$$\frac{\partial\rho}{\partial z} = \frac{\partial\Theta}{\partial z} \left(\frac{\partial\rho}{\partial\Theta} \right)_{s,p} + \frac{\partial S}{\partial z} \left(\frac{\partial\rho}{\partial S} \right)_{\Theta,p}, \quad (23.251)$$

where Θ is potential temperature or conservative temperature and S is salinity. The spatial gradients of temperature and salinity are available in the model as spanning variables, while the partial derivatives of density are available at the tracer points (as components of the `Dens` model variable). The equations above are therefore discretized as

$$\delta_x \rho_{i,j,k}^{(ip)} = (\delta_x \Theta)_{i-1+ip,j,k}^E (\rho, \Theta)_{i,j,k} + (\delta_x S)_{i-1+ip,j,k}^E (\rho, S)_{i,j,k} \quad (23.252)$$

$$\delta_y \rho_{i,j,k}^{(jq)} = (\delta_y \Theta)_{i,j-1+jq,k}^N (\rho, \Theta)_{i,j,k} + (\delta_y S)_{i,j-1+jq,k}^N (\rho, S)_{i,j,k} \quad (23.253)$$

$$\delta_z \rho_{i,j,k}^{(kr)} = (\delta_z \Theta)_{i,j,k-1+kr}^b (\rho, \Theta)_{i,j,k} + (\delta_z S)_{i,j,k-1+kr}^b (\rho, S)_{i,j,k}. \quad (23.254)$$

These formulae are similar to those given in Griffies (2004) equations (16.64) and (16.65), however the indexing scheme differs. The indexing scheme used here directly reflects that used in the code implementation.

Calculation of the density gradients are performed in the subroutine `gradrho`. After calculating the gradient, various adjustments may be made to the vertical derivative $\delta_z \rho_{i,j,k}^{(kr)}$ to produce the *adjusted vertical density derivative*.

23.7.1.1 Optional adjustments of the vertical density derivative

The following calculations are performed in the subroutine `adjust_drhodz`. The first adjustment ensures that the adjusted vertical derivative is always less than zero, i.e. it enforces a stable stratification. As per equation (15.15) of Griffies (2004), the adjusted vertical density derivative is to set

$$\delta_z \rho_{i,j,k}^{(kr)} \rightarrow \min \left(\delta_z \rho_{i,j,k}^{(kr)}, -\epsilon \right). \quad (23.255)$$

This method guarantees that the adjusted derivative is always less than zero and thus prevents numerical instabilities in subsequent calculations.

The second adjustment is a vertical smoothing process. If the namelist parameter `drhodz_smooth_vert == .true.` then a simple 1-2-1 filter is applied to each point between $k = 2$ and $k = \text{kbot} - 2$, i.e. all those points on which the derivative is defined for the full three point stencil. The formula for this filter can be written as

$$\delta_z \rho_{i,j,k}^{(kr)} \rightarrow \frac{\delta_z \rho_{i,j,k-1}^{(kr)} + \delta_z \rho_{i,j,k+1}^{(kr)}}{4} + \frac{\delta_z \rho_{i,j,k}^{(kr)}}{2}. \quad (23.256)$$

The vertical filtering is performed by the subroutine `vert_smooth` in the module `ocean_nphysics_util_new`. This module has a namelist parameter `num_121_passes`, which controls how many times the vertical smoothing process is performed. By default this value is set to 1, however larger values can be set to obtain an even smoother vertical gradient.

The final adjustment made is a horizontal smoothing process. If the namelist parameter `drhodz_smooth_horz == .true.` then the transformation

$$\delta_z \rho_{i,j,k}^{(kr)} \rightarrow \delta_z \rho_{i,j,k}^{(kr)} + \Delta \tau \cdot \text{LAP_T} \left(\delta_z \rho_{i,j,k}^{(kr)}, (\kappa_g)_{i,j} \right) \quad (23.257)$$

is applied. The operator `LAP_T` is defined in the module `ocean_operators` and performs horizontal diffusive smoothing. The diffusivity parameter is defined as

$$(\kappa_g)_{i,j} = v_g (L_g)_{i,j} \quad (23.258)$$

where v_g is specified by the namelist parameter `vel_micom_smooth` and L_g is the grid length scale, defined as

$$(L_g)_{i,j} = H \left(\Delta x_{i,j}^T, \Delta y_{i,j}^T \right). \quad (23.259)$$

The name “Micom” is motivated by a suggestion by Eric Chassignet, in which he noted that this is how the Miami Isopycnal Ocean Model based its diffusivities in the early 2000s.

23.7.1.2 Diagnostics

The density gradients are available as diagnostics via the names `drhodb`, `drhody` and `drhodz`. The diagnosed values are calculated at the tracer points as the average over the spanning variables, giving

$$\delta_x \rho_{i,j,k} = \sum_{ip} \delta_x \rho_{i,j,k}^{(ip)} \quad (23.260)$$

$$\delta_y \rho_{i,j,k} = \sum_{jq} \delta_y \rho_{i,j,k}^{(jq)} \quad (23.261)$$

$$\delta_z \rho_{i,j,k} = \sum_{kr} \delta_z \rho_{i,j,k}^{(kr)}. \quad (23.262)$$

23.7.2 Neutral Slopes

We now detail how neutral slopes are computed in MOM.

23.7.2.1 Neutral Slope Vector

The neutral slope vector

$$\mathbf{S} = - \left(\frac{\rho_{,\Theta} \nabla_s \Theta + \rho_{,S} \nabla_s S}{\rho_{,\Theta} \partial_z \Theta + \rho_{,S} \partial_z S} \right) \quad (23.263)$$

(Griffies (2004) equation 13.42) is discretized by combining two spanning variables, which leads to the slope vector components being defined over a centred triad group (see Section 23.2.5). By taking all available combinations of density gradients at the tracer point we obtain the following discretization

$$(S_x)_{i,j,k}^{(ip,kr)} = - \left(\frac{\delta_x \rho_{i,j,k}^{(ip)}}{\delta_z \rho_{i,j,k}^{(kr)}} \right). \quad (23.264)$$

To avoid dividing by zero, the slopes are only calculated over those grid points where the vertical density derivative is well defined, based on the grid's tracer mask. Since the vertical density gradient is not defined through the surface or bottom boundaries, we set the slopes at these boundaries to their adjacent values,

$$(S_x)_{i,j,1}^{(ip,0)} = (S_x)_{i,j,1}^{(ip,1)} \quad (23.265)$$

$$(S_x)_{i,j,kmt}^{(ip,1)} = (S_x)_{i,j,kmt}^{(ip,0)}. \quad (23.266)$$

The y slope is computed in an identical manner, replacing x with y and ip with jq .

23.7.2.2 Neutral Slope Magnitude

If we average over horizontal half cells of slope, we obtain the vertical half cell slope vector

$$(S_x)_{i,j,k}^{(kr)} = \frac{1}{2} \sum_{ip} (S_x)_{i,j,k}^{(ip,kr)} \quad (23.267)$$

$$(S_y)_{i,j,k}^{(kr)} = \frac{1}{2} \sum_{jq} (S_y)_{i,j,k}^{(jq,kr)}, \quad (23.268)$$

from which we can calculate the magnitude of the slope vector over the vertical half cells,

$$|\mathbf{S}|_{i,j,k}^{(kr)} = \sqrt{\left((S_x)_{i,j,k}^{(kr)} \right)^2 + \left((S_y)_{i,j,k}^{(kr)} \right)^2}. \quad (23.269)$$

As well as calculating the magnitude of the slope vector over the vertical half cells, we also calculate the average slope vector at each point, defined as

$$(S_x)_{i,j,k} = \frac{1}{4} \sum_{ip,kr} (S_x)_{i,j,k}^{(ip,kr)} \quad (23.270)$$

$$(S_y)_{i,j,k} = \frac{1}{4} \sum_{jq,kr} (S_y)_{i,j,k}^{(jq,kr)}. \quad (23.271)$$

This allows us to calculate the magnitude of the slope vector at the tracer point as

$$|\mathbf{S}|_{i,j,k} = \sqrt{(S_x)_{i,j,k}^2 + (S_y)_{i,j,k}^2}. \quad (23.272)$$

The subroutine `neutral_slopes` returns the slope vector over the centered triad groups, $\left((S_x)_{i,j,k}^{(ip,kr)}, (S_y)_{i,j,k}^{(jq,kr)} \right)$, and the slope vector magnitudes $|\mathbf{S}|_{i,j,k}^{(kr)}$ and $|\mathbf{S}|_{i,j,k}$.

23.7.2.3 Diagnostics

The components of the slope vector and the slope vector magnitude are diagnosed at the tracer point via the diagnostic table entries `slopex`, `slopey` and `absslope`. These diagnostics represent the values calculated in equations (23.270), (23.271) and (23.272), respectively.

23.7.3 Buoyancy

The squared buoyancy frequency is defined as

$$N^2 = -\frac{g}{\rho_0} \rho_{,z}. \quad (23.273)$$

This equation is discretized directly in accordance with the vertical derivative of the locally referenced potential density

$$(N^2)_{i,j,k}^{(kr)} = -\frac{g}{\rho_0} (\rho_{,z})_{i,j,k}^{(kr)}. \quad (23.274)$$

The average over the vertical spans of this quantity is diagnosed via the diagnostic table entry `N2`.

A related quantity of interest is the product $N\Delta z$ over the vertical half cells. This value is calculated as

$$(N\Delta z)_{i,j,k}^{(kr)} = \sqrt{(N^2)_{i,j,k}^{(kr)} \Delta z_{i,j,k}^{(kr)}}. \quad (23.275)$$

Since calculation of the adjusted vertical density derivative is always negative, the term under the square root is always positive.

23.7.4 Baroclinic Gravity Wave Speed

We also make use of the first baroclinic mode gravity wave speed (Griffies (2004) equation (14.83)), which is approximated as

$$c_1 = \frac{1}{\pi} \int_{-H}^0 N \, dz. \quad (23.276)$$

The discretization of this speed becomes

$$(c_1)_{i,j} = \frac{1}{\pi} \sum_{k=1}^{nk} \sum_{kr} (N\Delta z)_{i,j,k}^{(kr)}, \quad (23.277)$$

which is diagnosed via the diagnostic table entry `gravity_wave_speed`.

23.7.5 Rossby Radius

When calculating the Rossby radius, different equations apply depending on whether or not a point is near the equator. The respective equations, given in Griffies (2004) equations (14.81) and (14.82) are

$$\lambda_{\text{non-eq}} = \frac{c_1}{|f|} \quad (23.278)$$

$$\lambda_{\text{eq}} = \sqrt{\frac{c_1}{2\beta}}. \quad (23.279)$$

The Rossby radius is then defined as the smallest of these two values at any given point, leading to

$$\lambda_1 = \min(\lambda_{\text{eq}}, \lambda_{\text{non-eq}}). \quad (23.280)$$

The respective discretizations of these equations are

$$(\lambda_{\text{non-eq}})_{i,j} = \frac{(c_1)_{i,j}}{|f|_{i,j}} \quad (23.281)$$

$$(\lambda_{\text{eq}})_{i,j} = \sqrt{\frac{(c_1)_{i,j}}{2\beta_{i,j}}} \quad (23.282)$$

$$(\lambda_1)_{i,j} = \min((\lambda_{\text{eq}})_{i,j}, (\lambda_{\text{non-eq}})_{i,j}) \quad (23.283)$$

Calculation of these values is performed in the subroutine `compute_rossby_radius` and the Rossby radius is diagnosed via the diagnostic table entry `rossby_radius`.

23.7.6 Neutral Boundary Layers

In regions near the surface of the ocean, the vertical stratification of density becomes weak, which leads to a region of steep neutral slope. This neutral boundary layer can be defined as the region above which the magnitude of the neutral slope vector is above some threshold S_{max} . More precisely, we can define the index of the surface boundary layer as

$$(k_{\text{surf}})_{i,j} = \max(\{0\} \cup \{k : |\mathbf{S}|_{i,j,k'} > S_{\text{max}} \text{ if } k' \leq k\}). \quad (23.284)$$

This vertical index is calculated in the subroutine `neutral_blayer`, using a vertical search algorithm from the surface. The value of $(k_{\text{surf}})_{i,j}$ is used in subsequent calculations related to the [Ferrari et al. \(2008\)](#) and [Danabasoglu et al. \(2008\)](#) methods to obtain various values from the base of the surface boundary layer. The boundary layer index is diagnosed via the diagnostic table entry `ksurf_blayer`.

23.7.7 Summary

The density based calculations discussed above are all handled within the subroutine `density_calculations`. This routine returns values for

- SLOPE VECTOR: $((S_x)_{i,j,k}^{(ip.kr)}, (S_y)_{i,j,k}^{(jq.kr)})$
- SLOPE VECTOR MAGNITUDE: $|\mathbf{S}|_{i,j,k}^{(kr)}$ and $|\mathbf{S}|_{i,j,k}$
- BUOYANCY FREQUENCY: $(N^2)_{i,j,k}^{(kr)}$ and $(N\Delta z)_{i,j,k}^{(kr)}$
- GRAVITY WAVE SPEED: $(c_1)_{i,j}$; the Rossby radius, $(\lambda_1)_{i,j}$
- SURFACE BOUNDARY LAYER INDEX: $(k_{\text{surf}})_{i,j}$.

These density gradient based values will be used in subsequent calculations. It should be noted that at this stage, the actual density gradient is no longer required, as these derived quantities are sufficient for the remaining calculations.

23.8 Specification of the diffusivity

Equations (23.20) and (23.28) each feature a diffusivity, which must be computed according to some prescription. The module `ocean_nphysics_diff` is responsible for this calculation, via its public interface subroutine `compute_diffusivity`. In this module, values are computed for the neutral diffusivity A_R and the skew or GM diffusivity κ . The final skew or GM diffusivity is diagnosed via the diagnostic table entry `agm_array` while the neutral diffusivity is diagnosed via the table entry `aredi_array`.

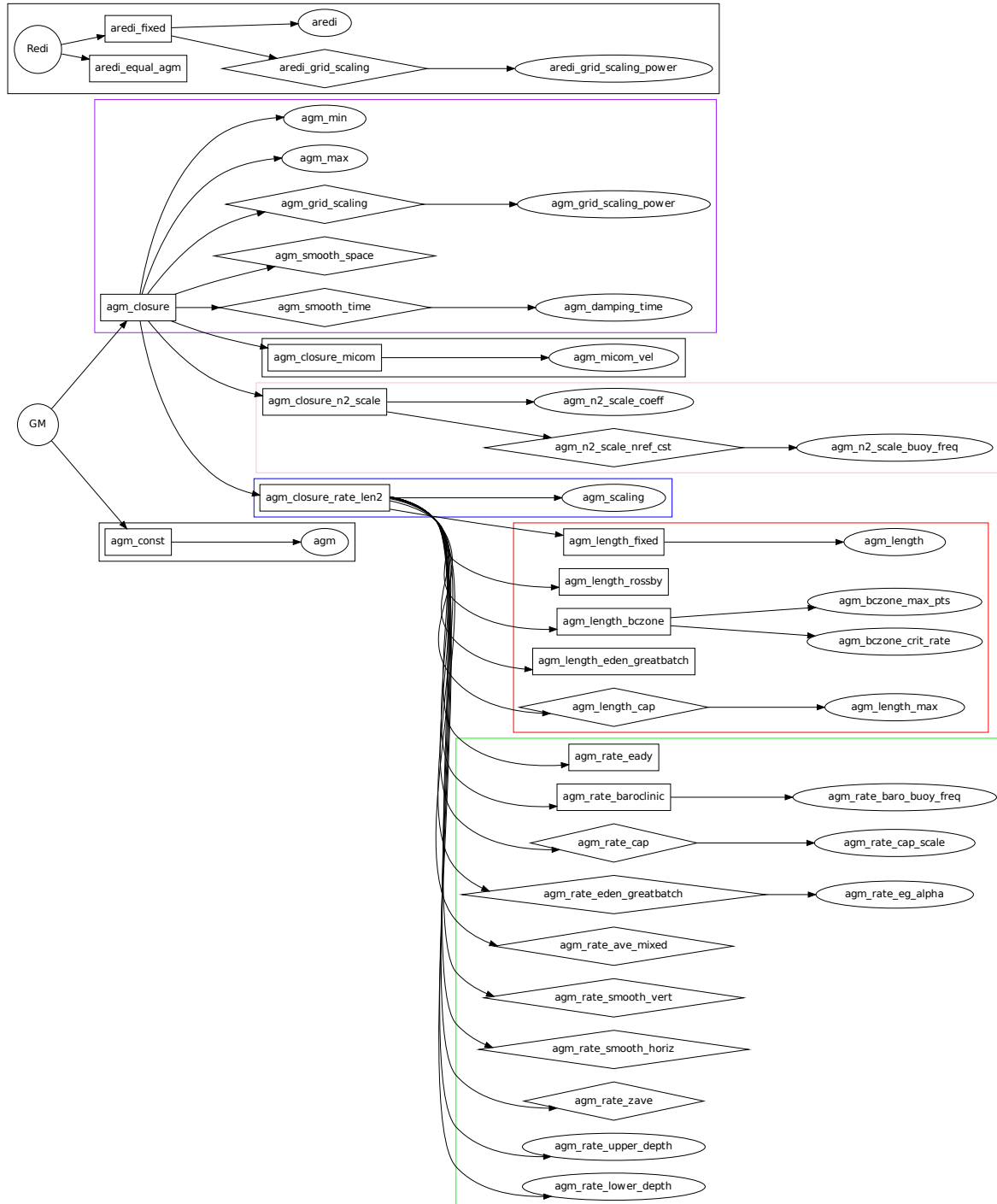


Figure 23.5: Diffusivity module namelist parameters and their relationships.

23.8.1 Namelist Parameters

Specification of diffusivities for the neutral physics parameterization can be performed in many different ways. MOM presently supports over 24,000 different combinations of boolean options. However, not all

combinations are physically sensible. It is thus important to understand how the parameters fit together to ensure that a physically meaningful set of options is selected. Figure 23.5 gives a visual representation of all namelist parameters in the `ocean_nphysics_diff` module. Parameters are split into logical groups according to their function and dependencies. The top level nodes, indicated by circles, are not actual parameters themselves, but serve to break the parameters into their main groups. Rectangular and diamond nodes represent boolean options, while elliptical nodes represent numerical values.

More specifically, the diamond nodes indicate boolean switches that may be selected independently from other options in their grouping. Rectangular nodes at the same level as each other indicate where a selection of one particular option must be made. For example, in the “Redi” group, either `aredi_fixed` or `aredi_equal_agm` must be true. If a boolean node is set to false, then child nodes in the tree are not considered. If a boolean node is set true, then its child nodes will be considered and must either be set to an appropriate value, or rely on the default value. Users are advised to consult this graph to ensure that a valid combination of namelist parameters is used. The subroutine `check_nml_options` verifies that a valid combination of parameters has been chosen when the module is initialised. If an invalid combination is picked then a fatal error is raised. However, the error checking evaluates only basic settings, so that if a “valid” option is chosen, there is no guarantee that it will be physically sensible.

23.8.2 Fixed GM Diffusivity

The simplest method for setting the GM diffusivity is to use a fixed or constant diffusivity. If the namelist parameter `agm_const == .true.`, then a space-time constant diffusivity value is used. This choice leads to the result

$$\kappa_{i,j,k} = \kappa_0, \quad (23.285)$$

where κ_0 is set by the namelist parameter `agm`.

23.8.3 Flow Dependent GM Diffusivity

Research during the past decade has indicated that a more proper choice for the GM diffusivity is one that is determined by the properties of the flow. These space-time dependent schemes come in many flavors, with MOM offering many options.

23.8.3.1 Selecting a GM Diffusivity Closure

To enable one of the closure methods for a flow dependent GM diffusivity, one needs to set

```
agm_closure == .true.
```

Figure 23.5 reveals three main options to choose:

- `agm_closure_micom` `FIXME` This option is not flow dependent; it is only slightly more complex than constant `agm`. It should thus be allowed even if `agm_closure == .false.`
- `agm_closure_n2_ref`
- `agm_closure_rate_len2`.

Furthermore, if `agm_closure_rate_len2` is chosen, then one of four length scales and one of two growth rate options must be chosen. Based on these options, a diffusivity will be calculated, giving a flow dependent value for $\kappa_{i,j,k}$ at each grid point.

23.8.3.2 Diffusivity Postprocessing

Each of the closure methods produces an array of diffusivities calculated at each grid point. A common set of postprocessing techniques are subsequently applied. If `agm_grid_scaling == .true.` then the grid scaling technique, described in Section 23.8.3.3, is applied to the diffusivity array. An upper and lower

bound, as set by the namelist parameters `agm_max` and `agm_min`, is then applied to all elements of the array, giving

$$\kappa_{i,j,k} \rightarrow \min(\kappa_{\max}, \max(\kappa_{\min}, \kappa_{i,j,k})). \quad (23.286)$$

The value of κ as calculated at this stage is diagnosed via the diagnostic table entry `agm_fast`, as it represents the diffusivity before high-frequency components are filtered out below.

If `agm_smooth_time == .true.` then a low pass time filter is applied to the diffusivity. This filter is calculated as

$$\kappa_{i,j,k} \rightarrow \kappa_{i,j,k}(\tau - 1) - \gamma(\kappa_{i,j,k}(\tau - 1) - \kappa_{i,j,k}) \quad (23.287)$$

where $\gamma = \frac{\Delta\tau}{\tau_k}$ and τ_k is set to the namelist parameter `agm_damping_time`. Next, if `agm_smooth_space == .true.` then the diffusivity array is horizontally smoothed with the `horz_smooth` function. Neither the space nor time filter are typically used at GFDL, preferring instead to use the unfiltered value. Finally, the diffusivity array is masked using the grid's tracer mask to give

$$\kappa_{i,j,k} \rightarrow M_{i,j,k} \kappa_{i,j,k}. \quad (23.288)$$

23.8.3.3 Grid Scaling

The grid scaling of GM diffusivity is applied as a function of the grid length and the Rossby radius, so that

$$\kappa_{i,j,k} \rightarrow \frac{1}{1 + \left(\frac{(\lambda_1)_{i,j}}{(L_g)_{i,j}} \right)^\gamma} \kappa_{i,j,k}, \quad (23.289)$$

where the value of γ is set by the namelist parameter `agm_grid_scaling_power`. This function serves to decrease the diffusivity as the Rossby radius increases and the grid scale decreases. That is, the diffusivity is decreased in regions where the grid scale is smaller than the Rossby radius. The intent is to reduce the effects of the parameterization in regions where the flow field may otherwise be explicitly representing the mesoscale eddies.

23.8.4 MICOM

If `agm_closure_micom == .true.` then the GM diffusivity is calculated as the product of a characteristic velocity and length scale

$$\kappa = v_0 L_g. \quad (23.290)$$

The length scale, L_g , is taken as the grid length, described in Section 23.7.1. The velocity v_0 is given by the namelist parameter `agm_micom_vel`. This choice leads to the discretization

$$\kappa_{i,j,k} = v_0 L_{g_{i,j}}. \quad (23.291)$$

23.8.5 Buoyancy Scaling

Another method of computing the GM diffusivity is to take a fixed diffusivity and then scale it according to the squared buoyancy frequency at each grid point. This method is selected by setting

$$\text{agm_closure_n2_scale} == \text{.true..}$$

The formula for the diffusivity is then

$$\kappa = \kappa_0 \left(\frac{N^2}{N_{\text{ref}}^2} \right). \quad (23.292)$$

The reference squared buoyancy frequency can either be taken to be a constant value or the computed value at the base of the surface boundary layer. If `agm_n2_scale_nref_cst == .true.` then the value of

N_{ref}^2 is set by the square of the value of the namelist parameter `agm_n2_scale_buoy_freq`. Otherwise the discretization

$$(N_{\text{ref}}^2)_{i,j} = \frac{1}{2} \sum_{kr} (N^2)^{(kr)}_{i,j,k_{\text{surf},i,j}} \quad (23.293)$$

is used. This choice uses the boundary layer index calculated in Section 23.7.6. The reference squared buoyancy frequency is diagnosed via the diagnostic table entry name `N2_ref`. The final discretization is then given by

$$\kappa_{i,j,k} = \kappa_0 \frac{\frac{1}{2} \sum_{kr} (N^2)^{(kr)}_{i,j,k}}{(N_{\text{ref}}^2)_{i,j}}, \quad (23.294)$$

where κ_0 is set by the namelist parameter `agm_n2_scale_coeff`. This diffusivity is depth dependent.

23.8.6 Rate Length Squared

By appealing to a dimensional argument it is clear that a diffusivity can be calculated from the product of a squared length and divided by a time scale (Held and Larichev (1996), Visbeck et al. (1997))

$$\kappa = \sigma L^2. \quad (23.295)$$

MOM allows for a number of different methods to calculate these characteristic scales. By setting

$$\text{agm_closure_rate_len2} == \text{.true.},$$

in conjunction with appropriate rate and length parameters, this method is used to calculate the GM diffusivity. The discretized diffusivity is given by

$$\kappa_{i,j,k} = \alpha \sigma_{i,j,k} L_{i,j,k}^2, \quad (23.296)$$

where α is given by the namelist parameter `agm_scaling`

23.8.6.1 Growth Rate

The growth rate term used by MOM is based on that used in Visbeck et al. (1997), which gives (Griffies (2004) equation (14.80))

$$\sigma = \frac{f}{D} \int_{-D_b}^{-D_t} \text{Ri}^{-1/2} dz, \quad (23.297)$$

where the Richardson number is (Griffies (2004) equation (14.88))

$$\text{Ri} = \left(\frac{f}{N |\mathbf{S}|} \right)^2. \quad (23.298)$$

Equation (23.297) thus becomes

$$\sigma = \frac{1}{D} \int_{-D_b}^{-D_t} N |\mathbf{S}| dz. \quad (23.299)$$

Raw Growth Rate We denote $N |\mathbf{S}|$ the *raw growth rate*, and can calculate it in two different ways. The first simply uses the buoyancy frequency and neutral slope to give

$$N |\mathbf{S}| = \sqrt{N^2} |\mathbf{S}|. \quad (23.300)$$

This growth rate is equivalent to calculating the Eady growth rate Visbeck et al. (1997); Griffies (2004).

The second method is to set the raw growth rate to be proportional to the baroclinicity, giving

$$N|\mathbf{S}| \propto |\nabla_z \rho| \quad (23.301)$$

$$\propto |\rho_z| \sqrt{\frac{\rho_x^2 + \rho_y^2}{\rho_z^2}} \quad (23.302)$$

$$\propto N^2 |\mathbf{S}| \quad (23.303)$$

$$= \frac{N^2 |\mathbf{S}|}{N_0}, \quad (23.304)$$

where N_0 is a constant frequency of proportionality. This corresponds to the method discussed in [Griffies \(2004\)](#) Section 14.4.10, with a default value of $N_0 = 0.004\text{s}^{-1}$. This method is also used in [Farneti et al. \(2010\)](#).

If the namelist parameter `agm_rate_eady == .true.`, then the first option from above is used and the final discretization of the raw growth rate is

$$(N|\mathbf{S}|)_{i,j,k} = \frac{1}{2} \sum_{kr} \sqrt{(N^2)_{i,j,k}^{(kr)}} |\mathbf{S}|_{i,j,k}^{(kr)}, \quad (23.305)$$

where the two vertical stencils have been averaged over. Alternately, if `agm_rate_baroclinic == .true.`, we have

$$(N|\mathbf{S}|)_{i,j,k} = \frac{1}{2N_0} \sum_{kr} (N^2)_{i,j,k}^{(kr)} |\mathbf{S}|_{i,j,k}^{(kr)}, \quad (23.306)$$

where N_0 is set by the namelist parameter `agm_rate_baro_buoy_freq`.

Adjusted Growth Rate Having calculated the raw growth rate, $N|\mathbf{S}|$, a number of subsequent operations can be performed on this value to obtain the inverse time scale σ .

If the namelist parameter `agm_rate_cap == .true.` then the growth rate has a harmonic scaling applied to give

$$(N|\mathbf{S}|)_{i,j,k} \rightarrow H\left((N|\mathbf{S}|)_{i,j,k}, (\sigma_{\max})_{i,j}\right). \quad (23.307)$$

The value of σ_{\max} is controlled by a combination of the Coriolis parameter ([Griffies \(2004\)](#) equation (4.14)) and the dimensionless namelist parameter $\alpha = \text{agm_rate_cap_scale}$, and is computed as

$$|f|_{i,j} = 2\Omega_e |\sin(\phi_{i,j})| \quad (23.308)$$

$$(\sigma_{\max})_{i,j} = \alpha |f|_{i,j}. \quad (23.309)$$

where Ω_e is the angular rotation rate of the earth, given by `omega_earth` from the `ocean_parameters` module, and $\phi_{i,j}$ is the tracer grid point latitude.

If the namelist parameter `agm_rate_ave_mixed == .true.` then the growth rate within the surface mixed layer is set to the vertical average over this region. The depth of the mixed layer D_M is calculated by the subroutine `calc_mixed_layer_depth` from the `ocean_tracer_diag` module and can be diagnosed via the diagnostic table entry `mL_depth`. The vertical averaged growth rate over this region is then given by

$$\overline{N|\mathbf{S}|}_M = \frac{1}{D_M} \int_{-D_M}^0 N|\mathbf{S}| dz \quad (23.310)$$

$$= \frac{\int_{-D_M}^0 N|\mathbf{S}| dz}{\int_{-D_M}^0 dz}. \quad (23.311)$$

The discretization of the mixed layer averaged growth rate is given by

$$\overline{N|S|}_{M,i,j} = \frac{\sum_{k=2}^{k_M} (N|S|)_{i,j,k} \Delta z_{i,j,k}}{\sum_{k=2}^{k_M} \Delta z_{i,j,k}} \quad (23.312)$$

where the lower boundary index is defined as

$$k_{M,i,j} = \max\{k : D_{i,j,k} \leq (D_M)_{i,j}\}. \quad (23.313)$$

Note that the lower index in equation (23.312) is $k = 2$, rather than $k = 1$. This setting is because the growth rate at the surface level $k = 1$ cannot be trusted. The mixed layer averaged growth rate is diagnosed via the diagnostic table entry `ave_mL_rate`. Having calculated the average, those points within the mixed layer are set to this value

$$N|S|_{i,j,k} \rightarrow (\overline{N|S|}_M)_{i,j} \text{ if } D_{i,j,k} < (D_M)_{i,j}. \quad (23.314)$$

If the namelist parameter `agm_rate_smooth_vert == .true.` then vertical smoothing is applied to $N|S|$, using the `vert_smooth` function described in Section 23.7.1.

If `agm_rate_smooth_horiz == .true.` then horizontal smoothing of $N|S|$ is performed. This is done with the `S2D` subroutine from the `ocean_operators` module.

If the namelist parameter `agm_rate_edden_greatbatch == .true.` then another growth rate cap applied. First, the Eden-Greatbatch rate is calculated as

$$\sigma_{EG} = \alpha \frac{c_1}{\lambda_1}, \quad (23.315)$$

where α is a dimensionless scaling constant set by the namelist parameter `agm_rate_eg_alpha`. The Eden-Greatbatch rate is diagnosed via the name `eg_rate`. The cap is then applied using the harmonic mean function, leading to

$$N|S|_{i,j,k} \rightarrow H\left((N|S|)_{i,j,k}, \alpha \frac{(c_1)_{i,j}}{(\lambda_1)_{i,j}}\right). \quad (23.316)$$

Once all the caps and filters have been applied, the vertical average of $N|S|$ is calculated, giving

$$\overline{N|S|} = \frac{1}{D} \int_{-D_b}^{-D_t} N|S| dz \quad (23.317)$$

$$= \frac{\int_{-D_b}^{-D_t} N|S| dz}{\int_{-D_b}^{-D_t} dz}. \quad (23.318)$$

The values of D_t and D_b are set by the namelist parameters `agm_rate_upper_depth` and `agm_rate_lower_depth` respectively. The default values for these bounds are set to $D_t = 100\text{m}$ and $D_b = 2000\text{m}$ respectively. A discussion on these choices can be found in Griffies (2004) Section 14.4.9.

The discretization of equation (23.318) is given by

$$\overline{N|S|}_{i,j} = \frac{\sum_{k=k_b}^{k_t} (N|S|)_{i,j,k} \Delta z_{i,j,k}}{\sum_{k=k_b}^{k_t} \Delta z_{i,j,k}}. \quad (23.319)$$

where the upper and lower boundary indices are defined as

$$(k_b)_{i,j} = \max\{k : D_{i,j,k} \leq D_b\} \quad (23.320)$$

$$(k_t)_{i,j} = \min\{k : D_{i,j,k} \geq D_t\}. \quad (23.321)$$

The calculation of $\overline{N|S|}$ is performed in the function `vertical_average`.

The final growth rate can now be determined. There are two available options, either the vertically averaged growth rate can be used, or the unaveraged values of $N|\mathbf{S}|$ can be used. These options are controlled by the namelist parameter `agm_rate_zave`. If true, then the growth rate is set to

$$\sigma_{i,j,k} = \overline{N|\mathbf{S}|}_{i,j}, \quad (23.322)$$

otherwise we have

$$\sigma_{i,j,k} = (N|\mathbf{S}|)_{i,j,k}. \quad (23.323)$$

If the vertically averaged rate is not used then the diffusivity will be depth dependent.

The raw growth rate, $N|\mathbf{S}|$; growth rate, σ ; and vertically averaged growth rate, $\overline{N|\mathbf{S}|}$, are diagnosed via the diagnostic table entries `raw_growth_rate`, `growth_rate` and `growth_rate_zave` respectively.

23.8.6.2 Length

The length scale can be computed in a number of different ways. The length scale calculations are performed in the subroutine `compute_length`.

If a non-constant length is used (i.e. `agm_length_fixed == .false.`) then the computed length is scaled against the grid length, L_g , using the harmonic mean function (Griffies (2004) equation (14.104)). This choice gives

$$L_{i,j,k} \rightarrow H(L_{i,j,k}, L_{g_{i,j}}). \quad (23.324)$$

Once the length is calculated, an optional upper limit can be applied. If `agm_length_cap == .true.` then

$$L_{i,j,k} \rightarrow \min(L_{i,j,k}, L_{\max}) \quad (23.325)$$

where L_{\max} is set by the namelist parameter `agm_length_max`. The final length scale is diagnosed by the name `agm_length`.

The following options are available for computing the length scale.

Constant: If `agm_length_fixed == .true.` then the length scale is set to a constant value,

$$L_{i,j,k} = L_{\text{const}}, \quad (23.326)$$

where L_{const} is set by the namelist parameter `agm_length`.

Rossby Radius: If `agm_length_rossby == .true.` then the length scale is set by the Rossby radius, as defined in Section 23.7.5, which gives

$$L_{i,j,k} = (\lambda_1)_{i,j}. \quad (23.327)$$

Motivation for this choice is given in Griffies (2004) Section 14.4.4.

Baroclinic Zone Radius: If `agm_length_bczone == .true.` then the length scale is set by the width of the baroclinic zone, as done in Visbeck et al. (1997). The algorithm used in MOM was first used in the Hadley Centre model Gordon et al. (2000) and was first implemented in MOM in version 3 Pacanowski and Griffies (1999).

The algorithm used to calculate the length scale R_{bc} is based on the vertically averaged growth rate and is described in Section 23.8.6.3. The diffusivity calculation length scale is then given by

$$L_{i,j,k} = (R_{bc})_{i,j}. \quad (23.328)$$

Eden-Greatbatch: If `agm_length_eden_greatbatch == .true.` then the length scale is set using the method described in [Eden et al. \(2009\)](#). This takes the length scale as the minimum of the Rhines length and the Rossby radius, where the Rossby radius is as calculated above and the Rhines length is a function of the growth rate and is given by

$$L_{Rhi} = \frac{\sigma}{\beta}, \quad (23.329)$$

where β is the beta plane parameter ([Griffies \(2004\)](#) Section 18.1.3). The value of L_{Rhi} is diagnosed via the name `rhines_length`. The discretized Eden-Greatbatch length is thus given by

$$\beta_{i,j} = 2.28^{-11} |\cos(\phi_{i,j})| \quad (23.330)$$

$$L_{i,j,k} = \min\left((\lambda_1)_{i,j}, \frac{\sigma_{i,j,k}}{\beta_{i,j}}\right). \quad (23.331)$$

Since this length scale is based on the growth rate, it introduces a depth dependence to the diffusivity.

23.8.6.3 Baroclinic Zone Width

A length scale can be defined as the width of the baroclinic zone. This region is defined as those grid points with an average growth rate $\overline{N|S|}_{i,j}$ above a certain threshold, as set by the namelist parameter `agm_bczone_crit_rate`. Those points outside of the baroclinic zone are set to have a length scale equal to the grid length, L_g .

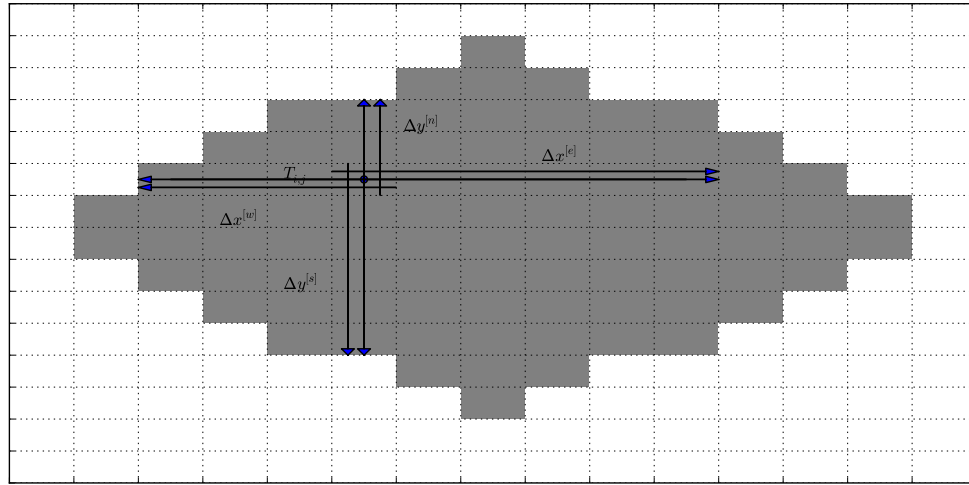


Figure 23.6: The grey region indicates where the baroclinic rate is above the threshold. The arrows indicate the extant of the baroclinic region for the point $T_{i,j}$ when the maximum number of points is set to 5.

For points inside the baroclinic zone, the length scale is calculated as a function of the distance to the edge of the zone in each of the four directions, as shown in Figure 23.6. To find the width in each direction from a given point, a search is performed in each direction. The search begins at the point and then considers each point in the direction until either a point outside the baroclinic zone is reached, or a maximum number of cells are considered. The maximum number of points to consider is set by the namelist parameter `agm_bczone_max_pts`. The directional width is then taken as the sum of the tracer cells widths of all those points within the baroclinic zone in the given direction.

In Figure 23.6, the search is performed in each direction with the maximum number of points set to 5. In the north, south and west directions, the search continues to the edge of the baroclinic zone, whilst to

the east it reaches the maximum number of points. The directional widths, $\Delta x^{[E]}$, $\Delta x^{[W]}$, $\Delta y^{[N]}$ and $\Delta y^{[S]}$ are indicated by the single headed arrows.

Having calculated the directional widths, the total widths can be calculated as

$$\Delta x_{i,j}^{[EW]} = \Delta x^{[E]} + \Delta x^{[W]} - \Delta x_{i,j} \quad (23.332)$$

$$\Delta y_{i,j}^{[NS]} = \Delta y^{[N]} + \Delta y^{[S]} - \Delta y_{i,j}, \quad (23.333)$$

where we have subtracted the tracer width of the cell to avoid double counting. These widths are indicated by double headed arrows in Figure 23.6.

From this point, the baroclinic length scale is defined as the maximum of these total widths, scaled according to how centred it is within the zone. This specification leads to the final equations

$$(L_{bc})_{i,j} = \frac{\min(\Delta x_{i,j}^{[E]}, \Delta x_{i,j}^{[W]})}{\max(\Delta x_{i,j}^{[E]}, \Delta x_{i,j}^{[W]})} \Delta x_{i,j}^{[EW]} \text{ if } \Delta x_{i,j}^{[EW]} \geq \Delta y_{i,j}^{[NS]} \quad (23.334)$$

$$(L_{bc})_{i,j} = \frac{\min(\Delta y_{i,j}^{[N]}, \Delta y_{i,j}^{[S]})}{\max(\Delta y_{i,j}^{[N]}, \Delta y_{i,j}^{[S]})} \Delta y_{i,j}^{[NS]} \text{ otherwise.} \quad (23.335)$$

The value of L_{bc} is diagnosed via the name `bczone_radius`.

23.8.7 Redi Diffusivity

The calculation of the Redi diffusivity is much simpler than that of the GM diffusivity. Either a constant value is used, or the GM diffusivity is used. In most cases, one should choose to set the Redi diffusivity equal to the GM diffusivity. However, tuning purposes may motivate taking different values.

If the namelist parameter `aredi_equal_agm == .true.` then the Redi diffusivity is set to be identical to the GM diffusivity, giving

$$(A_R)_{i,j,k} = \kappa_{i,j,k}. \quad (23.336)$$

If `aredi_fixed == .true.` then a constant diffusivity is used, giving

$$(A_R)_{i,j,k} = A_0, \quad (23.337)$$

where A_0 is set by the namelist parameter `aredi`. In this case, grid scaling can also be applied to the diffusivity. If `aredi_grid_scaling == .true.` then grid scaling is applied as described in Section 23.8.3.3, giving

$$(A_R)_{i,j,k} \rightarrow \frac{1}{1 + \left(\frac{(\lambda_1)_{i,j}}{(L_g)_{i,j}} \right)^\gamma} (A_R)_{i,j,k}, \quad (23.338)$$

where in this case, γ is set by the namelist parameter `aredi_grid_scaling_power`.

23.9 Summary of the notation

This section summarizes the notation used in this chapter.

23.9.1 General

Value	Variable	Units
Δx	Grid%dx	m
Δy	Grid%dy	m
Δz	Thickness%dz	m
Δx^E	Grid%dxte	m
Δy^N	Grid%dytn	m
Δz^b	Thickness%dzwt	m
$\Delta x^{(0)}$	Grid%dtw	m
$\Delta x^{(1)}$	Grid%dte	m
$\Delta y^{(0)}$	Grid%dts	m
$\Delta y^{(1)}$	Grid%dttn	m
$\Delta z^{(0)}$	Thickness%dzttup	m
$\Delta z^{(1)}$	Thickness%dzttlo	m
ΔA	Grid%dat	m
D	Thickness%depth_zt	m
M	Grid%tmask	-
L_g	grid_length	m
g	grav	ms^{-2}
ρ_0	rho0	kg m^{-3}
π	pi	-
Ω_e	omega_earth	s^{-1}
ϕ	Grid%phit	rad
ϵ	epsln	context dependent
$\Delta \tau$	Time_steps%dttime_t	s
S_{\max}	smax	-

23.9.2 Tracer Gradients

Value	Variable	Units
T_n	T_prog(n)%field(:, :, :, Time%taum1)	$[T_n]$
$(\delta_x T_n)^E$	dTdx(:, :, :, n)	$[T_n]\text{m}^{-1}$
$(\delta_y T_n)^N$	dTdy(:, :, :, n)	$[T_n]\text{m}^{-1}$
$(\delta_z T_n)^b$	dTdz(:, :, :, n)	$[T_n]\text{m}^{-1}$

23.9.3 Density Calculations

Value	Variable	Units
$(\delta_x \theta)^E$	dTdx(:,:,:,index_temp)	Km ⁻¹
$(\delta_y \theta)^N$	dTdy(:,:,:,index_temp)	Km ⁻¹
$(\delta_z \theta)^b$	dTdz(:,:,:,index_temp)	Km ⁻¹
$(\delta_x S)^E$	dTdx(:,:,:,index_salt)	psu m ⁻¹
$(\delta_y S)^N$	dTdy(:,:,:,index_salt)	psu m ⁻¹
$(\delta_z S)^b$	dTdz(:,:,:,index_salt)	psu m ⁻¹
ρ_θ	Dens%drhodT	kg m ⁻³ K ⁻¹
ρ_s	Dens%drhodS	kg m ⁻³ psu ⁻¹
$\delta_x \rho$	drhodx_x	kg m ⁻⁴
$\delta_y \rho$	drhody_y	kg m ⁻⁴
$\delta_z \rho$	drhodz_z	kg m ⁻⁴
κ_g	smooth_diff	m ² s ⁻¹
S_x	slope_x	-
S_y	slope_y	-
N^2	N2	s ⁻²
N	buoy_freq	s ⁻¹
c_1	gravity_wave_speed	ms ⁻¹
λ_1	rossby_radius	m
R_{raw}	rossby_radius_raw	m
R_{non-eq}	rossby_non_equator	m
R_{eq}	rossby_equator	m

23.9.4 Neutral Boundary Layer

Value	Variable	Units
D_E	eddy_depth	m
D_{BL}	depth_blayer_base	m
k_{surf}	ksurf_blayer	-
S_{base}	slope_blayer_base	-

23.9.5 Diffusivity

Value	Variable	Units
κ	agm_array	m^2s^{-1}
A_R	aredi_array	m^2s^{-1}
L	agm_length	m
σ	growth_rate	ms^{-1}
L_{Rhi}	rhines_length	m
R_{bc}	bczone_radius	m
N_{ref}	buoy_ref	s^{-1}
NS	raw_growth_rate	ms^{-1}
NS	growth_rate_zave	ms^{-1}
σ_{max}	growth_rate_max	ms^{-1}
D_M	ml_depth	m
NS_M	ave_ml_rate	ms^{-1}
σ_{EG}	eg_rate	ms^{-1}
$\Delta x^{[E]}$	e_zone	m
$\Delta x^{[W]}$	w_zone	m
$\Delta y^{[N]}$	n_zone	m
$\Delta y^{[S]}$	s_zone	m
$\Delta x^{[EW]}$	ewtot	m
$\Delta x^{[NS]}$	nstot	m

23.9.6 Tensors

Value	Variable	Units
$\frac{1}{\Delta A^T}$	Grid%datr	m^{-2}
$ f $	coriolis_param	s^{-1}
β	beta_param	$(\text{ms})^{-1}$
T_{sine}	depth_taper	-
T_{\tanh}^x	slope_taper_xz	-
T_{\tanh}^y	slope_taper_yz	-
T_N^x	neutral_taper_xz	-
T_N^y	neutral_taper_yz	-

23.9.7 Fluxes

Value	Variable	Units
$dz(\rho T_n)_{,t}^{\text{diff}}$	diff_th_tendency(n)	$\rho m T s^{-1}$
$dz(\rho T_n)_{,t}^{\text{skew}}$	skew_th_tendency(n)	$\rho m T s^{-1}$
$(G_n^d)^E$	diff_flux_x_xte(n)	$T k g s^{-1}$
$(G_n^d)^N$	diff_flux_y_ytn(n)	$T k g s^{-1}$
$(G_n^d)^b$	diff_flux_z_ztb(n)	$T k g s^{-1}$
$(\rho F_x^n \Delta A)^{\text{skew}}$	skew_flux_x_xte	$T k g s^{-1}$
$(\rho F_y^n \Delta A)^{\text{skew}}$	skew_flux_y_ytn	$T k g s^{-1}$
$(\rho F_z^n \Delta A)^{\text{skew}}$	skew_flux_z_ztb	$T k g s^{-1}$
$(\rho V K_{11})$	symm_mass_diff_11_xz	$k g m^2 s^{-1}$
$(\rho V K_{22})$	symm_mass_diff_22_yz	$k g m^2 s^{-1}$
$(\rho V K_{33})$	symm_mass_diff_33_z	$k g m^2 s^{-1}$
$(\rho V K_{13})$	symm_mass_diff_13_xz	$k g m^2 s^{-1}$
$(\rho V K_{23})$	symm_mass_diff_23_yz	$k g m^2 s^{-1}$
$(\rho V A_{13})$	skew_mass_diff_13_xz	$k g m^2 s^{-1}$
$(\rho V A_{23})$	skew_mass_diff_23_yz	$k g m^2 s^{-1}$
$(\rho V K_{11})^E$	symm_mass_diff_11_xte	$k g m^2 s^{-1}$
$(\rho V K_{22})^N$	symm_mass_diff_22_ytn	$k g m^2 s^{-1}$
$(\rho V K_{13})^E$	symm_mass_diff_h_xz	$k g m^2 s^{-1}$
$(\rho V K_{23})^N$	symm_mass_diff_h_yz	$k g m^2 s^{-1}$
$(\rho V K_{31})^b$	symm_mass_diff_v_xz	$k g m^2 s^{-1}$
$(\rho V K_{32})^b$	symm_mass_diff_v_yz	$k g m^2 s^{-1}$
$(\rho V A_{13})^E$	skew_mass_diff_h_xz	$k g m^2 s^{-1}$
$(\rho V A_{23})^N$	skew_mass_diff_h_yz	$k g m^2 s^{-1}$
$(\rho V A_{31})^b$	skew_mass_diff_v_xz	$k g m^2 s^{-1}$
$(\rho V A_{32})^b$	skew_mass_diff_v_yz	$k g m^2 s^{-1}$
$(\rho V A_R)^E$	simple_mass_diff_11_xte	$k g m^2 s^{-1}$
$(\rho V A_R)^N$	simple_mass_diff_22_ytn	$k g m^2 s^{-1}$
(ρV_x)	rho_qcv_xz	$k g$
(ρV_y)	rho_qcv_yz	$k g$
$(\Delta z')^E$	min_dzt_xte_z	m
$(\Delta z')^N$	min_dzt_ytn_z	m
ρ	rho	$k g m^{-3}$
$(\rho V K_{33}^{\text{crit}})$	m33_crit_z	$k g m^2 s^{-1}$
$(\rho V K_{33}^{\text{exp}})$	m33_explicit_z	$k g m^2 s^{-1}$
$(\rho V K_{33}^{\text{imp}})$	m33_implicit	$k g m^2 s^{-1}$
$(\rho V K_{33}^{\text{exp}})^b$	m33_explicit_ztb	$k g m^2 s^{-1}$
$(\rho K_{33}^{\text{imp}})^n$	T_prog(n)%K33_implicit	$k g m^{-1} s^{-1}$

RESTRATIFICATION BY SUBMESOSCALE EDDIES

Contents

24.1	Basics of the scheme	347
24.2	Skew tracer flux components	349
24.3	Eddy induced transport	350
24.3.1	Overturning circulation	350
24.3.2	Diagnosing the streamfunction in Ferret from MOM output	351
24.3.3	Advective tracer fluxes	351
24.4	Eddy advection implementation	352
24.5	Cautionary remarks on compute_psi_legacy	352
24.6	Horizontal diffusion associated with submesoscale processes	353

This chapter documents the MOM implementation of the parameterization by [Fox-Kemper et al. \(2008b\)](#), with further details discussed in [Fox-Kemper et al. \(2008a\)](#) and [Fox-Kemper et al. \(2011\)](#). This scheme parameterizes the restratification effects of submesoscale eddies in the ocean mixed layer. These effects occur on a time scale much shorter than the mesoscale eddies parameterized via the neutral physics scheme of [Gent and McWilliams \(1990\)](#) and [Gent et al. \(1995\)](#). The following MOM module is directly connected to the material in this chapter:

ocean_param/lateral/ocean_submesoscale.F90.

24.1 Basics of the scheme

The parameterization is based on the calculation of a vector streamfunction Ψ (with units $\text{length}^2 \text{ time}^{-1}$)

$$\Psi = \left(\frac{C_e \mu h^2 g \Delta s}{\rho_o L_f \sqrt{f^2 + \tau^{-2}}} \right) \hat{\mathbf{z}} \wedge \overline{\nabla \gamma^z}. \quad (24.1)$$

In this equation, we have

- $0.06 \leq C_e \leq 0.08$ is a dimensionless number;
- $\mu = [1 - (1 - 2d/h)^2][1 + 5/21(1 - 2d/h)^2]$ is a non-negative vertical structure function $0 \leq \mu \leq 1$ in the mixed layer, with zero values outside the mixed layer where $d \geq h$ and $d \leq 0$;
- $d = -z + \eta$ is the depth of seawater, defined as the vertical distance from the ocean free surface;
- η is the deviation of the ocean free surface from the resting state at $z = 0$;

- g is the gravitational acceleration;
- h is the mixed layer thickness;
- $\overline{\nabla \gamma^z}$ is the mixed layer averaged horizontal gradient of locally referenced potential density;
- f is the Coriolis parameter;
- τ is a time scale for the submesoscale eddies (order few days);
- L_f is a length scale for the width of the submesoscale eddies (order 5km);
- Δs is the horizontal grid spacing;
- ρ_o is the constant Boussinesq density.

Written in components, the streamfunction is given by

$$\Psi = \Gamma \mu(-\overline{\gamma_{,y}^z}, \overline{\gamma_{,x}^z}, 0), \quad (24.2)$$

where

$$\Gamma = \frac{C_e h^2 g \Delta s}{\rho_o L_f \sqrt{f^2 + \tau^{-2}}} \quad (24.3)$$

is shorthand for the non-negative dimensionful scalar contributions, with physical dimensions $\text{m}^6/(\text{sec kg})$. The dimensionless function $\mu(z)$ carries the only vertical dependence of the streamfunction Ψ . We make the following observations.

- The formulation of [Fox-Kemper et al. \(2008b\)](#) is in terms of the mixed layer depth, rather than the planetary boundary layer depth. The reasoning is that the planetary boundary layer can be very small under stable buoyancy forcing. However, the submesoscale eddies remain even in these situations, so long as the mixed layer depth is nontrivial. MOM has an option for setting h according to either the mixed layer depth or the planetary boundary layer depth, with the mixed layer depth the recommended choice.
- [Fox-Kemper et al. \(2008b\)](#) pose the parameterization in terms of the horizontal buoyancy gradient, which is related to the gradient of the locally referenced potential density via

$$-\rho_o \nabla b = g \nabla \gamma \quad (24.4)$$

$$= g(\gamma_{,\theta} \nabla \theta + \gamma_{,S} \nabla S), \quad (24.5)$$

where θ is the potential temperature, S is the salinity, and

$$\gamma_{,\theta} = \frac{\partial \gamma}{\partial \theta} \quad (24.6)$$

$$\gamma_{,S} = \frac{\partial \gamma}{\partial S} \quad (24.7)$$

are the density partial derivatives.

- The front length can either be a constant, or computed according to the first baroclinic Rossby radius over the mixed layer

$$L_f = \frac{h \overline{N^z}}{f} \quad (24.8)$$

with $\overline{N^z}$ the buoyancy frequency averaged over the depth of the mixed layer, and h is the mixed layer thickness. Substitution of the front length (24.8) into the streamfunction (24.1) leads to

$$\Psi = \left(\frac{C_e \mu h g \Delta s}{\rho_o \overline{N^z}} \right) \left(\frac{f}{\sqrt{f^2 + \tau^{-2}}} \right) \hat{\mathbf{z}} \wedge \overline{\nabla \gamma^z}. \quad (24.9)$$

In this way, the streamfunction goes to zero as the equator is approached.

24.2 Skew tracer flux components

A method for implementing the effects of the streamfunction (24.1) on tracer concentration is to do so via a skew diffusion, as motivated by the Griffies (1998) method for the Gent et al. (1995) scheme. Skew diffusion for the submesoscale restratification parameterization is defined by an anti-symmetric stirring tensor

$$\mathbf{A} = \Gamma \mu \begin{pmatrix} 0 & 0 & -\overline{\gamma_{,x}^z} \\ 0 & 0 & -\overline{\gamma_{,y}^z} \\ \overline{\gamma_{,x}^z} & \overline{\gamma_{,y}^z} & 0 \end{pmatrix} \quad (24.10)$$

acting on the gradient of the tracer concentration, so that

$$F^i = -A^{ij} \left(\frac{\partial C}{\partial x^j} \right), \quad (24.11)$$

with summation over $j = 1, 2, 3$ implied on the right hand side. Components to the skew tracer flux are given by

$$F^{(x)} = \Gamma \mu \overline{\gamma_{,x}^z} C_{,z} \quad (24.12)$$

$$F^{(y)} = \Gamma \mu \overline{\gamma_{,y}^z} C_{,z} \quad (24.13)$$

$$F^{(z)} = -\Gamma \mu \overline{\nabla \gamma^z} \cdot \nabla C, \quad (24.14)$$

which can be written in terms of the vector streamfunction

$$F^{(x)} = \Psi^{(y)} C_{,z} \quad (24.15)$$

$$F^{(y)} = -\Psi^{(x)} C_{,z} \quad (24.16)$$

$$F^{(z)} = -\Psi^{(y)} C_{,x} + \Psi^{(x)} C_{,y}. \quad (24.17)$$

It is revealing to consider the special case of potential temperature for a linear equation of state

$$\gamma = \rho_o - \alpha \theta, \quad (24.18)$$

with $\alpha > 0$ constant. In this case, the skew temperature flux components are given by

$$\mathbf{F}^{(h)} = -\Gamma \mu \alpha \theta_{,z} \overline{\nabla \theta^z} \quad (24.19)$$

$$F^{(z)} = \Gamma \mu \alpha \nabla \theta \cdot \overline{\nabla \theta^z}. \quad (24.20)$$

With a stable stratification where $\theta_{,z} > 0$, the horizontal flux is directed opposite to the vertically averaged horizontal temperature gradient: $\mathbf{F}^{(h)} \propto -\overline{\nabla \theta^z}$, and so is downgradient in this sense.

The technology for discretizing the neutral physics operators (e.g., neutral diffusion as in Griffies et al. (1998), and skew diffusion from Gent et al. (1995) as in Griffies (1998)) is useful for rotated diffusion and skew diffusion. We can thus make use of this technology for discretizing the submesoscale closure of Fox-Kemper et al. (2008b). The full tracer transport tensor, which is the sum of neutral diffusion plus Gent et al. (1995) skew diffusion plus submesoscale restratification, is given by

$$\mathbf{J} = \begin{pmatrix} A_I & 0 & (A_I - \kappa) S^{(x)} - \Gamma \mu \overline{\gamma_{,x}^z} \\ 0 & A_I & (A_I - \kappa) S^{(y)} - \Gamma \mu \overline{\gamma_{,y}^z} \\ (A_I + \kappa) S^{(x)} + \Gamma \mu \overline{\gamma_{,x}^z} & (A_I + \kappa) S^{(y)} + \Gamma \mu \overline{\gamma_{,y}^z} & A_I S^2 \end{pmatrix} \quad (24.21)$$

$$= \begin{pmatrix} A_I & 0 & (A_I - \kappa) S^{(x)} - \Psi^{(y)} \\ 0 & A_I & (A_I - \kappa) S^{(y)} + \Psi^{(x)} \\ (A_I + \kappa) S^{(x)} + \Psi^{(y)} & (A_I + \kappa) S^{(y)} - \Psi^{(x)} & A_I S^2 \end{pmatrix}, \quad (24.22)$$

where A_I is the neutral diffusivity, κ is the GM-diffusivity, and \mathbf{S} is the neutral slope vector. It is straightforward to incorporate the submesoscale closure into the neutral physics module. If doing so, one would

then implement the full transport tensor (24.22). However, there are occasions where one does not wish to turn on the neutral physics parameterizations. For example, in mesoscale eddying simulations, one may choose to remove the neutral physics closure, but retain the submesoscale closure. We thus prefer to develop a separate module for the submesoscale closure, taking only what we need from the neutral physics module.

24.3 Eddy induced transport

The vector streamfunction (24.1) gives rise to an eddy induced velocity

$$\begin{aligned}\mathbf{v}^* &= \nabla \wedge \Psi \\ &= (-\partial_z \Psi^{(y)}, \partial_z \Psi^{(x)}, \partial_x \Psi^{(y)} - \partial_y \Psi^{(x)})\end{aligned}\quad (24.23)$$

and an associated volume transport within the mixed layer. There is zero net horizontal volume transport

$$\int_{-H}^{\eta} dz \mathbf{u}^* = 0, \quad (24.24)$$

since the vector streamfunction vanishes at the ocean surface and at the base of the mixed layer. A zero net volume transport is also the case for the eddy induced transport from the Gent et al. (1995) mesoscale parameterization, where the Gent et al. (1995) quasi-Stokes streamfunction vanishes at the ocean surface and bottom (McDougall and McIntosh, 2001).

24.3.1 Overturning circulation

For a vertical position z within the mixed layer, the meridional volume transport passing beneath this depth z , zonally integrated within a basin or over the globe, is computed by the integral (see Section 32.2.3)

$$\begin{aligned}T^{(y)}(y, z, t) &= - \int dx \int_{-H}^z dz' v^* \\ &= - \int dx \int_{-h}^z dz' \partial_z \Psi^{(x)} \\ &= - \int dx \Psi^{(x)}(x, y, z, t) \\ &= \int dx \Gamma \mu \overline{\gamma}_{,y}^z\end{aligned}\quad (24.25)$$

since the streamfunction vanishes at $z \leq -h$ (beneath the mixed layer). Likewise, the zonal transport within the mixed layer is

$$\begin{aligned}T^{(x)}(x, z, t) &= - \int dy \int_{-H}^z dz' u^* \\ &= \int dy \int_{-h}^z dz' \partial_z \Psi^{(y)} \\ &= \int dy \Psi^{(y)}(x, y, z, t) \\ &= \int dy \Gamma \mu \overline{\gamma}_{,x}^z.\end{aligned}\quad (24.26)$$

Since $\Gamma \mu$ is single signed within the mixed layer, the sign of the horizontal transport in the mixed layer is given by the sign of $-\overline{\gamma}^z$. For example, with denser water towards the north, so that $\overline{\gamma}_{,y}^z > 0$, the meridional transport passing beneath a depth z within the mixed layer will be negative, $T^{(y)}(y, z, t) < 0$. This property of the transport provides a useful check that the scheme has been implemented in the model with the proper sign.

It is instructive to compare the volume transport from the submesoscale parameterization to that induced by the [Gent et al. \(1995\)](#) mesoscale scheme. For [Gent et al. \(1995\)](#), the horizontal component of the eddy-induced velocity is

$$\mathbf{u}^{\text{gm}} = -\partial_z (\kappa \mathbf{S}), \quad (24.27)$$

with $\mathbf{S} = -\nabla \gamma / \gamma_{,z}$ the neutral slope vector, and $\kappa > 0$ a diffusivity. The meridional transport is given by

$$\begin{aligned} T_{\text{gm}}^{(y)} &= - \int dx \int_{-H}^z dz' v^{\text{gm}} \\ &= - \int dx \left(\kappa \frac{\gamma_{,y}}{\gamma_{,z}} \right), \end{aligned} \quad (24.28)$$

where we set the streamfunction to zero at the ocean bottom. For a stable stratification with $\gamma_{,z} < 0$, the volume transport is directed in the sign of the meridional density gradient, which is analogous to the case for the submesoscale transport ([24.25](#)).

24.3.2 Diagnosing the streamfunction in Ferret from MOM output

To evaluate the overturning streamfunction when implementing the submesoscale closure using skew fluxes, we perform the following step in Ferret

$$T^{(y)}(y, z, t) = \text{ty_trans_submeso}[i = @sum] \quad \text{skew fluxes}, \quad (24.29)$$

which is just as for the skew flux implementation of [Gent et al. \(1995\)](#). Note that by definition, the vertically integrated transport from the submesoscale parameterization scheme vanishes. So there is no contribution from this scheme to the column integrated transport through straits and throughflows.

If implementing the submesoscale closure using advective fluxes (still under testing so not the default method), then the overturning should be computed in Ferret using the commands

$$T^{(y)}(y, z, t) = \text{ty_trans_submeso}[i = @sum, k = @rsum] - \text{ty_trans_submeso}[i = @sum, k = @sum] \quad \text{advective fluxes}, \quad (24.30)$$

which is just as for computing the streamfunction based on the resolved Eulerian flow ([Section 32.2.6](#)). Subtraction of the depth integral in equation ([24.30](#)) is necessary since Ferret starts the vertical sum from the surface ($k=1$) rather than from the bottom.

24.3.3 Advective tracer fluxes

Components to the advective tracer flux are given by

$$F^{(x)} = u^* C \quad (24.31)$$

$$F^{(y)} = v^* C \quad (24.32)$$

$$F^{(z)} = w^* C, \quad (24.33)$$

which can be written in terms of the vector streamfunction

$$F^{(x)} = - \left(\frac{\partial \Psi^{(y)}}{\partial z} \right) C \quad (24.34)$$

$$F^{(y)} = \left(\frac{\partial \Psi^{(x)}}{\partial z} \right) C \quad (24.35)$$

$$F^{(z)} = \left(\frac{\partial \Psi^{(y)}}{\partial x} - \frac{\partial \Psi^{(x)}}{\partial y} \right) C. \quad (24.36)$$

Comparison should be made to the corresponding skew flux components given by equations ([24.15](#))–([24.17](#)). Note that the vertical structure for the horizontal advective transport velocity components is given by the vertical derivative of the structure function

$$\frac{\partial \mu}{\partial z} = - \left(\frac{1 - 2d/H}{21 H} \right) (64 + 40(1 - 2d/H)^2), \quad (24.37)$$

where again $d = -z + \eta$ and H is the mixed layer depth.

24.4 Eddy advection implementation

The skew flux approach in Section 24.2 is the standard method used in MOM, as documented and illustrated in Fox-Kemper et al. (2011). However, the skew flux suffers from a lack of monotonicity constraints, with such allowing for the introduction of spurious extrema. There are some cases when this non-monotonicity may become a problem, especially with finer resolution simulations where gradients are stronger.

We have not pursued the development of monotonicity constraints for skew fluxes. Instead, we have provided an option in MOM to compute the advective flux using a monotonic advection scheme. For this purpose, it is necessary to compute the eddy induced transport at the tracer cell faces, with the continuum form given in Section 24.3.3. To compute the vertical derivative of the vector streamfunction, we perform the analytical derivative of the vertical structure function and evaluate that derivative at the model depths. Use of the continuity equation, as for the resolved Eulerian mass transport, then allows for the vertical component to be diagnosed.

Once all three advective mass transport components are known, one can then use the transports to compute tracer flux components via an advection scheme. The first order upwind scheme is presently the only working scheme implemented in MOM for the submesoscale parameterization. Tests with the Sweby scheme are incomplete. Either way, the use of a dissipative advection implementation of the submesoscale parameterization introduces some mixing that is absent when using the skew flux approach. There are two reasons we favour the introduction of enhanced mixing for this parameterization.

- The submesoscale parameterization in theory adds an extra advective transport to the tracer fields in the mixed layer. It is arguable (Baylor Fox-Kemper, personal communication, 2011) that the parameterization should *also* provide some added lateral mixing in the mixed layer. The use of dissipative advection schemes acts, through numerical truncation errors, to add such mixing.
- Because the submesoscale parameterization acts only in the mixed layer, there is little concern that adding mixing through dissipative advection will adversely compromise the simulation. Indeed, given the many sources for grid noise arising from this scheme, the addition of mixing from dissipative advection in many ways enhances the physical integrity of the scheme.

There are many issues remaining to be resolved with the advective approach. Notably, in the presence of topography, the approach has been found to produce some rather large tendencies near topography. We have implemented a masking scheme to remove the fluxes at cells next to land. But that scheme does not fully remove the problems. It is unclear whether this issue is a “bug” or a “feature” of the advective implementation.

24.5 Cautionary remarks on `compute_psi_legacy`

The preferred organization of the numerical implementation of the submesoscale parameterization is as follows:

- Compute the depth-independent portion of the vector streamfunction. Perform any desired smoothing or limiting directly on this depth-independent portion.
- Project the depth-independent portion vertically by multiplying by the vertical structure function $\mu(z)$.

This is the overall philosophy taken with the MOM code as of Feb 2012. Unfortunately, earlier implementations computed the three-dimensional streamfunction and applied limiters and smoothers to the three-dimensional streamfunction. The problem with this approach is that the analytical properties of the vertical structure are generally corrupted when smoothing or limiting on the full three-dimensional streamfunction. Namely, both limiting and smoothing can produce multiple extrema in the vertical, as opposed to the desired unimodal structure. Here are more details of the problems.

- **PROBLEMS WITH `smooth_psi`:** A numerical calculation of the vector streamfunction can at times introduce noise in the horizontal directions, so that a horizontal smoothing operator is motivated. However, examples have been found where smoothing the three-dimensional streamfunction produces multiple extrema in the vertical, rather than a single extrema designed into the analytical unimodal structure function $\mu(z)$. The problem of multiple extrema arises from horizontal smoothing of the vector streamfunction, via `smooth_psi`, across regions of different mixed layer depths. This lateral smoothing mixes across the different structure functions, resulting in the potential for multiple extrema.
- **PROBLEMS WITH `limit_psi_legacy`:** The vector streamfunction can at times become quite large. If the vertical gradient is correspondingly large, the model can be prone to instabilities. MOM provides the following means to limit the value of the streamfunction magnitude

$$\Psi_{\text{limited}} = \text{sign}(\Psi) \min[v_{\text{max}} \Delta z, \text{abs}(\Psi)], \quad (24.38)$$

where v_{max} is a namelist-specified velocity scale on the order of 1m/sec, and Δz is the grid cell thickness. This limiter is not appropriate, since it is not the magnitude of the streamfunction that is problematic, it is instead the vertical derivative. Nonetheless, it is what was originally coded.

Consistent with this limiter being naive, it can introduce some rather unphysical side-effects. Depending on the value of v_{max} and Δz , examples have been found where `limit_psi` sets the streamfunction to a vertical constant in the upper ocean where the vertical grid is fine (i.e., small Δz), yet returns the streamfunction to its normal values in the region where vertical thickness coarsens (larger Δz). In effect, this corrupted streamfunction places all restratification effects into the deeper ocean region where the streamfunction has a jump, and removes restratification from the upper ocean where the streamfunction is a vertical constant. Both effects are in fact opposite the theoretical intention of the [Fox-Kemper et al. \(2008b\)](#) scheme, so that `limit_psi` can be particularly egregious at corrupting the submesoscale scheme.

For the above reasons, `compute_psi_legacy` should be avoided if one also aims to use the `limit_psi` or `smooth_psi` options. A far more sensible approach is in the newer `compute_psi` routine, where limiters and smoothers are first applied to the depth independent portion of the streamfunction. After all such operations are performed, the vertical structure function is applied to render the three-dimensional streamfunction. In this manner, the proper analytic properties of the streamfunction are maintained, so there are no spurious jumps in the vertical derivative.

24.6 Horizontal diffusion associated with submesoscale processes

We raised some issues in Sections 24.4 and 24.5 regarding the potential for the submesoscale parameterization to introduce spurious extrema. The presence of these extrema raises the issue whether the parameterization should *also* include a mixing parameterized as downgradient diffusion. Adding such mixing will in principle suppress the occurrences of spurious dispersion that create extrema. It also acknowledges that the submesoscale processes create both an advective restratification as well as a lateral diffusive effect, both within the upper ocean mixed layer. Such arguments are analogous to those used by [Treguier et al. \(1997\)](#), [Ferrari et al. \(2008\)](#), and [Ferrari et al. \(2010\)](#) for parameterizing mesoscale eddy processes in the upper ocean boundary layer, in which an additional horizontal diffusion is added to the eddy-induced advection.

For the above reasons, MOM has an option within the submesoscale module that provides for a nonzero horizontal tracer diffusion in addition to the eddy-induced advection from the vector streamfunction. This option, `submeso_horz_diffuse`, is implemented using the magnitude of the vector streamfunction (units of $\text{m}^2 \text{sec}^{-1}$) to specify an effective diffusivity, in which case the downgradient horizontal diffusive flux for all tracers is given by

$$\mathbf{F}^{(\text{submeso})} = -\rho \gamma |\Psi| \nabla_s C, \quad (24.39)$$

where γ is a dimensionless scaling parameter that can be used to increase or decrease the diffusivity beyond that set by $|\Psi|$.

LATERAL FRICTION METHODS

Contents

25.1	Introduction	356
25.2	Lateral friction options in MOM	356
25.2.1	Older options just for the B-grid	356
25.2.2	Laplacian and biharmonic modules for the B and C grids	357
25.3	Continuum formulation for the friction operator	357
25.3.1	Laplacian friction	357
25.3.2	Biharmonic friction	358
25.4	Lateral friction operator for B-grid MOM	359
25.4.1	Discrete dissipation functional for the B-grid	359
25.4.2	Grid cell distances and subcell volumes	360
25.4.3	Derivative operators	360
25.4.4	Tension and strain for the subcells	360
25.4.5	Functional derivative of e_T and e_S	363
25.4.6	Tidy form for the discretized friction	364
25.4.7	Tension and strain in the quadrants	365
25.4.8	Comments	366
25.4.9	Discretized Smagorinsky viscosity	366
25.5	Lateral friction operator for C-grid MOM	367
25.5.1	Discrete functional derivative and the grid stencil	367
25.5.2	Deformation rates	367
25.5.3	Stress tensor components	368
25.5.4	Orientation angle	369
25.5.5	Viscosities	370
25.5.6	Volumes	370
25.5.7	Discrete C-grid friction operators	370
25.6	Boundary conditions	371
25.6.1	B-grid	371
25.6.2	C-grid	371

The purpose of this chapter is to summarize methods for implementing lateral viscous friction in MOM.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_param/lateral/ocean_lapcst_friction.F90
ocean_param/lateral/ocean_lapgen_friction.F90
ocean_param/lateral/ocean_lapcgrid_friction.F90
ocean_param/lateral/ocean_bihcst_friction.F90
ocean_param/lateral/ocean_bihgen_friction.F90
ocean_param/lateral/ocean_bihcgrid_friction.F90
```

Some of the material in this chapter can also be found in Part 5 of [Griffies \(2004\)](#).

25.1 Introduction

Ocean models require some form of viscous friction for numerical reasons. One way to understand the numerical need is that the Reynolds number for traditional advection schemes should be kept less than unity, with a length scale given by the model grid scale used to set the Reynolds number. With grid scales of many kilometers rather than the millimeters corresponding to the ocean's Kolmogorov scale, numerical viscosity must be increased accordingly. For the most part, ocean modelers employ lateral viscous friction as a numerical closure. As a numerical closure, the aim is to reduce friction to low levels while suppressing numerical instabilities and/or noise. Hence, lateral friction is generally *tuned*. Experience with global models reveals a nontrivial sensitivity to details of friction. For example, friction affects transport through frictionally controlled passages, modifies the Gulf Stream separation point ([Chassignet and Garraffo, 2001](#)), strongly influences the equatorial undercurrent strength ([Large et al., 2001](#)), and changes poleward heat transport by altering transport in boundary currents and gyres. In practice, friction tuning can be one of the least satisfying aspects of ocean climate model construction.

Given the need to apply friction for numerical reasons, is it necessary to damp toward a state of zero kinetic energy? Certainly that is what molecular viscosity does, but the ocean models are so far from resolving the Kolmogorov scale that analogies to molecular friction are unwarranted. Nonetheless, damping toward rest is the common approach. There are, however, arguments that suggest dissipation should instead damp the solution toward a nonzero flow state. [Holloway \(1992\)](#) proposes a practical means to realize this approach, with Chapter 26 detailing the MOM implementation.

25.2 Lateral friction options in MOM

MOM has both Laplacian and biharmonic friction operator options. Both operators can be employed at the same time, such as used in the one-degree global model documented by [Dunne et al. \(2012\)](#) and [Dunne et al. \(2013\)](#). Most applications with finer resolution simulations choose just the biharmonic operator, such as in [Farneti et al. \(2010\)](#) and [Delworth et al. \(2012\)](#). Nonetheless, there are exceptions, such as the combined Laplacian and biharmonic approaches used for the mesoscale eddying simulations described by [Hecht et al. \(2008\)](#).

25.2.1 Older options just for the B-grid

The Laplacian and biharmonic friction modules

- `ocean_param/lateral/ocean_lapcst_friction.F90`
- `ocean_param/lateral/ocean_bihcst_friction.F90`

are based on the discretization method used in MOM prior to the year 2000. These two modules are largely frozen, with minimal support for further development, and they are supported only for use with the B-grid option in MOM. Consequently, options for computing the viscosity are somewhat limited when using these two modules.

25.2.2 Laplacian and biharmonic modules for the B and C grids

The friction modules

- `ocean_param/lateral/ocean_lapgen_friction_bgrid.F90`
- `ocean_param/lateral/ocean_bihgen_friction_bgrid.F90`
- `ocean_param/lateral/ocean_lapgen_friction_cgrid.F90`
- `ocean_param/lateral/ocean_bihgen_friction_cgrid.F90`

employ the B-grid and C-grid discretization methods of [Griffies and Hallberg \(2000\)](#). The B-grid methods are detailed in Chapter 19 of [Griffies \(2004\)](#) as well as Section 25.4 below. The C-grid methods are detailed in Section 25.5.

Each of the Laplacian and biharmonic modules support the following suite of options for computing the viscosity.

- The deformation dependent viscosity scheme of [Smagorinsky \(1963, 1993\)](#), as detailed by [Griffies and Hallberg \(2000\)](#) (see also Chapter 18 of [Griffies \(2004\)](#)) is available, and used quite frequently at GFDL.
- The anisotropic Laplacian and biharmonic scheme of [Large et al. \(2001\)](#); [Smith and McWilliams \(2003\)](#) (also see Chapter 17 of [Griffies \(2004\)](#)) has been used for the AR4 climate models at GFDL ([Griffies et al., 2005](#)).
- The Neptune scheme of [Holloway \(1992\)](#), in the form implemented by [Maltrud and Holloway \(2008\)](#) (see Chapter 26), is available. Tests with this scheme are sparse at GFDL, though many groups make use of it, particularly in the Arctic.
- The side-drag scheme from [Deremble et al. \(2012\)](#) introduces a side-wall drag scheme following from the drag coefficient used for computing bottom drag. It has been implemented for both the MOM B-grid and C-grid. Incomplete tests with this scheme on a B-grid indicate some sensitivity of western boundary current separation.
- Various static viscous profiles, in particular a method to enhance the viscosity next to western boundaries, have been implemented, and are readily modified for specific purposes. In particular, one may read in a static background viscosity from a file.

25.3 Continuum formulation for the friction operator

We summarize here the continuum formulation for Laplacian and biharmonic friction.

25.3.1 Laplacian friction

As shown in Section 17.8.4 of [Griffies \(2004\)](#), the components to the horizontal friction operator, $F^{(a)}$, are proportional to the functional derivative of the dissipation functional

$$\rho F^{(a)} = -\frac{1}{2dV} \left(\frac{\partial \mathcal{S}}{\partial u^{(a)}} \right), \quad (25.1)$$

where $a = 1, 2$ label the horizontal directions, dV is the volume element, and $(u^{(1)}, u^{(2)}) = (u, v)$ is the horizontal velocity. The dissipation functional \mathcal{S} is given by the volume integral (equation (17.130) of [Griffies \(2004\)](#))

$$\mathcal{S} = \int \left[A(e_T^2 + e_S^2) - 2D\Delta^2 \right] \rho dV, \quad (25.2)$$

where A is the lateral isotropic viscosity, $D < 2A$ (equation (17.105) of Griffies (2004)) is the lateral anisotropic viscosity, and

$$2\Delta = e_S \cos 2\theta - e_T \sin 2\theta \quad (25.3)$$

orients the friction operator relative to an angle θ and an orientation unit vector $\hat{s} = (\cos \theta, \sin \theta)$ (see equations (17.99) and (17.101) of Griffies (2004)). Fundamental to the calculation of lateral friction are the horizontal tension and shear deformation rates

$$e_T = dy \left(\frac{\partial(u/dy)}{\partial x} \right) - dx \left(\frac{\partial(v/dx)}{\partial y} \right) \quad (25.4)$$

$$e_S = dx \left(\frac{\partial(u/dx)}{\partial y} \right) + dy \left(\frac{\partial(v/dy)}{\partial x} \right), \quad (25.5)$$

with the grid factors dx and dy generally non-constant on the sphere. After some manipulations (see Section 17.20.3 of Griffies (2004)), we are led to the friction operator arising from lateral shears on the sphere

$$\rho F^x = \frac{1}{(dy)^2} \left(\frac{\partial[(dy)^2 \tau^{xx}]}{\partial x} \right) + \frac{1}{(dx)^2} \left[\frac{\partial[(dx)^2 \tau^{xy}]}{\partial y} \right] \quad (25.6)$$

$$\rho F^y = \frac{1}{(dx)^2} \left(\frac{\partial[(dx)^2 \tau^{yy}]}{\partial y} \right) + \frac{1}{(dy)^2} \left(\frac{\partial[(dy)^2 \tau^{xy}]}{\partial x} \right). \quad (25.7)$$

The symmetric and trace-free friction stress tensor, associated with lateral shears, has components given by (equations (19.7) and (19.8) of Griffies (2004))

$$\tau^{xx} = \rho(A e_T + D \Delta \sin 2\theta) \quad (25.8)$$

$$\tau^{xy} = \rho(A e_S - D \Delta \cos 2\theta) \quad (25.9)$$

$$\tau^{yy} = -\tau^{xx} \quad (25.10)$$

$$\tau^{yx} = \tau^{xy}. \quad (25.11)$$

25.3.2 Biharmonic friction

The general formulation of biharmonic friction is a straightforward extension of the Laplacian friction. What is done is to iterate twice on the Laplacian approach. More precisely, the components F_B^m of the biharmonic friction vector are derived from the covariant divergence¹

$$\rho F_B^m = \Theta_{;n}^{mn} \quad (25.12)$$

where

$$\Theta^{mn} = -\rho B(2E^{mn} - g^{mn} E_p^p), \quad (25.13)$$

$B > 0$ has units of $L^2 t^{-1/2}$. As shown in Section 17.9.2 of Griffies (2004), use of this “square root” biharmonic viscosity is prompted by the desire to dissipate kinetic energy. This detail matters for cases with a nonconstant viscosity. Θ^{mn} has the same form as the stress tensor used for the isotropic Laplacian friction, except with a minus sign. Components of the symmetric “strain” tensor are given by

$$E_{mn} = (F_{m;n} + F_{n;m})/2. \quad (25.14)$$

F^m are components to the friction vector determined through the covariant divergence of the Laplacian frictional stress tensor

$$\begin{aligned} \rho F^m &= \tau_{;n}^{mn} \\ &= [B\rho(2e^{mn} - g^{mn} e_p^p)]_{;n} \end{aligned} \quad (25.15)$$

¹All labels in this section run over $m, n, p = 1, 2$. The semi-colon represents a covariant derivative. Full details are given in Chapter 17 of Griffies (2004).

as derived in the previous sections, where the only difference is that the viscosity used for computing the stress tensor τ^{mn} is now set to B , and the dimensions on F^m are $L t^{-3/2}$.

This approach ensures that the biharmonic friction is derived from the divergence of a symmetric tensor Θ^{mn} , hence ensuring a proper angular momentum budget. Additionally, the computational work necessary to compute the Laplacian friction is directly employed for the biharmonic friction. Finally, as shown in the next subsection, this form for biharmonic friction also dissipates kinetic energy.

25.4 Lateral friction operator for B-grid MOM

Chapter 19 in [Griffies \(2004\)](#) provides a detailed derivation of the discrete friction operator for the B-grid. We present here a summary of that material.

25.4.1 Discrete dissipation functional for the B-grid

The discrete dissipation functional takes the form

$$\begin{aligned} \mathcal{S} &= \sum_{i,j} \sum_{n=1}^{12} V(n) [A(n) (e_T^2(n) + e_S^2(n)) - 2D(n) \Delta^2(n)] \\ &\equiv \sum_{i,j} \mathcal{S}_{i,j}. \end{aligned} \quad (25.16)$$

Figure 25.1 illustrates the nearest neighbor stencil used for discretizing the dissipation functional using second order numerics. The summation $n = 1, 12$ arises from the 12 subcells to which the velocity point $U_{i,j}$ contributes when discretizing the functional on a B-grid. $V(n)$ are the volumes of each of the subcells, $A(n)$ are the viscosities, and $e_T(n)$, $e_S(n)$, and $\Delta(n)$ are the corresponding tensions, strains, and anisotropic contributions, respectively.

The physical component of the friction vector acting at the velocity cell $U_{i,j,k}$ is given by the discrete functional derivative

$$\begin{aligned} -F_{i,j}^{(b)} &= \frac{1}{2V_{i,j}^U} \frac{\partial \mathcal{S}_{i,j}}{\partial (u^{(b)})_{i,j}} \\ &= \frac{1}{V_{i,j}^U} \sum_{n=1}^{12} V(n) \left(A(n) e_T(n) \frac{\partial e_T(n)}{\partial (u^{(b)})_{i,j}} + A(n) e_S(n) \frac{\partial e_S(n)}{\partial (u^{(b)})_{i,j}} - 2D(n) \Delta(n) \frac{\partial \Delta(n)}{\partial (u^{(b)})_{i,j}} \right) \\ &= \frac{1}{\rho_0 V_{i,j}^U} \sum_{n=1}^{12} V(n) \left(\frac{\partial e_T(n)}{\partial (u^{(b)})_{i,j}} \tau^{xx}(n) + \frac{\partial e_S(n)}{\partial (u^{(b)})_{i,j}} \tau^{xy}(n) \right). \end{aligned} \quad (25.17)$$

In this equation, $b = 1, 2$ labels the generalized zonal and meridional directions, the discrete depth label k was dropped since all points are at the same level,

$$\left((u^{(1)})_{i,j}, (u^{(2)})_{i,j} \right) = (u_{i,j}, v_{i,j}) \quad (25.18)$$

are physical components of the velocity vector,

$$V_{i,j,k}^U = dx_{i,j} dy_{i,j} dz_{i,j,k} \quad (25.19)$$

is the velocity cell volume,

$$\tau^{xx}(n) = \rho [A(n) e_T(n) + D(n) \Delta(n) \sin 2\theta] \quad (25.20)$$

$$\tau^{xy}(n) = \rho [A(n) e_S(n) - D(n) \Delta(n) \cos 2\theta] \quad (25.21)$$

are the stress tensor components, and the orientation angle θ determines the orientation of the friction (equation (25.3)). The remainder of this section is devoted to performing the discrete functional derivatives and then manipulating the results into a tidy expression for the discrete friction operator.

25.4.2 Grid cell distances and subcell volumes

Enumerating the 12 quarter cell volumes constitutes an important part of the discretization. Assume knowledge of the distance along each of the four sides of a velocity and tracer cell, as well as the distance from the velocity and tracer points to the sides of their respective cells. Although nonuniform grids with general coordinates do not allow an exact (i.e., analytically exact) area calculation, the horizontal area of the quarter cells is well approximated using this information. Even so, for the purpose of discretizing friction, there is no reason to take pains to compute a very accurate quarter cell area. Indeed, the aim is to realize the dissipative property of friction with minimal computational expense. Hence, make the simplification that the volume of a quarter cell is equal to one-quarter the volume of the corresponding velocity cell. In summary, we employ the following grid information (refer to Figure 25.1):

- The longitudinal spacing, in meters, between $U_{i,j}$ and $U_{i+1,j}$ is $dx_{ue_{i,j}}$. The latitudinal spacing, in meters, between $U_{i,j}$ and $U_{i,j+1}$ is $dy_{un_{i,j}}$.
- The volume of a quarter cell is taken equal to one-quarter the volume of the velocity cell where the quarter cell lives. For example, quarter cells 1, 2, 3, and 4 live inside velocity cell $U_{i,j,k}$ and so have volume

$$(1/4) V_{i,j,k}^U = (1/4) dx_{ue_{i,j}} dy_{un_{i,j}} dz_{u_{i,j,k}}, \quad (25.22)$$

whereas quarter cells 5 and 6 live inside velocity cell $U_{i+1,j,k}$ and so have volume

$$(1/4) V_{i+1,j,k}^U = (1/4) dx_{ue_{i+1,j}} dy_{un_{i+1,j}} dz_{u_{i+1,j,k}}. \quad (25.23)$$

In summary, the volumes of the 12 quarter cells are taken to be

$$\begin{aligned} V(1) &= V(2) = V(3) = V(4) = (1/4) V_{i,j,k}^U \\ V(5) &= V(6) = (1/4) V_{i+1,j,k}^U \\ V(7) &= V(8) = (1/4) V_{i-1,j,k}^U \\ V(9) &= V(10) = (1/4) V_{i,j+1,k}^U \\ V(11) &= V(12) = (1/4) V_{i,j-1,k}^U. \end{aligned} \quad (25.24)$$

25.4.3 Derivative operators

We use the second order finite difference approximations to the derivative of velocity fields

$$\delta_x u_{i,j} = \frac{u_{i+1,j} - u_{i,j}}{dx_{ue_{i,j}}} \quad (25.25)$$

$$\delta_y u_{i,j} = \frac{u_{i,j+1} - u_{i,j}}{dy_{un_{i,j}}}. \quad (25.26)$$

The distances $dx_{ue_{i,j}}$ and $dy_{un_{i,j}}$ represent the zonal and meridional distances between the velocity points (Figure 9.6). All metric stretching factors are absorbed into these grid distances according to the discussion in Section 21.12.4 of Griffies (2004).

25.4.4 Tension and strain for the subcells

A decision must be made regarding the form of tension and strain to discretize. The goal here is to work with a discretization involving the least amount of grid factors in order to minimize computational expense. The first forms of tension and strain are more symmetric,

$$e_T = dy(u/dy)_{,x} - dx(v/dx)_{,y} \quad (25.27)$$

$$e_S = dx(u/dx)_{,y} + dy(v/dy)_{,x} \quad (25.28)$$

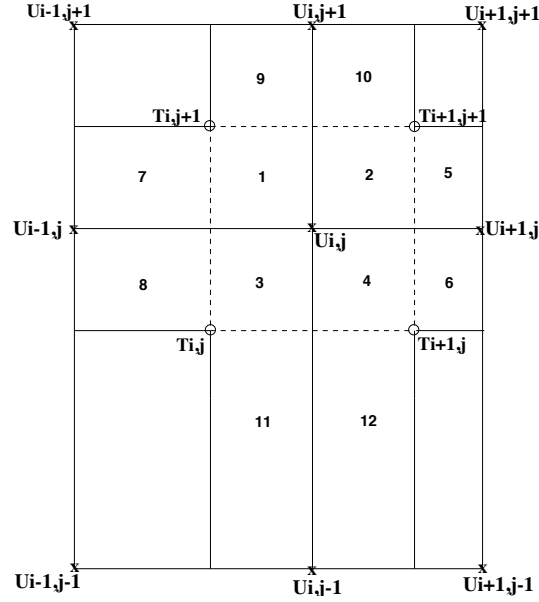


Figure 25.1: Stencil for the discrete horizontal friction functional. The 12 quarter cells each contain contributions to the functional from the central velocity point $U_{i,j}$. The tracer points $T_{i,j}$ are oriented according to the B-grid (see Figure 9.1). This figure is taken after Figure 19.2 of Griffies (2004).

and lead to a compact form for the discretized friction operator (see below), directly analogous to the continuum form given by equations (25.6) and (25.7). The second forms isolate the Cartesian expression

$$e_T = u_{,x} - v_{,y} + v \partial_y \ln dx - u \partial_x \ln dy \quad (25.29)$$

$$e_S = u_{,y} + v_{,x} - u \partial_y \ln dx - v \partial_x \ln dy. \quad (25.30)$$

The second forms require less computation, since the metric terms

$$(M_T)_{i,j} = -u_{i,j} (\partial_x \ln dy)_{i,j} + v_{i,j} (\partial_y \ln dx)_{i,j} \quad (25.31)$$

$$(M_S)_{i,j} = -u_{i,j} (\partial_y \ln dx)_{i,j} - v_{i,j} (\partial_x \ln dy)_{i,j} \quad (25.32)$$

are common to each of the four surrounding triads, and the grid factors

$$dh1dy_{i,j} = (\partial_y \ln dx)_{i,j} \quad (25.33)$$

$$dh2dx_{i,j} = (\partial_x \ln dy)_{i,j} \quad (25.34)$$

can be computed at the start of the integration. Hence, we choose the first form of the deformation rates to develop the discrete friction, and the second form to evaluate the deformation rates within the discrete friction. For purposes of completeness, both discrete forms of the deformation rates are displayed in the following.

The first form for tension in the 12 subcells is

$$e_T(1) = dyue_{i-1,j} \delta_x(u_{i-1,j}/dyu_{i-1,j}) - dxun_{i,j} \delta_y(v_{i,j}/dxu_{i,j}) \quad (25.35)$$

$$e_T(2) = dyue_{i,j} \delta_x(u_{i,j}/dyu_{i,j}) - dxun_{i,j} \delta_y(v_{i,j}/dxu_{i,j}) \quad (25.36)$$

$$e_T(3) = dyue_{i-1,j} \delta_x(u_{i-1,j}/dyu_{i-1,j}) - dxun_{i,j-1} \delta_y(v_{i,j-1}/dxu_{i,j-1}) \quad (25.37)$$

$$e_T(4) = dyue_{i,j} \delta_x(u_{i,j}/dyu_{i,j}) - dxun_{i,j-1} \delta_y(v_{i,j-1}/dxu_{i,j-1}) \quad (25.38)$$

$$e_T(5) = dyue_{i,j} \delta_x(u_{i,j}/dyu_{i,j}) - dxun_{i+1,j} \delta_y(v_{i+1,j}/dxu_{i+1,j}) \quad (25.39)$$

$$e_T(6) = dyue_{i,j} \delta_x(u_{i,j}/dyu_{i,j}) - dxun_{i+1,j-1} \delta_y(v_{i+1,j-1}/dxu_{i+1,j-1}) \quad (25.40)$$

$$e_T(7) = dyue_{i-1,j} \delta_x(u_{i-1,j}/dyu_{i-1,j}) - dxun_{i-1,j} \delta_y(v_{i-1,j}/dxu_{i-1,j}) \quad (25.41)$$

$$e_T(8) = dyue_{i-1,j} \delta_x(u_{i-1,j}/dyu_{i-1,j}) - dxun_{i-1,j-1} \delta_y(v_{i-1,j-1}/dxu_{i-1,j-1}) \quad (25.42)$$

$$e_T(9) = dyue_{i-1,j+1} \delta_x(u_{i-1,j+1}/dyu_{i-1,j+1}) - dxun_{i,j} \delta_y(v_{i,j}/dxu_{i,j}) \quad (25.43)$$

$$e_T(10) = dyue_{i,j+1} \delta_x(u_{i,j+1}/dyu_{i,j+1}) - dxun_{i,j} \delta_y(v_{i,j}/dxu_{i,j}) \quad (25.44)$$

$$e_T(11) = dyue_{i-1,j-1} \delta_x(u_{i-1,j-1}/dyu_{i-1,j-1}) - dxun_{i,j-1} \delta_y(v_{i,j-1}/dxu_{i,j-1}) \quad (25.45)$$

$$e_T(12) = dyue_{i,j-1} \delta_x(u_{i,j-1}/dyu_{i,j-1}) - dxun_{i,j-1} \delta_y(v_{i,j-1}/dxu_{i,j-1}) \quad (25.46)$$

and the first form for strain is

$$e_S(1) = dxun_{i,j} \delta_y(u_{i,j}/dxu_{i,j}) + dyue_{i-1,j} \delta_x(v_{i-1,j}/dyu_{i-1,j}) \quad (25.47)$$

$$e_S(2) = dxun_{i,j} \delta_y(u_{i,j}/dxu_{i,j}) + dyue_{i,j} \delta_x(v_{i,j}/dyu_{i,j}) \quad (25.48)$$

$$e_S(3) = dxun_{i,j-1} \delta_y(u_{i,j-1}/dxu_{i,j-1}) + dyue_{i-1,j} \delta_x(v_{i-1,j}/dyu_{i-1,j}) \quad (25.49)$$

$$e_S(4) = dxun_{i,j-1} \delta_y(u_{i,j-1}/dxu_{i,j-1}) + dyue_{i,j} \delta_x(v_{i,j}/dyu_{i,j}) \quad (25.50)$$

$$e_S(5) = dxun_{i+1,j} \delta_y(u_{i+1,j}/dxu_{i+1,j}) + dyue_{i,j} \delta_x(v_{i,j}/dyu_{i,j}) \quad (25.51)$$

$$e_S(6) = dxun_{i+1,j-1} \delta_y(u_{i+1,j-1}/dxu_{i+1,j-1}) + dyue_{i,j} \delta_x(v_{i,j}/dyu_{i,j}) \quad (25.52)$$

$$e_S(7) = dxun_{i-1,j} \delta_y(u_{i-1,j}/dxu_{i-1,j}) + dyue_{i-1,j} \delta_x(v_{i-1,j}/dyu_{i-1,j}) \quad (25.53)$$

$$e_S(8) = dxun_{i-1,j-1} \delta_y(u_{i-1,j-1}/dxu_{i-1,j-1}) + dyue_{i-1,j} \delta_x(v_{i-1,j}/dyu_{i-1,j}) \quad (25.54)$$

$$e_S(9) = dxun_{i,j} \delta_y(u_{i,j}/dxu_{i,j}) + dyue_{i-1,j+1} \delta_x(v_{i-1,j+1}/dyu_{i-1,j+1}) \quad (25.55)$$

$$e_S(10) = dxun_{i,j} \delta_y(u_{i,j}/dxu_{i,j}) + dyue_{i,j+1} \delta_x(v_{i,j+1}/dyu_{i,j+1}) \quad (25.56)$$

$$e_S(11) = dxun_{i,j-1} \delta_y(u_{i,j-1}/dxu_{i,j-1}) + dyue_{i-1,j-1} \delta_x(v_{i-1,j-1}/dyu_{i-1,j-1}) \quad (25.57)$$

$$e_S(12) = dxun_{i,j-1} \delta_y(u_{i,j-1}/dxu_{i,j-1}) + dyue_{i,j-1} \delta_x(v_{i,j-1}/dyu_{i,j-1}). \quad (25.58)$$

The second forms for tension and strain are

$$e_T(1) = \delta_x u_{i-1,j} - \delta_y v_{i,j} + (M_T)_{i,j} \quad e_S(1) = \delta_y u_{i,j} + \delta_x v_{i-1,j} + (M_S)_{i,j} \quad (25.59)$$

$$e_T(2) = \delta_x u_{i,j} - \delta_y v_{i,j} + (M_T)_{i,j} \quad e_S(2) = \delta_y u_{i,j} + \delta_x v_{i,j} + (M_S)_{i,j} \quad (25.60)$$

$$e_T(3) = \delta_x u_{i-1,j} - \delta_y v_{i,j-1} + (M_T)_{i,j} \quad e_S(3) = \delta_y u_{i,j-1} + \delta_x v_{i-1,j} + (M_S)_{i,j} \quad (25.61)$$

$$e_T(4) = \delta_x u_{i,j} - \delta_y v_{i,j-1} + (M_T)_{i,j} \quad e_S(4) = \delta_y u_{i,j-1} + \delta_x v_{i,j} + (M_S)_{i,j} \quad (25.62)$$

$$e_T(5) = \delta_x u_{i,j} - \delta_y v_{i+1,j} + (M_T)_{i+1,j} \quad e_S(5) = \delta_y u_{i+1,j} + \delta_x v_{i,j} + (M_S)_{i+1,j} \quad (25.63)$$

$$e_T(6) = \delta_x u_{i,j} - \delta_y v_{i+1,j-1} + (M_T)_{i+1,j} \quad e_S(6) = \delta_y u_{i+1,j-1} + \delta_x v_{i,j} + (M_S)_{i+1,j} \quad (25.64)$$

$$e_T(7) = \delta_x u_{i-1,j} - \delta_y v_{i-1,j} + (M_T)_{i-1,j} \quad e_S(7) = \delta_y u_{i-1,j} + \delta_x v_{i-1,j} + (M_S)_{i-1,j} \quad (25.65)$$

$$e_T(8) = \delta_x u_{i-1,j} - \delta_y v_{i-1,j-1} + (M_T)_{i-1,j} \quad e_S(8) = \delta_y u_{i-1,j-1} + \delta_x v_{i-1,j} + (M_S)_{i-1,j} \quad (25.66)$$

$$e_T(9) = \delta_x u_{i-1,j+1} - \delta_y v_{i,j} + (M_T)_{i,j+1} \quad e_S(9) = \delta_y u_{i,j} + \delta_x v_{i-1,j+1} + (M_S)_{i,j+1} \quad (25.67)$$

$$e_T(10) = \delta_x u_{i,j+1} - \delta_y v_{i,j} + (M_T)_{i,j+1} \quad e_S(10) = \delta_y u_{i,j} + \delta_x v_{i,j+1} + (M_S)_{i,j+1} \quad (25.68)$$

$$e_T(11) = \delta_x u_{i-1,j-1} - \delta_y v_{i,j-1} + (M_T)_{i,j-1} \quad e_S(11) = \delta_y u_{i,j-1} + \delta_x v_{i-1,j-1} + (M_S)_{i,j-1} \quad (25.69)$$

$$e_T(12) = \delta_x u_{i,j-1} - \delta_y v_{i,j-1} + (M_T)_{i,j-1} \quad e_S(12) = \delta_y u_{i,j-1} + \delta_x v_{i,j-1} + (M_S)_{i,j-1}. \quad (25.70)$$

25.4.5 Functional derivative of e_T and e_S

Choosing to work with the discretized first form of tension (equation (25.27)) leads to the functional derivative of the tension within the central four subcells

$$\frac{\partial e_T(1)}{\partial(u^{(b)})_{i,j}} = \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^1}{dyu_{i,j}} + \frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^2}{dxu_{i,j}} \quad (25.71)$$

$$\frac{\partial e_T(2)}{\partial(u^{(b)})_{i,j}} = -\frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^1}{dyu_{i,j}} + \frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^2}{dxu_{i,j}} \quad (25.72)$$

$$\frac{\partial e_T(3)}{\partial(u^{(b)})_{i,j}} = \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^1}{dyu_{i,j}} - \frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^2}{dxu_{i,j}} \quad (25.73)$$

$$\frac{\partial e_T(4)}{\partial(u^{(b)})_{i,j}} = -\frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^1}{dyu_{i,j}} - \frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^2}{dxu_{i,j}} \quad (25.74)$$

and to the strain within the same cells

$$\frac{\partial e_S(1)}{\partial(u^{(b)})_{i,j}} = -\frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^1}{dxu_{i,j}} + \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^2}{dyu_{i,j}} \quad (25.75)$$

$$\frac{\partial e_S(2)}{\partial(u^{(b)})_{i,j}} = -\frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^1}{dxu_{i,j}} - \frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^2}{dyu_{i,j}} \quad (25.76)$$

$$\frac{\partial e_S(3)}{\partial(u^{(b)})_{i,j}} = \frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^1}{dxu_{i,j}} + \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^2}{dyu_{i,j}} \quad (25.77)$$

$$\frac{\partial e_S(4)}{\partial(u^{(b)})_{i,j}} = \frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^1}{dxu_{i,j}} - \frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^2}{dyu_{i,j}}. \quad (25.78)$$

The functional derivatives for tension and strain in the other eight cells are given by

$$\frac{\partial e_T(5)}{\partial(u^{(b)})_{i,j}} = -\frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^1}{dyu_{i,j}} \quad \frac{\partial e_S(5)}{\partial(u^{(b)})_{i,j}} = -\frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^2}{dyu_{i,j}} \quad (25.79)$$

$$\frac{\partial e_T(6)}{\partial(u^{(b)})_{i,j}} = -\frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^1}{dyu_{i,j}} \quad \frac{\partial e_S(6)}{\partial(u^{(b)})_{i,j}} = -\frac{dyue_{i,j}}{dxue_{i,j}} \frac{\delta_b^2}{dyu_{i,j}} \quad (25.80)$$

$$\frac{\partial e_T(7)}{\partial(u^{(b)})_{i,j}} = \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^1}{dyu_{i,j}} \quad \frac{\partial e_S(7)}{\partial(u^{(b)})_{i,j}} = \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^2}{dyu_{i,j}} \quad (25.81)$$

$$\frac{\partial e_T(8)}{\partial(u^{(b)})_{i,j}} = \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^1}{dyu_{i,j}} \quad \frac{\partial e_S(8)}{\partial(u^{(b)})_{i,j}} = \frac{dyue_{i-1,j}}{dxue_{i-1,j}} \frac{\delta_b^2}{dyu_{i,j}} \quad (25.82)$$

$$\frac{\partial e_T(9)}{\partial(u^{(b)})_{i,j}} = \frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^2}{dxu_{i,j}} \quad \frac{\partial e_S(9)}{\partial(u^{(b)})_{i,j}} = -\frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^1}{dxu_{i,j}} \quad (25.83)$$

$$\frac{\partial e_T(10)}{\partial(u^{(b)})_{i,j}} = \frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^2}{dxu_{i,j}} \quad \frac{\partial e_S(10)}{\partial(u^{(b)})_{i,j}} = -\frac{dxun_{i,j}}{dyun_{i,j}} \frac{\delta_b^1}{dxu_{i,j}} \quad (25.84)$$

$$\frac{\partial e_T(11)}{\partial(u^{(b)})_{i,j}} = -\frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^2}{dxu_{i,j}} \quad \frac{\partial e_S(11)}{\partial(u^{(b)})_{i,j}} = \frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^1}{dxu_{i,j}} \quad (25.85)$$

$$\frac{\partial e_T(12)}{\partial(u^{(b)})_{i,j}} = -\frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^2}{dxu_{i,j}}. \quad \frac{\partial e_S(12)}{\partial(u^{(b)})_{i,j}} = \frac{dxun_{i,j-1}}{dyun_{i,j-1}} \frac{\delta_b^1}{dxu_{i,j}}. \quad (25.86)$$

25.4.6 Tidy form for the discretized friction

Focus first on the zonal friction with $b = 1$. Using the volumes enumerated in Section 25.4.2, and dropping the vertical label k since it is the same for all terms, yields for the tension

$$-\sum_{n=1}^{12} V(n) \tau^{xx}(n) \frac{\partial e_T(n)}{\partial (u^{(1)})_{i,j}} = \left(\frac{dyue_{i,j}}{4 dxue_{i,j} dyu_{i,j}} \right) \left([\tau^{xx}(5) + \tau^{xx}(6)] V_{i+1,j}^U + [\tau^{xx}(2) + \tau^{xx}(4)] V_{i,j}^U \right) \\ - \left(\frac{dyue_{i-1,j}}{4 dxue_{i-1,j} dyu_{i-1,j}} \right) \left([\tau^{xx}(1) + \tau^{xx}(3)] V_{i,j}^U + [\tau^{xx}(7) + \tau^{xx}(8)] V_{i-1,j}^U \right). \quad (25.87)$$

Figure 25.1 indicates that a velocity point $U_{i,j}$ is associated with four triads, each of which is used to construct a tension and strain along with a viscosity. This arrangement motivates the following notation (see Figure 25.2):

$$\begin{aligned} (1,1) &= \text{northeast triad} \\ (0,1) &= \text{northwest triad} \\ (0,0) &= \text{southwest triad} \\ (1,0) &= \text{southeast triad.} \end{aligned} \quad (25.88)$$

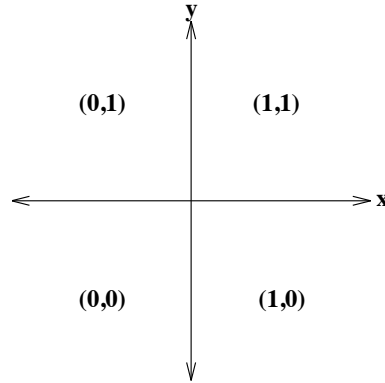


Figure 25.2: Notation for the quadrants surrounding a velocity point.

With this notation,

$$-\sum_{n=1}^{12} V(n) \tau^{xx}(n) \frac{\partial e_T(n)}{\partial (u^{(1)})_{i,j}} = \left(\frac{dyue_{i,j}}{4 dxue_{i,j} dyu_{i,j}} \right) \sum_{ip=0}^1 V_{i+ip,j}^U \sum_{jq=0}^1 (\tau^{xx})_{(i+ip,j)}^{(1-ip,jq)} \\ - \left(\frac{dyue_{i-1,j}}{4 dxue_{i-1,j} dyu_{i,j}} \right) \sum_{ip=0}^1 V_{i-1+ip,j}^U \sum_{jq=0}^1 (\tau^{xx})_{(i-1+ip,j)}^{(1-ip,jq)}. \quad (25.89)$$

The two terms on the right-hand side are centered on the east and west faces, respectively, of the $U_{i,j}$ velocity cell. Consequently, introduce the finite difference derivative operator to yield

$$-\sum_{n=1}^{12} V(n) \tau^{xx}(n) \frac{\partial e_T(n)}{\partial (u^{(1)})_{i,j}} = \left(\frac{dxu_{i,j}}{4 dyu_{i,j}} \right) \delta_x \left[\frac{dyue_{i-1,j}}{dxue_{i-1,j}} \sum_{ip=0}^1 V_{i-1+ip,j}^U \sum_{jq=0}^1 (\tau^{xx})_{(i-1+ip,j)}^{(1-ip,jq)} \right]. \quad (25.90)$$

Dividing by the velocity cell volume $V_{i,j}^U = dxu_{i,j} dyu_{i,j} dzu_{i,j}$ leads to

$$-\left(\frac{1}{V_{i,j}^U} \right) \sum_{n=1}^{12} V(n) \tau^{xx}(n) \frac{\partial e_T(n)}{\partial (u^{(1)})_{i,j}} = \left(\frac{1}{4 dzu_{i,j} (dyu_{i,j})^2} \right) \delta_x \left[\frac{dyue_{i-1,j}}{dxue_{i-1,j}} \sum_{ip=0}^1 V_{i-1+ip,j}^U \sum_{jq=0}^1 (\tau^{xx})_{(i-1+ip,j)}^{(1-ip,jq)} \right]. \quad (25.91)$$

Similar manipulations apply for the strain terms, thus yielding

$$-\left(\frac{1}{V_{i,j}^U}\right) \sum_{n=1}^{12} V(n) \tau^{xy}(n) \frac{\partial e_S(n)}{\partial (u^{(1)})_{i,j}} = \left(\frac{1}{4 \text{dzu}_{i,j} (\text{dxu}_{i,j})^2}\right) \delta_y \left[\frac{\text{dxun}_{i,j-1}}{\text{dyun}_{i,j-1}} \sum_{jq=0}^1 V_{i,j-1+jq}^U \sum_{ip=0}^1 (\tau^{xy})_{(i,j-1+jq)}^{(ip,1-jq)} \right]. \quad (25.92)$$

Bringing the two pieces together leads to the zonal friction acting at the velocity cell $U_{i,j}$

$$\begin{aligned} \rho_o F_{i,j}^{(x)} = & \left(\frac{1}{4 \text{dzu}_{i,j} (\text{dyu}_{i,j})^2}\right) \delta_x \left[\frac{\text{dyue}_{i-1,j}}{\text{dxue}_{i-1,j}} \sum_{ip=0}^1 V_{i-1+ip,j}^U \sum_{jq=0}^1 (\tau^{xx})_{(i-1+ip,j)}^{(1-ip,jq)} \right] \\ & + \left(\frac{1}{4 \text{dzu}_{i,j} (\text{dxu}_{i,j})^2}\right) \delta_y \left[\frac{\text{dxun}_{i,j-1}}{\text{dyun}_{i,j-1}} \sum_{jq=0}^1 V_{i,j-1+jq}^U \sum_{ip=0}^1 (\tau^{xy})_{(i,j-1+jq)}^{(ip,1-jq)} \right]. \end{aligned} \quad (25.93)$$

By inspection, the meridional friction is given by

$$\begin{aligned} \rho_o F_{i,j}^{(y)} = & \left(\frac{1}{4 \text{dzu}_{i,j} (\text{dyu}_{i,j})^2}\right) \delta_x \left[\frac{\text{dyue}_{i-1,j}}{\text{dxue}_{i-1,j}} \sum_{ip=0}^1 V_{i-1+ip,j}^U \sum_{jq=0}^1 (\tau^{xy})_{(i-1+ip,j)}^{(1-ip,jq)} \right] \\ & - \left(\frac{1}{4 \text{dzu}_{i,j} (\text{dxu}_{i,j})^2}\right) \delta_y \left[\frac{\text{dxun}_{i,j-1}}{\text{dyun}_{i,j-1}} \sum_{jq=0}^1 V_{i,j-1+jq}^U \sum_{ip=0}^1 (\tau^{xx})_{(i,j-1+jq)}^{(ip,1-jq)} \right]. \end{aligned} \quad (25.94)$$

Comparison with the continuum friction components given by equations (25.6) and (25.7) indicates that the discretization is consistent, that is, the discrete friction reduces to the continuum friction as the grid size goes to zero.

25.4.7 Tension and strain in the quadrants

There are four tensions and strains corresponding to the four triads surrounding each velocity point. Referring to Figure 25.2, assuming the central point is $U_{i,j}$, discretize the tensions and strains starting from the second form of the continuous tension and strain (equations (25.29) and (25.30)) to find

$$(e_T)_{i,j,(0,1)} = \delta_x u_{i-1,j} - \delta_y v_{i,j} + (M_T)_{i,j} \quad (25.95)$$

$$(e_T)_{i,j,(1,1)} = \delta_x u_{i,j} - \delta_y v_{i,j} + (M_T)_{i,j} \quad (25.96)$$

$$(e_T)_{i,j,(0,0)} = \delta_x u_{i-1,j} - \delta_y v_{i,j-1} + (M_T)_{i,j} \quad (25.97)$$

$$(e_T)_{i,j,(1,0)} = \delta_x u_{i,j} - \delta_y v_{i,j-1} + (M_T)_{i,j} \quad (25.98)$$

$$(e_S)_{i,j,(0,1)} = \delta_y u_{i,j} + \delta_x v_{i-1,j} + (M_S)_{i,j} \quad (25.99)$$

$$(e_S)_{i,j,(1,1)} = \delta_y u_{i,j} + \delta_x v_{i,j} + (M_S)_{i,j} \quad (25.100)$$

$$(e_S)_{i,j,(0,0)} = \delta_y u_{i,j-1} + \delta_x v_{i-1,j} + (M_S)_{i,j} \quad (25.101)$$

$$(e_S)_{i,j,(1,0)} = \delta_y u_{i,j-1} + \delta_x v_{i,j} + (M_S)_{i,j}. \quad (25.102)$$

In general, the four tensions can be written

$$(e_T)_{i,j,(ip,jq)} = \delta_x u_{i+ip-1,j} - \delta_y v_{i,j+jq-1} + (M_T)_{i,j} \quad (25.103)$$

and the four strains can be written

$$(e_S)_{i,j,(ip,jq)} = \delta_y u_{i,j+jq-1} + \delta_x v_{i+ip-1,j} + (M_S)_{i,j} \quad (25.104)$$

where $ip = 0, 1$ and $jq = 0, 1$. Notably, the metric terms

$$(M_T)_{i,j} = -u_{i,j} (\partial_x \ln dy)_{i,j} + v_{i,j} (\partial_y \ln dx)_{i,j} \quad (25.105)$$

$$(M_S)_{i,j} = -u_{i,j} (\partial_y \ln dx)_{i,j} - v_{i,j} (\partial_x \ln dy)_{i,j} \quad (25.106)$$

are common to the four triads, and so only need be computed once per velocity point. Generally, the four tensions and strains are computed in the model and are then used to compute the friction operator. When the Smagorinsky viscosity scheme (Section 25.2 as well as Section 19.3.10 of [Griffies \(2004\)](#)) is enabled, they are used to compute the Smagorinsky viscosity as well.

25.4.8 Comments

Note a few points here related to the proposed discretization.

- The tension and strain for ocean points next to land contain a contribution from a velocity living at the land-sea interface. This velocity, due to the no-slip condition used in B-grid ocean models, has a zero value. In order to provide a full accounting of the generally strong shears next to no-slip walls present in the B-grid ocean models, it is important to include such contributions rather than masking them out.
- In the special case of a uniform Cartesian grid, a constant isotropic viscosity, and a zero anisotropic viscosity, the functionally derived discrete friction operator reduces to the 5-point discrete Laplacian.
- Practical experience has revealed problems with discretization for bottom grid cells when these cells are thin partial cells that are surrounded by thicker partial cells. The problem is that contributions from surrounding thick cells are then normalized by the thin dz of the central cell. To alleviate this problem, it is effective to use the traditional 5-point Laplacian operator for computing friction in the bottom grid cells.

25.4.9 Discretized Smagorinsky viscosity

The nonlinear Smagorinsky viscosity coefficient is determined in terms of the deformation rates e_T and e_S as well as the grid spacing. Since e_T and e_S involve terms with derivatives in both horizontal directions, an averaging must be performed to place them at a common grid position.

[Pacanowski et al. \(1991\)](#) defined both deformation rates at the north face of the U-cell. This is the natural position for the meridional derivative terms. However, to get the zonal derivative terms defined at the north face, it is necessary to average over the four zonal derivatives surrounding the north face. The problem with such “4 point” averages on the B-grid is that they can introduce computational modes. Computational modes are not always problematical if there are other processes that can suppress the growth of the modes. The problem with the computational modes in the Smagorinsky scheme is that they allow nontrivial field configurations yielding a zero deformation rate. Hence, they produce a zero Smagorinsky viscosity and so are not dissipated. Furthermore, these modes represent grid scale waves, which are the waves an ideal implementation of the Smagorinsky scheme should dissipate the most. Therefore, it is not acceptable to allow these modes in the discretized Smagorinsky scheme. Another approach is necessary.

The functional discretization described in this chapter eliminates the computational modes. For each velocity triad, there is a corresponding Smagorinsky viscosity. In particular, referring to the deformation rates defined in equation (25.102) yields the corresponding Smagorinsky diffusivities

$$A_{i,j,(ip,jq)} = (\Upsilon \Delta s / \pi)^2 |E|_{i,j,(ip,jq)} \quad (25.107)$$

where $ip = 0, 1$, $jq = 0, 1$ are the triad labels, and

$$[E_{i,j,(ip,jq)}]^2 = [(e_T)_{i,j,(ip,jq)}]^2 + [(e_S)_{i,j,(ip,jq)}]^2 \quad (25.108)$$

is the discrete total deformation rate. As mentioned earlier, one advantage of the functional approach over the “Laplacian plus metric” approach (see Section 19.4 of [Griffies \(2004\)](#)) is the exploitation of the deformation rates for computing both the Smagorinsky viscosity and the friction operator.

25.5 Lateral friction operator for C-grid MOM

The purpose of this section is to formulate a discrete lateral friction operator for the C-grid. We follow the B-grid approach by establishing a formalism based on taking the functional derivative of the dissipation functional. Given the different layout of velocity, the C-grid has a somewhat simpler stencil for the friction operator than the B-grid.

25.5.1 Discrete functional derivative and the grid stencil

The general approach used to derive a discrete friction operator is to first discretize the dissipation function, \mathcal{S} , and then to perform a discrete version of a functional derivative. The discrete functional derivative is merely a partial derivative of the discrete functional with respect to the discrete velocity components $u_{i,j}$ and $v_{i,j}$.

We start from the expression for the discrete friction operator (equation (25.17) as well as equation (19.4) of Griffies (2004))

$$-\rho F_{i,j}^x = \frac{1}{2V_{i,j}^{(u)}} \frac{\partial \mathcal{S}_{i,j}}{\partial u_{i,j}} = \frac{1}{V_{i,j}^{(u)}} \sum_n V(n) \left(\frac{\partial e_T(n)}{\partial u_{i,j}} \tau^{xx}(n) + \frac{\partial e_S(n)}{\partial u_{i,j}} \tau^{xy}(n) \right) \quad (25.109)$$

$$-\rho F_{i,j}^y = \frac{1}{2V_{i,j}^{(v)}} \frac{\partial \mathcal{S}_{i,j}}{\partial v_{i,j}} = \frac{1}{V_{i,j}^{(v)}} \sum_n V(n) \left(\frac{\partial e_T(n)}{\partial v_{i,j}} \tau^{xx}(n) + \frac{\partial e_S(n)}{\partial v_{i,j}} \tau^{yx}(n) \right). \quad (25.110)$$

The sum in these expressions extends over the stencil whereby the central velocity components $u_{i,j}$ and $v_{i,j}$ contribute to a non-zero functional derivative. For the B-grid, the stencil extends over 12 sub-regions, each with their own viscosity, volume $V(n)$, deformation rates, and stresses (Figure 25.1). The stencil is simpler for the C-grid.

To determine the stencil for the C-grid friction operator, consider the C-grid layout in Figure 25.3 and focus on the zonal velocity component $u_{i,j}$. This component participates in the calculation of the deformation rates e_T and e_S through both its zonal and meridional derivatives. Figure 25.4 exposes the points contributing to e_T and e_S where $u_{i,j}$ contributes. Note the 5-point stencil for the participating zonal velocity components, and the 4-point stencil for the participating meridional velocity components. The complement situation holds for the points where $v_{i,j}$ participates (second panel of Figure 25.4). It is notable that this stencil is smaller than the corresponding B-grid stencil shown in Figure 25.1.

25.5.2 Deformation rates

We now present details for deriving the discrete zonal friction operator on a C-grid. For this purpose, focus on the left panel of Figure 25.4, in which we identify the deformation rates that include a contribution from $u_{i,j}$

$$e_T^{i,j} = \left(\frac{dyt_{i,j}}{dxt_{i,j}} \right) \left(\frac{u_{i,j}}{dyt_{i,j}} - \frac{u_{i-1,j}}{dyt_{i-1,j}} \right) - \left(\frac{dxt_{i,j}}{dyt_{i,j}} \right) \left(\frac{v_{i,j}}{dxt_{i,j}} - \frac{v_{i,j-1}}{dxt_{i,j-1}} \right) \quad (25.111)$$

$$e_T^{i+1,j} = \left(\frac{dyt_{i+1,j}}{dxt_{i+1,j}} \right) \left(\frac{u_{i+1,j}}{dyt_{i+1,j}} - \frac{u_{i,j}}{dyt_{i,j}} \right) - \left(\frac{dxt_{i+1,j}}{dyt_{i+1,j}} \right) \left(\frac{v_{i+1,j}}{dxt_{i+1,j}} - \frac{v_{i+1,j-1}}{dxt_{i+1,j-1}} \right) \quad (25.112)$$

$$e_S^{i,j} = \left(\frac{dxu_{i,j}}{dyu_{i,j}} \right) \left(\frac{u_{i,j+1}}{dxte_{i,j+1}} - \frac{u_{i,j}}{dxte_{i,j}} \right) + \left(\frac{dyu_{i,j}}{dxu_{i,j}} \right) \left(\frac{v_{i+1,j}}{dytn_{i+1,j}} - \frac{v_{i,j}}{dytn_{i,j}} \right) \quad (25.113)$$

$$e_S^{i,j-1} = \left(\frac{dxu_{i,j-1}}{dyu_{i,j-1}} \right) \left(\frac{u_{i,j}}{dxte_{i,j}} - \frac{u_{i,j-1}}{dxte_{i,j-1}} \right) + \left(\frac{dyu_{i,j-1}}{dxu_{i,j-1}} \right) \left(\frac{v_{i+1,j-1}}{dytn_{i+1,j-1}} - \frac{v_{i,j-1}}{dytn_{i,j-1}} \right). \quad (25.114)$$

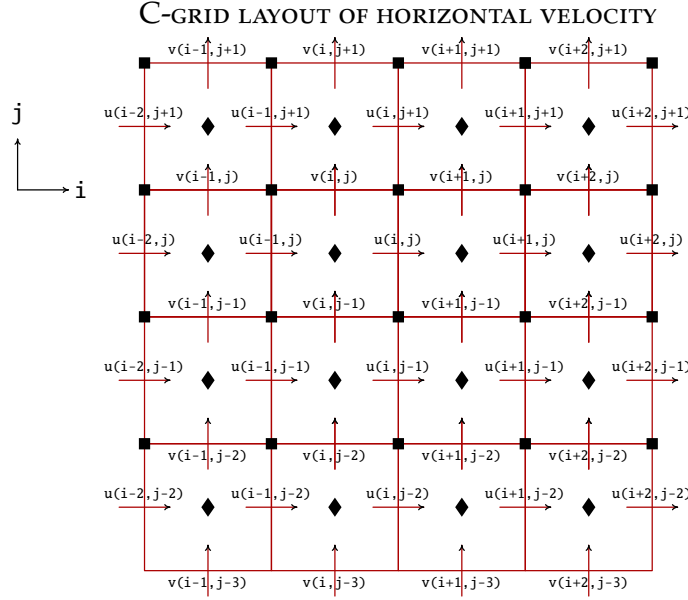


Figure 25.3: Array of C-grid velocity vectors, used to help deduce the friction operator stencil. The solid diamonds are the tracer points, which are also points where the deformation rate e_T is located on the C-grid. The solid squares are the vorticity points, which are also points where the deformation rate e_S is located on the C-grid. Figure 25.4 provides more details for the discrete version of e_T and e_S .

The corresponding discrete functional derivatives are given by

$$\frac{\partial e_T^{i,j}}{\partial u_{i,j}} = \left(\frac{dyt_{i,j}}{dxt_{i,j} dyte_{i,j}} \right) \quad (25.115)$$

$$\frac{\partial e_T^{i+1,j}}{\partial u_{i,j}} = - \left(\frac{dyt_{i+1,j}}{dxt_{i+1,j} dyte_{i,j}} \right) \quad (25.116)$$

$$\frac{\partial e_S^{i,j}}{\partial u_{i,j}} = - \left(\frac{dxu_{i,j}}{dyu_{i,j} dxt_{i,j}} \right) \quad (25.117)$$

$$\frac{\partial e_S^{i,j-1}}{\partial u_{i,j}} = \left(\frac{dxu_{i,j-1}}{dyu_{i,j-1} dxt_{i,j}} \right). \quad (25.118)$$

25.5.3 Stress tensor components

For discretization of the zonal friction $\rho F_{i,j}^x$, we need discrete versions of the stress tensor components (equations (25.8) and (25.9))

$$\tau^{xx} = (\rho/2) [2A e_T + D(e_S \cos 2\theta - e_T \sin 2\theta) \sin 2\theta] \quad (25.119)$$

$$\tau^{xy} = (\rho/2) [2A e_S - D(e_S \cos 2\theta - e_T \sin 2\theta) \cos 2\theta]. \quad (25.120)$$

The case of zero anisotropy in the viscosity tensor ($D = 0$) is simplest, in which case τ^{xx} is centered on the grid according to the discretization of e_T (solid diamonds on Figure 25.4), and τ^{xy} is centered according to the discretization of e_S (solid square on Figure 25.4). When anisotropy is present, both e_T and e_S contribute to each of the stress tensor components. This situation is awkward on the discrete C-grid since e_T and e_S are not co-located. We could choose a spatial averaging to define co-located deformation rates, but such averaging comes at the cost of a wider grid stencil. So instead of averaging, we choose to keep the same

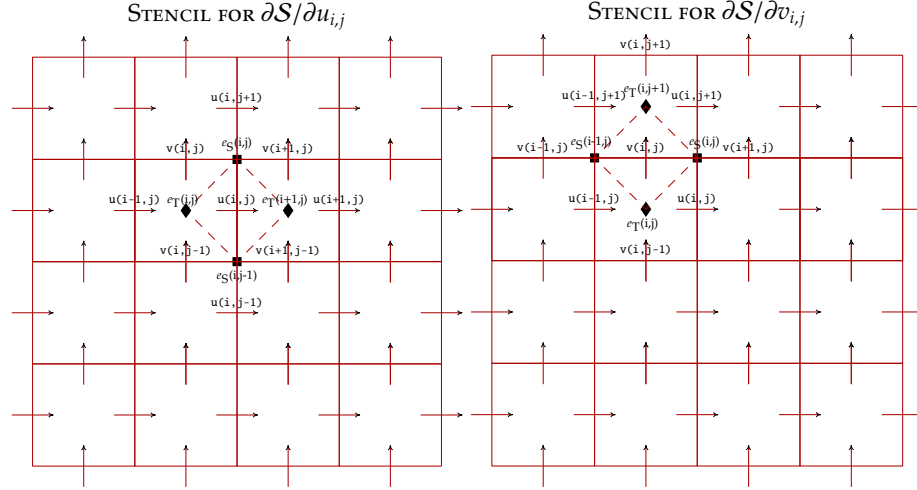


Figure 25.4: Left panel: stencil for the $u_{i,j}$ contribution to the lateral C-grid friction. Right panel: stencil for the $v_{i,j}$ contribution to the lateral C-grid friction. The solid diamonds denote the central point for the discrete version of $e_T = dy(u/dy)_{,x} - dx(v/dx)_{,y}$ (equation (25.27)), and the solid squares are for $e_S = dx(u/dx)_{,y} + dy(v/dy)_{,x}$ (equation 25.28)). We identify a dashed diamond region that connects the deformation rates used to compute the friction acting on the respective velocity components. In total, there are $5u + 4v$ velocity points impacting the zonal friction, and $5v + 4u$ velocity points impacting the meridional friction. Note the presence of a central 5-point stencil, which is typical of a discrete Laplacian operator. This stencil should be compared to the larger stencil required for the B-grid (see Figure 25.1).

stencil and to associate $e_T^{i,j}$ and $e_S^{i,j}$ as a pair, and $e_S^{i,j-1}$ and $e_T^{i+1,j}$ as another pair. In this way, the discrete $\rho F_{i,j}^x$ will make use of the stress tensor elements

$$\tau_{i,j}^{xx} = \frac{\rho_{i,j}}{2} \left[2A e_T^{i,j} + D \sin 2\theta \left(e_S^{i,j} \cos 2\theta - e_T^{i,j} \sin 2\theta \right) \right] \quad (25.121)$$

$$\tau_{i,j-1}^{xy} = \frac{\rho_{i+1,j}}{2} \left[2A e_S^{i,j-1} - D \cos 2\theta \left(e_S^{i,j-1} \cos 2\theta - e_T^{i+1,j} \sin 2\theta \right) \right]. \quad (25.122)$$

Likewise, the meridional friction $\rho F_{i,j}^y$ uses

$$\tau_{i,j}^{yy} = -\frac{\rho_{i,j}}{2} \left[2A e_T^{i,j} + D \sin 2\theta \left(e_S^{i,j} \cos 2\theta - e_T^{i,j} \sin 2\theta \right) \right] \quad (25.123)$$

$$\tau_{i-1,j}^{yx} = \frac{\rho_{i,j+1}}{2} \left[2A e_S^{i-1,j} - D \cos 2\theta \left(e_S^{i-1,j} \cos 2\theta - e_T^{i,j+1} \sin 2\theta \right) \right]. \quad (25.124)$$

Note that this prescription maintains the continuum symmetry of the stress tensor components

$$\tau_{i,j}^{xx} = -\tau_{i,j}^{yy} \quad (25.125)$$

$$\tau_{i,j}^{xy} = \tau_{i,j}^{yx}. \quad (25.126)$$

We detail choices for the orientation angle θ in Section 25.5.4, and the viscosities in Section 25.5.5, with these prescriptions maintaining the above symmetry. Yet note that the use of partial slip side boundaries generally breaks the symmetry $\tau_{i,j}^{xy} = \tau_{i,j}^{yx}$ (Section 25.6).

25.5.4 Orientation angle

As discussed by [Smith and McWilliams \(2003\)](#), the orientation angle θ is often prescribed according to the direction of the horizontal velocity field (equation (17.81) of [Griffies \(2004\)](#))

$$\hat{\mathbf{s}} = \frac{\mathbf{u}}{|\mathbf{u}|}. \quad (25.127)$$

It is awkward to compute this directional vector on the C-grid, given that the velocity components are not co-located. Rather than perform a spatial average, we ignore the offset and so compute

$$s_{i,j}^x = \frac{u_{i,j}}{|\mathbf{u}|} \quad (25.128)$$

$$s_{i,j}^y = \frac{v_{i,j}}{|\mathbf{u}|} \quad (25.129)$$

$$|\mathbf{u}|^2 = u_{i,j}^2 + v_{i,j}^2. \quad (25.130)$$

25.5.5 Viscosities

The isotropic viscosity, A , and anisotropic viscosity, D , are needed to compute the stress tensor. When choosing a Smagorinsky approach (Section 25.4.9), we need to determine which deformation rates to use. Following from that used for discretizing the stress tensor components in Section 25.5.3, we make the following choices

- For the diagonal components $\tau_{i,j}^{xx} = -\tau_{i,j}^{yy}$, we choose A, D to be proportional to the deformation rates $e_T^{i,j}$ and $e_S^{i,j}$.
- For the off-diagonal components $\tau_{i,j}^{xy} = \tau_{i,j}^{yx}$, we choose A, D to be proportional to the deformation rates $e_T^{i+1,j+1}$ and $e_S^{i,j}$.

25.5.6 Volumes

We must now choose the volume factors appearing in expression (25.109) for the zonal friction. As for the B-grid calculation in Section 25.4.2 (see also Section 19.3.3 of Griffies (2004)), we simplify the volume expressions to simplify the resulting friction operator. For this purpose, we take all volume factors to be equal to the volume of a cell in which $u_{i,j}$ is at the center

$$V_{i,j}^{(u)} = V(1) = V(2) = dx te_{i,j} dy te_{i,j} dz te_{i,j}. \quad (25.131)$$

25.5.7 Discrete C-grid friction operators

Using this expression for the volume factors in equation (25.109), along with the functional derivatives (25.111)-(25.114), leads to

$$-\rho F_{i,j}^x = \left(\frac{dyt_{i,j}}{dx te_{i,j} dy te_{i,j}} \right) \tau_{i,j}^{xx} - \left(\frac{dyt_{i+1,j}}{dx te_{i+1,j} dy te_{i,j}} \right) \tau_{i+1,j}^{xx} - \left(\frac{dxu_{i,j}}{dyu_{i,j} dx te_{i,j}} \right) \tau_{i,j}^{xy} + \left(\frac{dxu_{i,j-1}}{dyu_{i,j-1} dx te_{i,j}} \right) \tau_{i,j-1}^{xy} \quad (25.132)$$

Rearrangement renders for the zonal friction operator

$$\rho F_{i,j}^x = \left(\frac{dx te_{i,j}}{dy te_{i,j}} \right) \left(\frac{dyt_{i+1,j}}{dx te_{i+1,j}} \tau_{i+1,j}^{xx} - \frac{dyt_{i,j}}{dx te_{i,j}} \tau_{i,j}^{xx} \right) + \left(\frac{dy te_{i,j}}{dx te_{i,j}} \right) \left(\frac{dxu_{i,j}}{dyu_{i,j}} \tau_{i,j}^{xy} - \frac{dxu_{i,j-1}}{dyu_{i,j-1}} \tau_{i,j-1}^{xy} \right) \quad (25.133)$$

$$= \left(\frac{dx te_{i,j}}{dy te_{i,j}} \right) \delta_x \left(\frac{dyt_{i,j}}{dx te_{i,j}} \tau_{i,j}^{xx} \right) + \left(\frac{dy te_{i,j}}{dx te_{i,j}} \right) \delta_y \left(\frac{dxu_{i,j-1}}{dyu_{i,j-1}} \tau_{i,j-1}^{xy} \right). \quad (25.134)$$

Similar manipulations yield for the meridional friction operator

$$\rho F_{i,j}^y = \left(\frac{dyt_{i,j}}{dx te_{i,j}} \right) \delta_y \left(\frac{dx te_{i,j}}{dyt_{i,j}} \tau_{i,j}^{yy} \right) + \left(\frac{dx te_{i,j}}{dyt_{i,j}} \right) \delta_x \left(\frac{dyu_{i-1,j}}{dxu_{i-1,j}} \tau_{i-1,j}^{yx} \right), \quad (25.135)$$

where again $\tau^{yy} = -\tau^{xx}$. Also, $\tau^{xy} = \tau^{yx}$, with the exception of partial-slip side boundaries, as discussed in Section 25.6.

25.6 Boundary conditions

The natural treatment of friction next to side boundaries is complementary between the B-grid and C-grid. We summarize these issues in this section.

25.6.1 B-grid

Land cells in MOM are defined through the tracer cells. A velocity point that sits at the corner of a land tracer cell, as per a B-grid, is given an identically zero value. It is for this reason that the B-grid MOM is said to use a no-slip side boundary condition.

There are very special cases where one may specify a free-slip boundary, but these cases make unrealistic assumptions about the land-sea masking, such as for a zonally periodic channel. MOM does not make these assumptions, which means that all B-grid simulations with MOM utilize a no-slip side boundary condition.

25.6.2 C-grid

Since land cells in MOM are defined through the tracer cells, it is possible for there to be a nonzero C-grid velocity crossing some of the faces of a grid cell adjacent to land. For example, consider the configuration in Figure 25.5. Rather than velocity sitting at the tracer cell corner, it is the deformation rate e_s . The following choices are available in MOM.

- **FREE-SLIP:** A simple choice for how to specify e_s for corner points adjacent to land is to set $e_s = 0$. This specification amounts to a *free-slip* condition. This is the default setting in MOM with the C-grid, with this specification provided by masking with the B-grid velocity mask. Partial slip conditions can also be specified, though further work is required with the code.
- **FREE-SLIP WITH ANISOTROPIC VISCOSITY:** When using the anisotropic viscosity scheme (Section 25.3), the stress tensor τ^{xy} generally has contributions from both e_s and e_T (equation (25.9)). To remain consistent with the free-slip side wall condition, we set $\tau^{xy} = 0$ on the land corner points, even when using anisotropic viscosity.
- **BIHARMONIC OPERATOR:** For the biharmonic operator, we set both e_s to zero on land corner points, as well as the strain associated with the Laplacian friction operator (equation (25.14)).
- **SIDE DRAG:** The side-drag scheme from Deremble et al. (2012) is implemented on the C-grid by setting

$$\tau^{xy} = -\rho C_d |u| u \quad \text{for } F^x \quad (25.136)$$

$$\tau^{yx} = -\rho C_d |v| v \quad \text{for } F^y, \quad (25.137)$$

where the u and v used in the respective stresses are taken from the value just in from the land boundary.

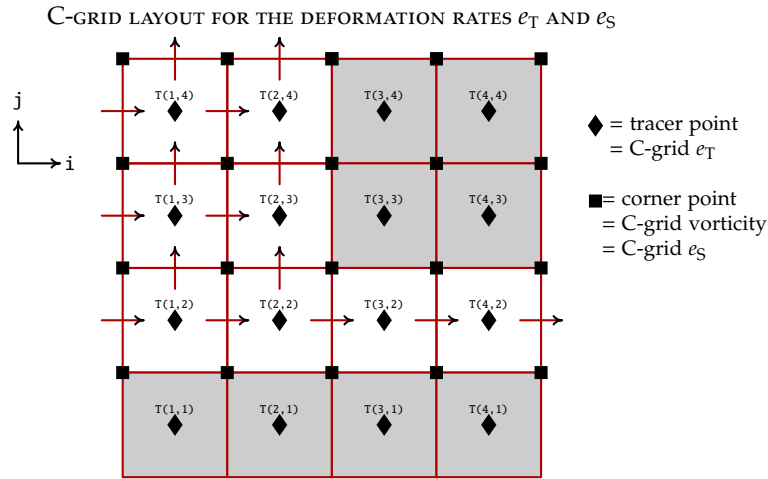


Figure 25.5: Land-sea example for the C-grid, with land cells shaded. The deformation rate e_T is centred at a tracer point (solid diamond), and the deformation rate e_S is at the solid squares, on the northeast corner of the tracer cell. The specification of e_S , or more generally τ^{xy} , determine how friction next to side boundaries impact the interior flow. The default for the C-grid MOM is a free-slip side wall condition, whereby $e_S = 0$ when it is located on a land cell corner. Operationally, this boundary condition is directly analogous to setting the B-grid velocity to zero at the same points. Physically, the impact on the interior flow is the opposite, with the C-grid specification of $e_S = 0$ leads to a free-slip whereas the B-grid specification $\mathbf{u} = 0$ leads to a no-slip.

EDDY-TOPOGRAPHY INTERACTION VIA NEPTUNE

Contents

26.1	Introduction	373
26.2	Basics of the parameterization in MOM	374
26.2.1	The Eby and Holloway (1994) approach	374
26.2.2	Maltrud and Holloway (2008) approach	374
26.2.3	Specifying the length scale	375
26.3	Topostrophy diagnostic	375

The purpose of this chapter is to present a method for parameterizing the interactions between unresolved mesoscale eddies and topography. The following MOM modules are directly connected to the material in this chapter:

```
ocean_param/lateral/ocean_lapgen_friction.F90
ocean_param/lateral/ocean_bihgen_friction.F90
ocean_param/lateral/ocean_lapcgrid_friction.F90
ocean_param/lateral/ocean_bihcgrid_friction.F90
```

26.1 Introduction

Based on statistical mechanics arguments, [Holloway \(1992\)](#) proposed that interactions between mesoscale eddies and topography result in a stress on the ocean with two important consequences: (1) the ocean is not driven toward a state of rest, (2) the resulting motion may have scales much larger than the eddy scales. That is, eddy-topography interactions can generate coherent mean flows on the scale of the topography. When Holloway described coastal currents that persistently flow against both the wind forcing and pressure gradient, the response was that it must be due to Neptune. Who else? Hence, the effect is referred to colloquially as the *Neptune effect*. The magnitude of the associated topographic stress is dependent on the correlation between pressure p and topographic gradients ∇H , and this correlation is largely unknown. But even if it is no larger than 0.1, the resulting topographic stress could be comparable in magnitude to that from the surface wind.

Kinetic energy input via a sub-grid scale (SGS) parameterization is perhaps the central weakness of the Neptune parameterization, where a more complete theory provides for total energy conservation (forcing and dissipation) at the SGS. Notably, no such SGS theory presently exists. Instead, numerical modelers

traditionally employ energy sinks via frictional dissipation toward a state of rest, as well as thickness diffusion to a state of zero available potential energy, both with no corresponding source except that available via forcing by surface momentum and buoyancy fluxes. [Holloway \(1992\)](#) argues that such is not an appropriate modeling practice, thus motivating the use of Neptune as a interim approach until a more complete theory is available.

There is concern about the stepwise representation of bottom topography in z-coordinate models. The use of partial cells in MOM enhances the model's ability to represent the topographic slope in a more faithful manner than with full cells. In general, however, the Neptune parameterization is an attempt to instruct the simulation about physical consequences due to eddy-topography interactions. The hope is that if the model can be suitably informed about these effects, it matters little that topography is only approximately represented.

The Neptune parameterization is implemented in MOM for both Laplacian friction and biharmonic friction. Neptune systematically damps the deformation rates towards a state of approximate maximum entropy. Notably, Neptune increases the kinetic energy as it damps the flow toward its maximum entropy state. Such occurs even for flows at rest experiencing no external forcing. As a result, Neptune spins-up the flow with shallow depths to the right of the downstream direction.

26.2 Basics of the parameterization in MOM

If the view is taken that equations of motion are solved for moments of probable flow, then those moments are forced in part by derivatives of the entropy distribution with respect to the realized moments. The entropy gradient is estimated as proportional to a departure of the realized moments from a state in which the entropy gradient is weak. This latter state is approximated by a depth-integrated transport streamfunction ψ^{nep} (units of volume per time) and maximum entropy velocity \mathbf{u}^{nep} .

26.2.1 The Eby and Holloway (1994) approach

An early suggestion for the streamfunction was given by [Eby and Holloway \(1994\)](#), who wrote

$$\psi^{\text{nep}} = -f L^2 H \quad (26.1)$$

$$\mathbf{u}^{\text{nep}} = \hat{\mathbf{z}} \wedge H^{-1} \nabla \psi^{\text{nep}} \quad (26.2)$$

where f is the Coriolis parameter, H is the ocean depth, and L is an adjustable length scale on the order of 10 km. If model resolution is coarse relative to the first deformation radius, \mathbf{u}^{nep} is roughly depth independent. Depth independence is assumed in the MOM implementation of Neptune. To parameterize the unresolved driving of the mean flow by eddy-topography interactions, the Neptune parameterization drives flow towards \mathbf{u}^{nep} using an eddy stress proportional to $(\mathbf{u}^{\text{nep}} - \mathbf{u})$. The [Eby and Holloway \(1994\)](#) method is implemented in MOM just for the Laplacian friction operator.

26.2.2 Maltrud and Holloway (2008) approach

For the biharmonic implementation of Neptune, we follow the approach of [Maltrud and Holloway \(2008\)](#), in which

$$\mathbf{u}^{\text{nep}} = - \left(\frac{f L^2}{H + H_{\min}} \right) \hat{\mathbf{z}} \wedge \nabla H. \quad (26.3)$$

Contrary to the [Eby and Holloway \(1994\)](#) implementation, there is no gradient acting on the Coriolis parameter in the [Maltrud and Holloway \(2008\)](#) approach. The minimum depth H_{\min} regularizes the scheme so that the velocity \mathbf{u}^{nep} does not get too large in very shallow water.

26.2.3 Specifying the length scale

As in [Eby and Holloway \(1994\)](#) and [Maltrud and Holloway \(2008\)](#), MOM provides the following specification for the length scale L in both of the above approaches

$$L = \gamma_{\text{neptune scaling}} \left[L_{\text{pole}} + (L_{\text{equator}} - L_{\text{pole}}) \left(\frac{1 + \cos 2\phi}{2} \right) \right], \quad (26.4)$$

where L_{pole} is the polar length scale, L_{equator} is the equatorial length scale, ϕ is the latitude, and $\gamma_{\text{neptune scaling}}$ is a dimensionless scaling coefficient used for further tuning. The tuning coefficient $\gamma_{\text{neptune scaling}}$ was set to unity in [Maltrud and Holloway \(2008\)](#), though they provided an extensive suite of smoothing operations on the bottom slope.

26.3 Topostrophy diagnostic

The obvious means for diagnosing the impacts of Neptune are to directly compare the simulated velocity field with versus without the parameterization. However, as noted by [Holloway et al. \(2007\)](#); [Merryfield and Scott \(2007\)](#); [Maltrud and Holloway \(2008\)](#); [Holloway \(2008\)](#), it is simpler to consider a scalar field, known as *topostrophy*, that directly measures the orientation of the velocity relative to the gradient of the topography

$$T^{\text{topostrophy}} = f(\hat{\mathbf{z}} \wedge \mathbf{u}) \cdot \nabla H. \quad (26.5)$$

This diagnostic is coded in MOM within the `ocean_velocity_diag.F90` module. The neptune parameterization tends to increase the topostrophy, with the trend consistent with topostrophy in eddying simulations and observations ([Holloway et al., 2007](#); [Merryfield and Scott, 2007](#); [Maltrud and Holloway, 2008](#); [Holloway, 2008](#)).

Ad hoc subgrid scale parameterizations

The purpose of this part of the manual is to describe certain of the subgrid scale (SGS) parameterizations of physical processes used in MOM, with the focus here on certain *ad hoc* approaches.

OVERFLOW SCHEMES

Contents

27.1	Motivation for overflow schemes	380
27.2	The sigma transport scheme	380
27.2.1	Sigma diffusion	380
27.2.2	Sigma advection	381
27.2.2.1	Sigma velocity derived from resolved velocity	382
27.2.2.2	Sigma velocity from a parameterization	382
27.2.2.3	Maintaining mass conservation	383
27.2.2.4	Dia-sigma transport	383
27.2.2.5	Mass sources	384
27.2.2.6	Undulating sigma layer thickness	384
27.2.2.7	Problems with the sigma advection scheme	384
27.2.3	Implementation of sigma transport	385
27.3	The Campin and Goosse (1999) scheme	385
27.3.1	Finding the depth of neutral buoyancy	386
27.3.2	Prescribing the downslope flow	387
27.3.3	Mass conservation and tracer transport	387
27.3.4	Implementation in MOM	388
27.3.4.1	Start of the integration	388
27.3.4.2	During a time step	388
27.4	Neutral depth over extended horizontal columns	389
27.5	Sigma friction	391

The purpose of this chapter is to detail various methods available in MOM for enhancing the transport of dense water downslope. Some of methods are implementations of schemes from the literature, some are unique to MOM, and some remain incomplete methods which are part of MOM only for use by those actively pursuing research into overflow algorithms. We present our prejudices in Section 27.4, though the user should recognize that much research still is underway towards finding a suitable overflow scheme for global ocean climate modelling.

There are four methods implemented in MOM described in this chapter, with the following modules containing the code:

```
ocean_param/lateral/ocean.sigma.transport.F90
ocean_param/lateral/ocean.mixdownslope.F90
ocean_param/sources/ocean.overflow.F90
ocean_param/sources/ocean.overexchange.F90.
```

27.1 Motivation for overflow schemes

As described by [Winton et al. \(1998\)](#), coarse resolution z-coordinate models generally have difficulty moving dense water from shallow to deep regions. The key problem is that too much dense water spuriously entrains with the ambient lighter fluid. Only when the topographic slope is resolved so that the grid spacing satisfies

$$|\nabla H| \leq \frac{\Delta z}{\Delta s}, \quad (27.1)$$

does the simulation begin to reach negligible levels of spurious entrainment. Resolving a slope of 1/100 with vertical resolution of $\Delta z = 20\text{m}$ thus requires horizontal grid spacing $\Delta s \approx 2\text{km}$. This resolution is one or two orders finer than the typical resolution of the 1-2 degree ocean climate models commonly used today. Furthermore, refined vertical resolution, desired for representing vertical physical processes, requires one to further refine the horizontal resolution required to resolve the slope. Notably, there is little difference between the representation of steeply sloping features via either full or partial steps in z-models (Section 5.1.2). Hence, steep “cliff” features remain ubiquitous in the typical ocean climate model using vertical coordinates with quasi-horizontal isosurfaces. Short of respecting the constraint (27.1), traditional tracer transport schemes (i.e., vertical convection; horizontal and vertical diffusion; and horizontal and vertical advection) are generally unable to transport dense waters into the abyss to the extent observed in Nature. This problem with spurious entrainment is shared by the quasi-horizontal vertical coordinates such as those discussed in Chapter 5.

In an attempt to resolve the spurious entrainment problem, modelers have formulated ways to embed terrain following transport schemes into geopotential or pressure coordinate models. These schemes generally assume the bottom ocean region is turbulent, and so well mixed and not subject to geostrophy. The resulting dynamics act to bring water downslope, eventually being entrained at a neutral buoyancy depth.

Some approaches aim to modify both the momentum and tracer equations, with [Killworth and Edwards \(1999\)](#) documenting a most promising approach. Unfortunately, when modifying the momentum equation so that pressure gradients are computed within the bottom boundary layer, difficulties handling this calculation have resulted in nontrivial problems with spurious transport, especially near the equator. It is for this reason that no global ocean climate model presently employs the [Killworth and Edwards \(1999\)](#), or analogous, scheme. Motivated by this difficulty, we *do not* consider any scheme in MOM that modifies the momentum equation. Instead, we focus exclusively on methods restricted to the tracer equation.

27.2 The sigma transport scheme

This section documents the scheme available in the module

`ocean_param/mixing/ocean-sigma-transport.F90`

The papers by [Beckmann and Döscher \(1997\)](#) and [Döscher and Beckmann \(2000\)](#) propose a method to incorporate a rudimentary terrain following turbulent layer in z-models, or more generally into any model with vertical coordinates having quasi-horizontal isosurfaces (e.g., pressure based vertical coordinates). They prescribe changes only to the tracer equation, in which there is advection and diffusion within a bottom turbulent layer. We term these transport mechanisms *sigma diffusion* and *sigma advection*, since the sigma vertical coordinate (Section 5.1.5) is terrain following.

By enabling a terrain oriented route for tracer transport, in addition to the usual grid oriented transport, the quasi-horizontal vertical coordinate models are now afforded an extra pathway for transporting dense water into the abyss.

27.2.1 Sigma diffusion

Diffusion oriented according to the bottom topography is referred to as *sigma diffusion* in the following. The diffusive flux between two adjacent cells living at the ocean bottom is given by

$$\mathbf{F}_\sigma = -A \nabla_\sigma T, \quad (27.2)$$

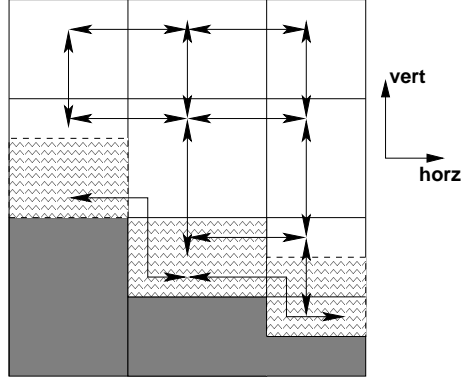


Figure 27.1: Schematic of the along-topography pathway for tracer transport afforded by the sigma transport scheme in MOM. Darkened regions denote land cells, and lightly hatched regions are within the bottom turbulent boundary region. This boundary layer generally can fit within a single bottom cell, as in the left turbulent boundary region; occupy a full cell, as in the middle region; or require more than one of the bottom cells, as in the right region. Tracers communicate with their grid aligned horizontal and vertical neighbors via the usual advection, diffusion, and convective processes. Tracers in the bottom turbulent layer can additionally communicate with their neighbors within the turbulent region via sigma diffusion and sigma advection.

with ∇_σ the horizontal gradient operator taken between cells in the sigma layer. Note that this flux vanishes if the tracer concentration is the same between two adjacent cells within the sigma layer. We follow the approach of [Döscher and Beckmann \(2000\)](#) in which sigma diffusion is strong when densities of the participating cells favors downslope motion. That is, the following diffusivity is used

$$A = \begin{cases} A_{\max} & \text{if } \nabla_\sigma \rho \cdot \nabla H < 0 \\ A_{\min} & \text{if } \nabla_\sigma \rho \cdot \nabla H \geq 0, \end{cases} \quad (27.3)$$

where $z = -H(x, y)$ is the bottom depth. Note that in practice, this constraint is applied separately in the two horizontal directions. That is, the zonal diffusivity is large if $\rho_{,x} H_{,x} < 0$ and the meridional diffusivity is large if $\rho_{,y} H_{,y} < 0$. A ratio of the two diffusivities A_{\max}/A_{\min} is a namelist parameter in MOM, with $\approx 10^6$ the default value as suggested by [Döscher and Beckmann \(2000\)](#).

An additional velocity dependent diffusion was also found by [Döscher and Beckmann \(2000\)](#) to be of use. In this case, an added sigma-diffusive flux in the zonal direction is computed using the diffusivity

$$A = \begin{cases} |u| \Delta x & \text{if } \rho_{,x} H_{,x} < 0 \text{ and } u H_{,x} > 0 \\ A_{\min} & \text{otherwise.} \end{cases} \quad (27.4)$$

In this expression, $|u|$ is the magnitude of the model's resolved zonal velocity component within the sigma layer, and Δx is the zonal grid spacing. An analogous meridional flux is computed as well.

Sigma diffusion can be specified to occur over an arbitrary layer thickness, even if this layer encompasses a non-integer number of bottom cells. If sigma diffusion is enabled *without* sigma advection, then this bottom layer is time independent.

27.2.2 Sigma advection

In addition to sigma diffusion, MOM allows for an advective contribution to the bottom boundary layer flow. This portion of the algorithm is experimental, and so *it is not recommended for general use*. We present the discussion here only to expose some initial thoughts on a possible new method, but recognize that the method as discussed here is incomplete. Note that the sigma advection scheme discussed here is distinct from the method proposed by [Beckmann and Döscher \(1997\)](#). Instead, the [Campin and Goosse](#)

(1999) scheme discussed in Section 27.3 employs an analogous advective transport method which has been implemented in MOM.

In the sigma advection scheme in MOM, there are two ways to determine the advective velocity components acting on tracers within the sigma layer. In both cases, if the deeper parcel within the sigma layer is denser than the shallower parcel, then the sigma advective transport is set to zero. Otherwise, it is active and thus contributes to the downslope tracer transport. This criteria translates into the constraint

$$\nabla_{\sigma} \rho \cdot \nabla H < 0 \quad \text{for density driven downslope flow,} \quad (27.5)$$

where ρ is the density within the bottom sigma layer. This constraint is the same as used to determine the value for the diffusivity discussed in Section 27.2.1.

27.2.2.1 Sigma velocity derived from resolved velocity

Beckmann and Döscher (1997) and Döscher and Beckmann (2000) determine the advective velocity components acting in the sigma layer from the model's resolved velocity components. In MOM, these velocity components are found by integrating the model's resolved horizontal advective velocity components within the bottom turbulent sigma layer.

27.2.2.2 Sigma velocity from a parameterization

Campin and Goosse (1999) suggest an additional approach to enhance the horizontal velocity available for downslope flow. In MOM, we add this velocity to the resolved velocity within the sigma layer determine as above.

Following Campin and Goosse (1999), assume the dense shallow parcel has a subgrid scale momentum associated with its downslope motion. The zonal momentum is assumed to be proportional to the topographic slope, $H_{,x}$, the acceleration from gravity, g , the amount of fluid within the cell participating in the downslope flow,

$$0 \leq \delta \leq 1, \quad (27.6)$$

and the density difference

$$\Delta \rho = dx \left(\frac{\partial \rho}{\partial x} \right)_{\sigma} \quad (27.7)$$

as measured in the zonal direction within the sigma layer. The momentum is retarded by frictional dissipation, μ (with units of inverse time). These considerations then lead to the momentum balance

$$\rho V^{(t)} \mu u^{\text{slope}} = -g \delta V^{(t)} \Delta \rho H_{,x} \text{sign}(H_{,x}) \quad (27.8)$$

where

$$V^{(t)} = dx dt * dy dt * dz t^{\sigma} \quad (27.9)$$

is the volume of the dense parcel within the sigma layer, we assume $\Delta \rho H_{,x} < 0$, as required for density favorable downslope flow (equation (27.5)), and $\text{sign}(H_{,x})$ sets the sign for the downslope velocity. Equation (27.8) is also used to determine a meridionally directed downslope transport, with the meridional topographic slope $H_{,y}$ replacing $H_{,x}$, and $\Delta \rho = dy \rho_{,y}$ the density difference between meridionally adjacent parcels.

Solving equation (27.8) for the velocity component u^{slope} yields

$$\rho dz t^{\sigma} u^{\text{slope}} = - \left(\frac{g \delta}{\mu} \right) H_{,x} \Delta \rho dz t^{\sigma} \text{sign}(H_{,x}). \quad (27.10)$$

With the depth H referring to the depth of a tracer cell, the slope $H_{,x}$ is defined at the zonal face of the tracer cell. Hence, the velocity component u^{slope} is likewise positioned at the zonal face. This is the desired position for the zonal advective tracer transport velocity component.

Campin and Goosse (1999) suggest the values for frictional drag

$$\mu = 10^{-4} \text{sec}^{-1} \quad (27.11)$$

and fraction of a cell participating in the transport

$$\delta = 1/3. \quad (27.12)$$

These parameters are namelists in MOM4. Using these numbers, with an absolute topographic slope of $|H_{,x}| \approx 10^{-3}$ and density difference $\Delta\rho \approx 1 \text{ kg m}^{-3}$, leads to

$$u^{\text{slope}} \approx .03 \text{ m sec}^{-1} \quad (27.13)$$

and the associated volume transport

$$U^{\text{slope}} = u^{\text{slope}} \times \text{thickness_sigma} \times \text{dyl}, \quad (27.14)$$

where `thickness_sigma` is the thickness of the sigma layer. With $u^{\text{slope}} \approx .03 \text{ ms}^{-1}$ corresponding to the speed of fluid within a sigma layer that is one-degree in width and 50m in thickness, we have a volume transport $U^{\text{slope}} \approx 0.2 \text{ Sv}$. Larger values are realized for steeper slopes, larger density differences, larger grid cells, and thicker sigma layers.

27.2.2.3 Maintaining mass conservation

Introducing horizontal advection within the sigma layer necessitates the consideration of mass conservation within this layer. Our focus here is just on the additional mass conservation issues arising from sigma advective transport.

The balance of mass within an arbitrary layer is detailed in Section 10.8. Assuming there is no transport through the bottom of the sigma layer into rock, we are led to the mass budget for the sigma layer (equation (10.108))

$$0 = \partial_t (dz \rho) - dz \rho \mathcal{S}^{(M)} + \nabla_\sigma \cdot (dz \rho \mathbf{u}) + (\rho w^{(\sigma)})_{\text{top of sigma layer}}. \quad (27.15)$$

Again, each term in this equation is associated *just* with the sigma transport process. Hence, the horizontal velocity \mathbf{u} as that obtained from the considerations given earlier in this section. However, the remaining terms have not been specified yet, and so must be set according to physical arguments and/or convenience.

27.2.2.4 Dia-sigma transport

First, consider the case of zero mass source arising from sigma transport, and a zero time tendency term $\partial_t (dz \rho)$ (such as occurs in the Boussinesq case assuming a constant sigma layer thickness). The mass budget within the sigma layer is thus closed by a dia-sigma transport

$$(\rho w^{(\sigma)})_{\text{top of sigma layer}} = -\nabla_\sigma \cdot (dz \rho \mathbf{u}). \quad (27.16)$$

This transport measures the amount of water that crosses the sigma layer from the surrounding fluid. This choice was taken by Beckmann and Döscher (1997) and Döscher and Beckmann (2000), and it was also employed by Campin and Goosse (1999).

Furthermore, Beckmann and Döscher (1997) suggest that to reduce the spurious entrainment associated with tracer advection aligned with the model's grid, it is appropriate to reduce, or remove, this advective transport within the sigma layer in favour of the sigma advection transport. Tang and Roberts (2005) also take this approach. Nonetheless, we do not follow this suggestion for the following reasons. First, it complicates the treatment of the advection operator by introducing an *ad hoc* parameter that partitions between sigma advection and grid aligned advection. Second, and primarily, we take the perspective that the sigma advection process is subgrid scale. Hence, it should act only in those cases where the resolved, grid aligned, velocity is unable to provide a sufficient downslope transport. We should thus *not* remove the grid aligned advective transport using an *ad hoc* specification. That is, we do not aim to remedy spurious entrainment arising from grid aligned advective transport by removing this transport altogether.

Given these objections, we do not pursue this approach further in MOM.

27.2.2.5 Mass sources

Next, consider the case where all mass is advected downslope within the sigma layer, with a zero time tendency term $\partial_t(dz \rho)$ and zero dia-sigma transport.¹ This assumption then leads to the sigma layer mass budget

$$dz \rho \mathcal{S}^{(M)} = \nabla_\sigma \cdot (dz \rho \mathbf{u}). \quad (27.17)$$

That is, the divergent horizontal advective transport within the sigma layer is balanced by a nonzero mass source. The horizontal integral of the mass source over the sigma layer vanishes, since the sigma advection velocity satisfies either the no-normal boundary condition at land/sea interfaces, or periodicity. Hence, the introduction of the nonzero mass source does not corrupt global mass conservation. It does, however, come at the cost of also requiring nonzero tracer sources; the introduction of new fluid locally requires also the introduction of nonzero tracer locally, since the fluid has some tracer content (e.g., a temperature). These tracer sources do not necessarily lead to a zero global net introduction of tracer. This approach is thus unacceptable.

27.2.2.6 Undulating sigma layer thickness

The time tendency $\partial_t(dz \rho)$ represents changes in the density weighted sigma layer thickness. It vanishes for a Boussinesq case if the sigma layer has constant thickness. However, if the sigma layer can inflate or deflate, this term remains nonzero. That is, without mass sources or without dia-sigma transport, the mass budget within the sigma layer takes the form

$$\partial_t(dz \rho) = -\nabla_\sigma \cdot (dz \rho \mathbf{u}). \quad (27.18)$$

Hence, the sigma layer undulates according to the convergence or divergence of mass advected within the layer. Its undulations are of just the magnitude needed to keep a zero dia-sigma transport. So the picture is of a blob of heavy fluid moving downslope, causing the sigma layer to undulate in order to accomodate the fluid motion. See Figure 2.7 for an illustration of this fluid motion. This approach is available only for models such as MOM that allow an arbitrary time dependent thickness for the sigma layer.

During some initial research, we have favoured this approach in MOM as it avoids objections raised about the previous alternatives. We do make some simplifications, and note that the approach has only recently (as of 2006) been tested, with some unfortunate problems discussed below. Here are some things to note.

- For the nonBoussinesq case, we replace the *in situ* density appearing in the time tendency with the constant Boussinesq density ρ_0 . Given uncertainties in many of the scheme's parameters, this replacement is justified.
- The sigma layer thickness is bounded from above and below by user specified values. Allowing the thickness to vary too far can lead to noisy behaviour. Settings bounds amounts to an implicit specification of detrainment whenever the thickness gets too large, and entrainment when it gets too small.
- It has proven useful to smooth the sigma layer thickness. An option is available to smooth the layer thickness with a Laplacian diffusion operator.

27.2.2.7 Problems with the sigma advection scheme

The most fundamental problem with the sigma advection scheme, as implemented according to equation (27.18), is that as mass converges to a region to thus expand the sigma layer, there is no corresponding dynamical mechanism to carry this perturbation away, and thus allow for an adjustment process. Instead, by only considering the mass conservation equation, with no dynamical equations, the sigma layer will generally grow without bound in regions where mass converges, or disappear in regions of divergence. This situation is not encountered in an isopycnal model, since these models have dynamical processes to adjust the fluctuating thicknesses.

¹Grid aligned advection generally leads to transport across the sigma layer.

Absent a dynamical mechanism for the adjustment, the sigma advection scheme must employ artificial limits on the layer thickness. These limiters impose, in effect, a detrainment or entrainment process to keep the layer thickness within the specified bounds. Such processes, however, have not yet been implemented in MOM, so the present scheme is incomplete. Without the entrainment and detrainment processes, the artificial limits, when imposed, allow for the tracer to realize extrema, since its time tendency is artificially altered. This is unacceptable, and so the scheme as presently implemented is unusable.

27.2.3 Implementation of sigma transport

Consistent with [Beckmann and Döscher \(1997\)](#), the turbulent bottom layer momentum equations remain the same as interior k-level cells. We now just allow tracers in the bottom turbulent layer to be affected by transport with their “sigma-neighbors” in addition to their horizontal and vertical neighbors. Figure 27.1 provides a schematic of the extra pathway available with sigma tracer transport.

The bottom turbulent sigma layer in MOM3 was appended to the very bottom of the model, and so effectively lived beneath the deepest rock. This approach is inconvenient for the following reasons.

- It makes for awkward analyses.
- It precludes direct comparison between models run with and without sigma-physics since the grid used by the two models is different.
- It makes it difficult to consider convergence when refining the grid mesh.

For these reasons, the bottom turbulent layer in MOM4 is included within the regular model domain. This is the approach used by [Beckmann and Döscher \(1997\)](#) (e.g., see their Figures 1 and 2).

The disadvantage of the MOM4p0 approach is that the bottom turbulent layer thickness *thickness_sigma* has a generally non-constant thickness and is determined by the thickness of the grid cell next to topography. In particular, with partial bottom steps, the effective turbulent layer thickness could be very thin, in which case thin cells act as a bottle-neck to bottom transport. This implementation is inconvenient.

In MOM, we allow for an arbitrarily thick bottom turbulent layer. We do so by incorporating the required grid cells into the bottom turbulent sigma layer. This approach requires some added accounting, but it is straightforward. In particular, the tracer concentration within the bottom turbulent layer is computed by

$$T_{\text{sigma}} = \frac{\sum_{\text{sigma}} \rho \, dz \, T}{\sum_{\text{sigma}} \rho \, dz}, \quad (27.19)$$

where the sum extends over the cells, including cell fractions, contained in the bottom turbulent layer. A time tendency is computed for sigma transport of T_{sigma} within the bottom turbulent sigma layer. The relative fraction of a grid cell participating in the bottom turbulent layer determines the magnitude of the tracer time tendency added to this cell.

27.3 The Campin and Goosse (1999) scheme

This section documents the scheme available in the module

`ocean_param/sources/ocean_overflow.F90.`

Consider a heavy water parcel sitting on top of a shelf/cliff that is horizontally adjacent to a lighter parcel sitting over a deeper water column. We may expect that the dense parcel will move off the shelf, down the slope, and into the deep. Along the way, entrainment will occur, with many important processes determining the details of the final water mass. This is indeed a cartoon of an important oceanic process forming much of the deep and intermediate waters in the ocean. Unfortunately, without some extra “engineering” help, [Winton et al. \(1998\)](#) show that coarse resolution z-models are incapable of providing the proper dynamical pathways for this transfer of dense shelf water into the deep. [Beckmann and Döscher \(1997\)](#) suggest one means to enhance the representation of this process, and we discussed this scheme in Section 27.2. [Campin and Goosse \(1999\)](#) propose yet another, which we detail in this section. Both schemes only affect the tracer equation.

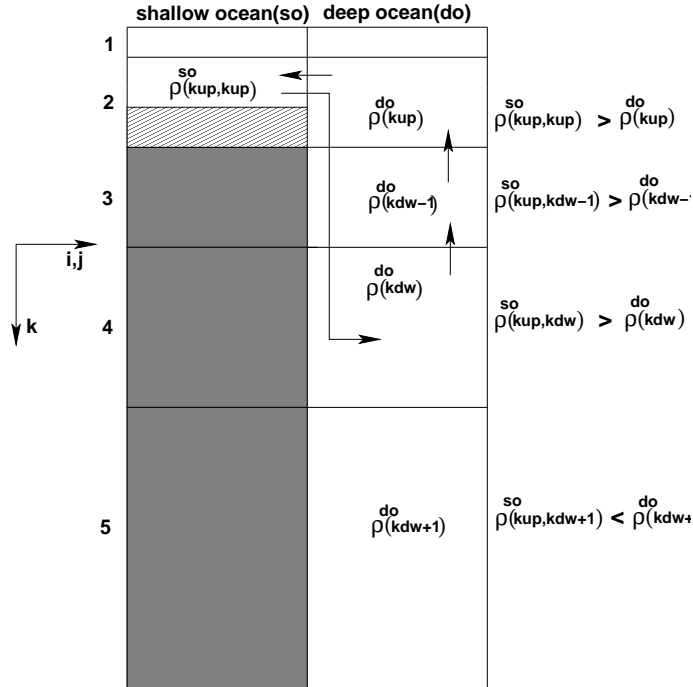


Figure 27.2: Schematic of the [Campin and Goosse \(1999\)](#) overflow method in the horizontal-vertical plane. The darkly filled region represents bottom topography using MOM4’s full cells. The lightly filled region represents topography filled by a partial cell. Generally, the thickness of a cell sitting on top of a topographic feature, as the $k = 2$ cell in the “so” column, is thinner than the corresponding cell in the deep-ocean column (the $k = 2$ cell in the “do” column). Shown are tracer cells, with arrows representing the sense of the scheme’s upstream advective transport. This figure is based on Figure 1 of [Campin and Goosse \(1999\)](#).

27.3.1 Finding the depth of neutral buoyancy

Figure 27.2 illustrates a typical situation in a horizontal-vertical plane. Here, we see a heavy parcel of *in situ* density $\rho^{so}(k = kup)$ sitting horizontally adjacent to a lighter parcel of *in situ* density $\rho^{do}(k = kup)$. The superscript “so” refers to water in the “shallow ocean” column, whereas “do” refers to water in the “deep ocean” column.

If the heavy parcel is allowed to adiabatically move off the shelf and then vertically within the deep column, it will equilibrate at its depth of neutral buoyancy. To compute the depth of neutral buoyancy, we evaluate the *in situ* density for the parcel taken at the local value for the *in situ* pressure of the environment where it may potentially equilibrate. For the example shown in Figure 27.2, with (i, j) setting the horizontal position of the shelf parcel and $(i + 1, j)$ setting the horizontal position of the deep column, we have

$$\rho^{so}(kup, kup) = \rho(s_{i,j,kup}, \theta_{i,j,kup}, p_{i,j,kup}) \quad (27.20)$$

$$\rho^{so}(kup, kdw - 1) = \rho(s_{i,j,kup}, \theta_{i,j,kup}, p_{i+1,j,kdw-1}) \quad (27.21)$$

$$\rho^{so}(kup, kdw) = \rho(s_{i,j,kup}, \theta_{i,j,kup}, p_{i+1,j,kdw}) \quad (27.22)$$

$$\rho^{so}(kup, kdw + 1) = \rho(s_{i,j,kup}, \theta_{i,j,kup}, p_{i+1,j,kdw+1}). \quad (27.23)$$

That is, we compute the density at the salinity and potential temperature of the shallow ocean parcel, $(s_{i,j,kup}, \theta_{i,j,kup})$, but at the *in situ* pressure for the respective grid cell in the deep column. The density is then compared to the density of the parcel at the *in situ* salinity, temperature, and pressure of the cells in the deep ocean column.

27.3.2 Prescribing the downslope flow

Following [Campin and Goosse \(1999\)](#), we assume that the dense parcel has a downslope momentum imparted to it. This momentum is proportional to the topographic slope, $H_{,x}$, the acceleration from gravity, g , the amount of fluid within the cell participating in the downslope flow,

$$0 \leq \delta \leq 1, \quad (27.24)$$

and the positive density difference

$$\Delta\rho = \rho^{so}(\text{kup}, \text{kup}) - \rho^{do}(\text{kup}) > 0. \quad (27.25)$$

The momentum is retarded by frictional dissipation, μ (in units of inverse time). These considerations then lead to the momentum balance

$$\rho_o V^{(t)} \mu u^{slope} = g \delta V^{(t)} \Delta\rho |H_{,x}| \quad (27.26)$$

where

$$V^{(t)} = dx t * dy t * dz t \quad (27.27)$$

is the volume of the dense parcel's tracer cell. Equation (27.26) is also used to determine a meridionally directed downslope transport, with the meridional topographic slope $\partial_y H$ replacing $\partial_x H$, and $\Delta\rho$ the density difference between meridionally adjacent parcels.

Solving equation (27.26) for the speed u^{slope} yields

$$u^{slope} = \left(\frac{g \delta}{\rho_o \mu} \right) |\partial_x H| \Delta\rho. \quad (27.28)$$

If the depth H refers to the depth of a tracer cell, then the absolute slope $|\partial_x H|$ is naturally defined at the zonal face of the tracer cell. Hence, the speed, u^{slope} , is likewise positioned at the zonal face. This is the desired position for an advective tracer transport velocity.

[Campin and Goosse \(1999\)](#) suggest the values $\mu = 10^{-4} \text{sec}^{-1}$ and $\delta = 1/3$. These parameters are set as namelists in MOM4. Using these numbers, with an absolute topographic slope of $|H_{,x}| \approx 10^{-3}$ and density difference $\Delta\rho \approx 1 \text{ kg m}^{-3}$, leads to the speed

$$u^{slope} \approx .03 \text{ m sec}^{-1}. \quad (27.29)$$

Associated with this downslope speed is a volume transport of fluid leaving the cell

$$U^{slope} = u^{slope} dz t_{min} dy t. \quad (27.30)$$

In this equation, $dz t_{min}$ is the minimum thickness of the shelf cell and the adjacent cell. This minimum operation is necessary when considering MOM4's bottom partial cells, whereby the bottom-most cell in a column can have arbitrary thickness (Figure 27.2). With $u^{slope} \approx .03 \text{ m s}^{-1}$ corresponding to the speed of fluid leaving a grid cell that is one-degree in width and 50m in depth, we have a volume transport $U^{slope} \approx 0.2 \text{ Sv}$. Larger values are easily realized for steeper slopes, larger density differences, and larger grid cells.

27.3.3 Mass conservation and tracer transport

To conserve mass throughout the system, the mass flux exiting the shelf cell and entering the deep cell must itself be returned from the adjacent cell. This situation then sets up a mass flux throughout the participating cells, where there is zero convergence of the flux and so zero net increase or decrease in mass. For the Boussinesq fluid, mass conservation is replaced by volume conservation. This redirected plumbing is shown in Figure 27.2.

The convergence-free seawater mass flux carries with it tracer mass. If there are differences in the tracer content of the cells, then the tracer flux will have a nonzero convergence, and so it moves tracer throughout the system. We use first-order upstream advective transport as a discretization of this process. First-order upstream advection is the simplest form of advection. Its large level of numerical diffusion is consistent with our belief that the bottom layer flows in the real ocean near steep topography are quite turbulent. Hence, although inappropriate for interior flows, we are satisfied with the use of upstream advection for the overflow scheme.

27.3.4 Implementation in MOM

This section details the implementation of the [Campin and Goosse \(1999\)](#) scheme in MOM.

27.3.4.1 Start of the integration

At the start of the model integration, it is necessary to determine those grid points where it is possible to have a downslope flow. For this purpose, we introduce the array $\text{topog_step}(i, j, m)$, with $m = 1, 2, 3, 4$ specifying in a counter-clockwise direction the four surrounding columns whose depths are to be compared to that at the central (i, j) point. Figure 27.3 illustrates this notation. If the adjacent column is deeper than the central point, thus representing a possible direction for downslope flow, then $\text{topog_step}(i, j, m)$ for this value of m is set to unity. Otherwise, $\text{topog_step}(i, j, m)$ for this m is zero. Note that with partial bottom cells, it is possible for an adjacent column to be deeper yet for the number of vertical cells to be the same in both columns. To initiate the downslope scheme of [Campin and Goosse \(1999\)](#), we insist that there be at least one more grid cell in the adjacent column.

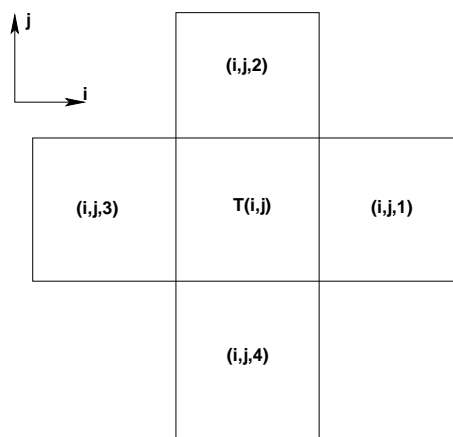


Figure 27.3: Plan view (x-y plane) of a tracer grid cell at (i, j) and its horizontally adjacent tracer cells. We label the adjacent cells $(i + 1, j), (i, j + 1), (i - 1, j), (i, j - 1)$ as $m = 1, 2, 3, 4$. Notice that we do not consider downslope flow along a diagonal direction.

27.3.4.2 During a time step

During each time step, we locate where downslope flow is favorable for points sitting on the ocean bottom at $(i, j, \text{kmt}(i, j))$. For each of the four directions ($m = 1, 2, 3, 4$) where $\text{topog_step}(i, j, m) = 1$, we check the density difference between the central point and the adjacent point. If the density of the central point is larger, then the [Campin and Goosse \(1999\)](#) scheme is used to initiate downslope transport. For these directions, we locate the depth of neutral buoyancy for the central point according to the discussion in Section 27.3.1, and so specify the number of vertical cells, kdw , participating in the transport. Note that we allow for downslope transport to occur in more than one direction, as occurs in those cases for a fixed (i, j) where $\text{topog_step}(i, j, m)$ has more than a single nonzero element.

Our prescription is mindful of the possibility for the shallow-cell to be a partially filled cell sitting on the topography. For this reason, the convergence-free volume transport associated with the downslope flow is weighted by the minimum vertical thickness of the two cells (equation (27.30)). Otherwise, it would be possible to flood a thin partial cell with a huge amount of tracer (e.g., heat).

We incorporate effects from the [Campin and Goosse \(1999\)](#) overflow scheme into MOM's tracer time tendency array. To derive the tendency, we proceed as for the river-mixing and cross-land mixing formulations discussed in [Griffies et al. \(2004\)](#) by focusing on the time evolution due to just the overflow process.

For the particular zonal-vertical case illustrated in Figure 27.3, we prescribe

$$\partial_t (V^{(t)} \rho C)_{i,j,kup}^{so} = \rho U^{slope} (C_{i+1,j,kup}^{do} - C_{i,j,kup}^{so}) \quad (27.31)$$

$$\partial_t (V^{(t)} \rho C)_{i+1,j,kup}^{do} = \rho U^{slope} (C_{i+1,j,kdw-1}^{do} - C_{i+1,j,kup}^{do}) \quad (27.32)$$

$$\partial_t (V^{(t)} \rho C)_{i+1,j,kdw-1}^{do} = \rho U^{slope} (C_{i+1,j,kdw}^{do} - C_{i+1,j,kdw-1}^{do}) \quad (27.33)$$

$$\partial_t (V^{(t)} \rho C)_{i+1,j,kdw}^{do} = \rho U^{slope} (C_{i,j,kup}^{so} - C_{i+1,j,kdw}^{do}), \quad (27.34)$$

where

$$\rho U^{slope} = u^{slope} \text{rho_dzt}_{min} \text{d}t, \quad (27.35)$$

with

$$\text{rho_dzt}_{(min)} = \min(\text{rho_dzt}_{i,j,kup}, \text{rho_dzt}_{i+1,j,kup}) \quad (27.36)$$

the minimum density weighted thickness of the two cells at $k = kup$. For the Boussinesq case, ρ factors are set to the constant reference density ρ_0 . Setting the tracer concentration to the same uniform value leads to vanishing time tendencies in each cell, thus reflecting volume/mass conservation. Additionally, summing these four equations leads to a vanishing right hand side, thus reflecting conservation of total tracer in the system. Since the downslope mixing has the form of an upstream advection, we discretize temporally by evaluating the tracer and density on the right hand side at the lagged time $\tau - 1$.

27.4 Neutral depth over extended horizontal columns

Both [Campin and Goosse \(1999\)](#) and [Beckmann and Döscher \(1997\)](#) provide quasi-physical approaches to the problem of simulating deep water formation near topography. Each provides plumbing routes beyond the local horizontal-vertical routes available in geopotential or pressure models. In this way, these methods afford a new means for *representing* the flows. Questions such as parameterizing the rates of entrainment, volume flux, etc. (e.g., [Killworth and Edwards \(1999\)](#)) are not directly addressed by these schemes, although the present schemes can be extended a bit to include such details.

Climate modelers generally gauge the utility of overflow schemes on the overall results. Namely, do the schemes provide a route for deep water formation near topographic gradients in a manner expected from observations? Details of the transport are often not the first priority. This situation is unsatisfying from a process physics perspective. It may, nonetheless, be the best available for many coarse resolution models.

In this section, we discuss our prejudices with MOM development. To start, consider the density structure in Figure 27.4. This figure illustrates a case where the sigma transport scheme of Section 27.2 does not prescribe enhanced downslope transport. The reason is that the sigma transport scheme only works with density within the bottom “sigma layer”. For this example, density at the bottom of the deeper column is greater than that on the shelf, and so there is no enhanced transport prescribed. In contrast, the [Campin and Goosse \(1999\)](#) scheme prescribes a downslope transport, with the dense shelf water moving to its neutral buoyancy depth. It is for this reason that we favour in MOM those downslope schemes where the depth of neutral buoyancy is determined, with this depth possibly above the ocean bottom.

Even within this example, however, there remain limitations of the [Campin and Goosse \(1999\)](#) scheme. The limitation is that their scheme only reaches out one grid box in the horizontal. That is, although the scheme is non-local in the vertical, it remains local in the horizontal. What can happen is the dense parcel will find itself denser than any parcel in the adjacent column, and so its resting place, with the [Campin and Goosse \(1999\)](#) algorithm, is at the bottom of the adjacent column, rather than at a neutral buoyancy depth. If given the opportunity to exchange with columns further removed from the central column, the parcel is afforded the opportunity to find a more suitable neutral buoyancy layer. This general result motivates us to consider two experimental schemes, whereby the notions of a neutral buoyancy level motivated from [Campin and Goosse \(1999\)](#) are extended to columns removed from the central column. As the parcel finds a more suitable resting place, it is assumed to exchange properties with the intermediate parcels, in a manner meant to represent entrainment as it moves downslope. The rates of transport remain a function of the topographic slope and the difference in density, just like the [Campin and Goosse \(1999\)](#) scheme.

There are two methods available in MOM for realizing these ideas. The first is implemented in the module

ocean_param/sources/ocean_overexchange.F90.

In this scheme, a dense shallow parcel is allowed to be transported horizontally over more than a single column, so long as it continues to remain on the bottom of the adjacent columns, thus affording it more opportunity to find its neutral buoyancy level. The exchange results in no net mass exchange between parcels, and so there is no need for an advective replumbing to be implemented, in contrast to the [Campin and Goosse \(1999\)](#) scheme. Here, the resolved dynamics adjust based on mixing of the water masses and the associated changes in density structure. This process then becomes directly analogous to the cross-land mixing formulation discussed in [Griffies et al. \(2004\)](#) and in Chapter 29. That is, we remove the intermediate cells from the process described in Section 27.3.4.2, and just focus on the single shallow and deep cell, thus leading to

$$\partial_t (V^{(t)} \rho C)_{i,j,kup}^{so} = \rho U^{slope} (C_{i+1,j,kdw}^{do} - C_{i,j,kup}^{so}) \quad (27.37)$$

$$\partial_t (V^{(t)} \rho C)_{i+1,j,kdw}^{do} = \rho U^{slope} (C_{i,j,kup}^{so} - C_{i+1,j,kdw}^{do}), \quad (27.38)$$

This parameterization is simpler to implement than the [Campin and Goosse \(1999\)](#) scheme, since we omit the intermediate cells from the process. This approach also does not rely on assumptions of a flow that may be set up in response to the exchange of fluid.

The second method is implemented in the module

ocean_param/mixing/ocean_mixdownslope.F90.

In this scheme, exchange of tracer occurs as a partial convective mixing process. We assume that a part of the shallow dense cell is transported downslope, and this then mixes with the intermediate cells with an efficiency proportional to the topographic slope and the density difference. In equations, we compute a combined mass of the mixed water according to

$$M_{sum} = M(s) + M(d), \quad (27.39)$$

where

$$M(s) = \gamma \text{rho_dzt}(s) \text{dat}(s) \quad (27.40)$$

is the mass of water in the shallow dense cell participating in the exchange, and

$$M(d) = \delta \text{rho_dzt}(d) \text{dat}(d) \quad (27.41)$$

is the mass of deep cell participating. In these equations, dat is the horizontal area of the cells, γ is the fraction of the shallow dense cell that is assumed to take part in the downslope transport, and δ is proportional to the topographic slope and the density difference between the shallow and deep parcel. If assumed to mix completely over a time step, then the resulting tracer concentration C_{mix} would be given by

$$M_{sum} C_{mix} = \gamma \text{rho_dzt}(s) \text{dat}(s) C(s) + \delta \text{rho_dzt}(d) \text{dat}(d) C(d). \quad (27.42)$$

Instead of mixing completely, which would require an adjustment process as in convection, we use the tracer concentration C_{mix} to deduce the following time tendencies which drive the cells toward the mixed concentration

$$\text{dat}(s) \text{tend}(s) = \left(\frac{M(s) \gamma}{\Delta t} \right) (C_{mix} - C(s)) \quad (27.43)$$

$$\text{dat}(d) \text{tend}(d) = \left(\frac{M(d) \delta}{\Delta t} \right) (C_{mix} - C(d)). \quad (27.44)$$

Given the mixed tracer concentration (27.42), we have

$$\text{dat}(s) \text{tend}(s) + \text{dat}(d) \text{tend}(d) = 0, \quad (27.45)$$

which reflects the conservation of tracer.

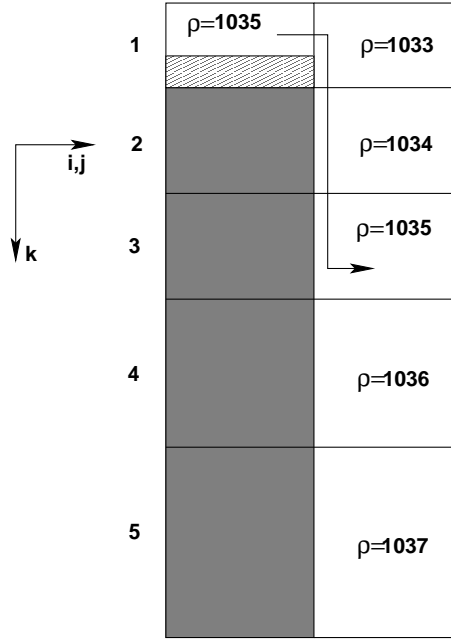


Figure 27.4: Schematic of a situation where a dense parcel sits on a shelf next to a column whose upper portion is light, but whose deeper portion is denser than the shelf. For this case, the [Campin and Goosse \(1999\)](#) scheme prescribes a transport between the shelf water at level 1 and the deeper water at level 3, with water bubbling upward to conserve mass as shown in Figure 27.2. In contrast, the sigma transport scheme will not prescribe any enhanced transport, since here the bottom of the deep column is denser than the shelf.

27.5 Sigma friction

The previous schemes introduce a new transport pathway for tracers. Nothing is done to the momentum equation. Another idea is to consider an enhancement of the vertical friction acting near to the bottom, with the friction introduced via a vertical viscosity. The effects of vertical friction are related, through geostrophy, to those from [Gent et al. \(1995\)](#), whereby density slopes are reduced without mixing of density classes ([Greatbatch and Lamb, 1990](#)). Alternatively, enhancing the vertical viscosity next to the bottom increases the Ekman layer thickness next to the bottom, and this breaks geostrophy, thus allowing for an easier downslope transit of the fluid. The scheme described in this section is *not* available in MOM. We mention it, nonetheless, as it may prove to be of use for some applications.

Following the scaling from [Campin and Goosse \(1999\)](#) discussed in Section 27.3.2, we define a vertical viscosity according to equation (27.26)

$$\begin{aligned} \kappa^\sigma &= dz t^\sigma u^{\text{slope}} \\ &= -\left(\frac{g \delta}{\mu \rho}\right) H_{,x} \Delta \rho dz t^\sigma \text{sign}(H_{,x}). \end{aligned} \quad (27.46)$$

Using the parameters from Section 27.3.2 leads to a vertical viscosity of

$$\kappa^\sigma = 0.15 \text{ m s}^{-2}, \quad (27.47)$$

with larger values for steeper topographic slopes and stronger density contrasts. We propose to introduce this viscosity throughout the sigma layer, and exponentially decrease it above the layer, with a relatively short decay scale

$$\kappa_{\text{decay}}^\sigma = 10 \text{ m}. \quad (27.48)$$

We suggest computing this viscosity separately for the two horizontal directions, and take the maximum of the two for the parameterization.

RIVER DISCHARGE INTO THE OCEAN MODEL

Contents

28.1	Introduction	393
28.2	General considerations	394
28.3	Steps in the algorithm	395

This chapter presents a method to distribute river runoff throughout a vertical column comprised of more than a single ocean model grid cell. This issue is most important for ocean models with fine vertical grid resolution. Care is taken to account for the needs to conserve properties using the constraints of a Boussinesq z-model where only the top cell is allowed to change its volume. The algorithm presented here is based on discussions with Mike Winton (Michael.Winton@noaa.gov).

We formulate the river mixing process in terms of tendencies added to the tracer concentration equations. Changes in the free surface height are handled just as for other forms of fresh water, such as precipitation and evaporation. This scheme was originally implemented in the Boussinesq z-model MOM4.0. It has been ported to the generalized level coordinate MOM4p1 and later MOM releases, only so far as adding the appropriate vertical grid factors. No more modifications have been made to exploit the added flexibility available with a generalized level model. Hence, the scheme is basically the same as that originally implemented in MOM4.0.

The following MOM module is directly connected to the material in this chapter:

ocean_param/sources/ocean_rivermix.F90

28.1 Introduction

Coupling rivers to an ocean model is necessary when building fully coupled climate models. For z-models, river discharge is typically given fully to the top model grid cell. Depending on the model resolution, dumping all the river properties to the top grid cell can cause problems. Notably, without enhanced mixing, a strong halocline can arise, with associated problems appearing due to noise from vertical advection across the strong front. This problem is enhanced in models with relatively fine vertical resolution, such as the 10m now common for the top grid cell in ocean climate models.

In the real world, there are two reasons that the halocline at river mouths is somewhat weaker than can occur in ocean climate models. First, river water does not generally fill only a single layer of some 10m depth. Instead, rivers discharge into the ocean over a vertical column whose depth can be deeper than 10m. Second, and more generally, river properties are mixed through a vertical column due to waves and tides near the coasts.

Two methods to relieve numerical problems can be considered. First, we can enhance vertical mixing of tracers in the region next to river mouths. This approach is straightforward and is available in MOM. In

detail, the enhanced mixing is strongest near the surface and tapers to zero at a specified depth. Such is the only method available to rigid lid ocean models for handling enhanced mixing at river mouths. Another method is to distribute the river water, along with its tracer content, over a pre-defined vertical column. Since the top model grid cell in the z-coordinate MOM4.0 is the only one capable of changing its volume through changes in the surface height, distributing river water into deeper cells must be done carefully. In the remainder of this chapter, we detail such a method. Its generalization to MOM4p1 and later releases is minimal, in that only the added vertical grid elements have been updated.

Note that we typically do not alter the transfer of momentum from the river to the ocean. Instead, we assume that river horizontal momentum is the same as the corresponding ocean cell, thus leading to no change in the ocean momentum associated with river discharge. This assumption may require modifications for careful studies of coastal processes, but it should be sufficient for ocean climate modeling.

28.2 General considerations

Consider a column of discrete ocean with kr cells in the vertical over which we aim to discharge river water:

$$kr = \# \text{ of vertical ocean cells into which river water is discharged.} \quad (28.1)$$

Allow the river to be discharging at a volume per area per time given by R , which has units of a velocity:

$$R = \text{volume per area per time of river water discharge.} \quad (28.2)$$

The river water flux R is distinguished in MOM from fresh water associated with evaporation and precipitation.

The tracer concentration within the river water is given by C_{river} :

$$C_{river} = \text{tracer concentration within river water.} \quad (28.3)$$

C_{river} is distinguished in MOM from the tracer concentration associated with evaporation and precipitation. What tracer concentration should be taken for the river water? Typically, we think of rivers at their discharge point as having tracers of uniform concentration. More information about tracer profiles requires a river model, and even so we may wish to summarize the river information prior to passing it into the ocean. Assuming a single uniform value for the river tracer concentrations, and absent a river model, it is typical to assume the following river tracer concentration

$$\theta_{river} = \theta_{ocean}^{k=1} \quad (28.4)$$

$$s_{river} = 0 \quad (28.5)$$

$$T_{river} = T_{ocean}^{k=1} \quad (28.6)$$

where $k = 1$ is the top cell of the ocean column into which the river water is discharged, θ is the potential temperature, which equals the *in situ* temperature at the ocean surface, s_{river} is the zero salinity of the fresh water river, and T_{river} is the concentration of a passive tracer in the river. By assuming $\theta_{river} = \theta_{ocean}^{k=1}$, vertically distributing river water acts to warm the ocean column in regions where the ocean surface is warmer than depth. In contrast, rivers with zero salinity do not alter the ocean salt content, yet they do reduce the salinity.

Over a leap-frog tracer time step $2dtts$, a thickness $H_{river} = R * 2dtts$ of river water is to be distributed throughout the vertical ocean column:

$$H_{river} = R * 2dtts = \text{river water thickness discharged per tracer leap-frog.} \quad (28.7)$$

Along with this distribution of river water into the ocean column, we distribute the tracer content of the river into the ocean column:

$$C_{river} H_{river} = \text{river tracer content discharged per tracer leap-frog.} \quad (28.8)$$

28.3 Steps in the algorithm

We derive the algorithm by considering the conservation equations for tracer in a vertical column of ocean model grid cells. To isolate effects from river discharge, we ignore all horizontal processes. Without distributing river runoff with depth, tracer conservation is given by

$$\begin{aligned}\partial_t (V \rho C)_{k=1} &= \rho_o A R C_{river} \\ \partial_t (V \rho C)_{k>1} &= 0\end{aligned}\quad (28.9)$$

where $V = Ah$ is the volume of a grid cell, C is the tracer concentration, ρ is the *in situ* density, R is the river discharge rate, and C_{river} is the concentration of tracer in the river. As the horizontal area A is constant, it can be dropped from the discussion. Conservation of total tracer in the four-box system is manifest by

$$\partial_t \sum_{k=1}^{kr} (h \rho C) = \rho_o R C_{river}. \quad (28.10)$$

Whatever is done to redistribute river runoff with depth, this conservation law must be preserved.¹

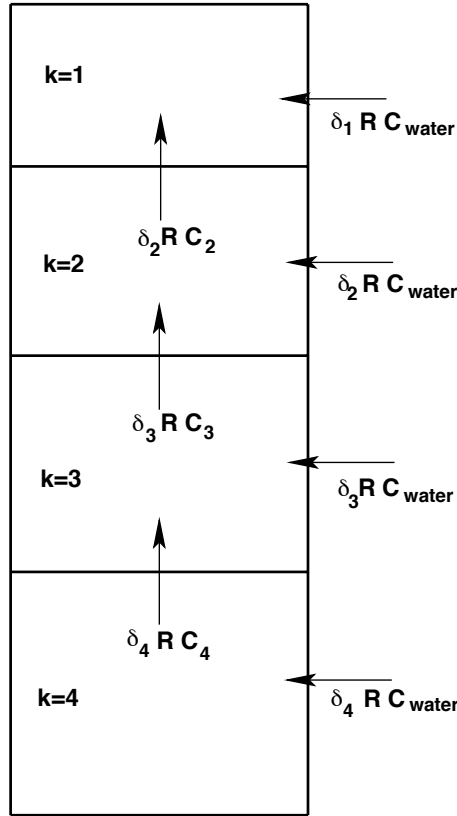


Figure 28.1: Schematic of river discharge algorithm for the case with $kr = 4$. We insert a fraction of the river water into grid cells throughout the column, with a corresponding amount leaving each cell bubbling upwards in order to conserve water mass/volume in the original implementation of MOM4.0, where grid cells cannot change volume and there are no source terms.

To derive the algorithm, we refer to Figure 28.1. Here, we prescribe that a fraction of the river water

¹The appearance of ρ_o on the right hand side, rather than ρ_{river} , is described in Chapter 2 as well as Griffies (2004).

and its tracer content is inserted into each of the cells within the column, where the fractions sum to unity

$$\sum_{k=1}^{kr} \delta_k = 1. \quad (28.11)$$

We choose the fractions according to the grid cell thickness

$$\delta_k = \frac{h_k}{\sum_{k=1}^{kr} h_k} \quad (28.12)$$

where h_k is the tracer cell thickness, and is known as $\text{dzt}_{i,j,k}$ in the MOM code.

Because the interior cell volumes remain constant in MOM4.0, the same amount of water that entered the cell via the river water must then leave. We assume that it leaves with the tracer concentration of the cell prior to the insertion of the river water. That is, by inserting some of the river water into the cell at tracer concentration C_{river} , we then displace the same amount of water but at concentration C_k . This displaced water is bubbled upwards towards the surface cell. Conservation equations for this algorithm take the form

$$\begin{aligned} \partial_t (h\rho C)_1 &= R[(\delta\rho C)_2 + \delta_1 \rho_o C_{river}] \\ \partial_t (h\rho C)_k &= R[(\delta\rho C)_{k+1} - (\delta\rho C)_k + \delta_k \rho_o C_{river}] \\ \partial_t (h\rho C)_{kr} &= R[-(\delta\rho C)_{kr} + \delta_{kr} \rho_o C_{river}] \end{aligned} \quad (28.13)$$

where the first equation is for $k = 1$, the second for $1 < k < kr$, and the third for $k = kr$. The algorithm has the appearance of upwind advection throughout the column. Hence, conservation of total tracer for the column is trivially verified. Formulation as a thickness-weighted tracer source leads to

$$\begin{aligned} \rho_o \text{dzt} \cdot \text{tracer-source}_1 &= R[(\delta\rho C)_2 + \delta_1 \rho_o C_{river}] \\ \rho_o \text{dzt} \cdot \text{tracer-source}_k &= R[(\delta\rho C)_{k+1} - (\delta\rho C)_k + \delta_k \rho_o C_{river}] \\ \rho_o \text{dzt} \cdot \text{tracer-source}_{kr} &= R[-(\delta\rho C)_{kr} + \delta_{kr} \rho_o C_{river}] \end{aligned} \quad (28.14)$$

For $kr = 1$, the method reduces to the default discharge of river into the top cell.

Chapter 29

CROSS-LAND MIXING

Contents

29.1	Introduction	397
29.2	Tracer and mass/volume compatibility	398
29.3	Tracer mixing in a Boussinesq fluid with fixed boxes	398
29.4	Mixing of mass/volume	399
29.4.1	Instantaneous and complete mixing	400
29.4.2	A finite time incomplete mixing	400
29.4.3	A finite time incomplete mixing for surface cells	401
29.5	Tracer and mass mixing	401
29.6	Formulation with multiple depths	402
29.6.1	MOM1 formulation of cross-land tracer mixing	402
29.6.2	Generalizing to free surface and non-Boussinesq	403
29.7	Suppression of B-grid null mode	404

The purpose of this chapter is to present the method used in MOM for mixing tracers and mass/volume across land separated points, such as across an unresolved Strait of Gibraltar. The material here is taken from the MOM4 Technical Guide of [Griffies et al. \(2004\)](#), with slight modifications to account for generalized vertical coordinates used in MOM.

The following MOM module is directly connected to the material in this chapter:

`ocean_param/sources/ocean_xlandmix.F90.`

29.1 Introduction

In climate modeling, it is often necessary to allow water masses that are separated by land to exchange properties. This situation arises in models when the grid mesh is too coarse to resolve narrow passage-ways that in reality provide crucial connections between water masses. For example, coarse grid spacing typically closes off the Mediterranean from the Atlantic at the Straits of Gibraltar. In this case, it is important for climate models to include the effects of salty water entering the Atlantic from the Mediterranean. Likewise, it is important for the Mediterranean to replenish its supply of water from the Atlantic to balance the net evaporation occurring over the Mediterranean region.

We describe here a method used in MOM to establish communication between bodies of water separated by land. The communication consists of mixing tracers and mass/volume between non-adjacent water columns. Momentum is not mixed. The scheme conserves total tracer content, total mass or volume

(depending on whether using the non-Boussinesq or Boussinesq versions of MOM), and maintains compatibility between the tracer and mass/volume budgets. It's only restriction is that no mixing occur between cells if their time independent thicknesses differ. This constraint is of little practical consequence.

29.2 Tracer and mass/volume compatibility

Consider two boxes with fluid masses $M^{(1)} = \rho^{(1)} V^{(1)}$ and $M^{(2)} = \rho^{(2)} V^{(2)}$ and tracer concentrations (tracer mass per mass of fluid) $T^{(1)}$ and $T^{(2)}$ (for a Boussinesq fluid, the density is set to the constant Boussinesq density ρ_0). A mixing process that conserves total tracer mass and total fluid mass must satisfy

$$\partial_t (T^{(1)} \rho^{(1)} V^{(1)} + T^{(2)} \rho^{(2)} V^{(2)}) = 0 \quad (29.1)$$

$$\partial_t (\rho^{(1)} V^{(1)} + \rho^{(2)} V^{(2)}) = 0. \quad (29.2)$$

Notably, mass conservation can be considered a special case of total tracer conservation when the tracer concentration is uniform and constant: $T \equiv 1$. This result provides an important *compatibility* constraint between the discrete tracer and mass/volume budgets. For constant volume boxes with a Boussinesq fluid, such as considered in rigid lid models, compatibility is trivial. For boxes which change in time, such as the top cells in MOM4.0 free surface or any box in the generalized vertical coordinates of later versions, then compatibility provides an important constraint on the methods used to discretize the budgets for mass/volume and tracer. The remainder of this chapter incorporates these ideas into the proposed cross-land mixing scheme.

29.3 Tracer mixing in a Boussinesq fluid with fixed boxes

To start in our formulation of cross-land mixing, let us consider mixing of two volumes of Boussinesq fluid, where the separate volumes remain constant in time

$$\partial_t V^{(1)} = \partial_t V^{(2)} = 0. \quad (29.3)$$

An example is the mixing between two constant volume grid cells. If the mixing takes place instantaneously and between the full contents of both boxes, as in convective adjustment, then the final tracer concentration in both boxes is given by

$$T_{final} = \frac{T^{(1)} V^{(1)} + T^{(2)} V^{(2)}}{V^{(1)} + V^{(2)}}. \quad (29.4)$$

It is assumed in convective mixing that the volumes of the two boxes remains unchanged. The picture is of an equal volume of water rapidly mixing from one box to the other, without any net transport between the boxes.

Instead of instantaneous and complete convective mixing, consider mixing of the two boxes at a volume rate U . That is, U represents an equal volume per time of water mixing between the boxes, with no net transport. As shown in Figure 29.1, U is chosen based on the observed amount of water exchanged through the passageway. Just as for convective adjustment, the volumes of the two boxes remains fixed. But the tracer concentrations now have a time tendency. One form for this tendency relevant for constant volume cells is given by

$$\partial_t (V^{(1)} T^{(1)}) = U (T^{(2)} - T^{(1)}) \quad (29.5)$$

$$\partial_t (V^{(2)} T^{(2)}) = U (T^{(1)} - T^{(2)}). \quad (29.6)$$

Since the volumes are constant, we can write these budgets in the form

$$\partial_t T^{(1)} = \frac{U}{V^{(1)}} (T^{(2)} - T^{(1)}) \quad (29.7)$$

$$\partial_t T^{(2)} = \frac{U}{V^{(2)}} (T^{(1)} - T^{(2)}), \quad (29.8)$$

This is the form of cross-land tracer mixing used in the rigid lid full cell MOM1.

In the real world, transport is often comprised of stacked flows where deep water flows one way and shallow water oppositely (e.g., see Figure 29.1). Hence, a more refined form of cross-land mixing may consist of upwind advective fluxes acting between non-local points in the model, where the advective velocity is specified based on observations. Such sophistication, however, is not implemented in MOM. Indeed, it is arguable that one may not wish to have more details than provided by the simpler form above, since more details also further constrain the solution.

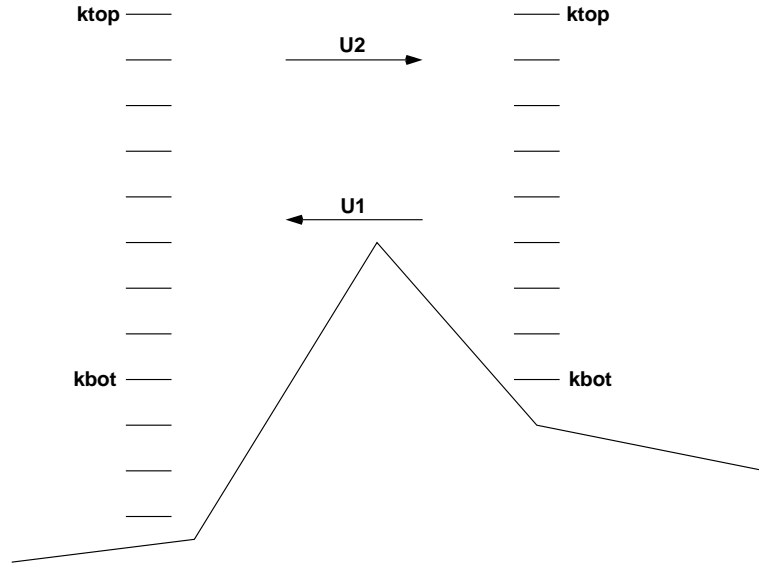


Figure 29.1: Schematic of cross-land mixing. The model's grid mesh is assumed too coarse to explicitly represent the lateral exchange of water masses. For this schematic, we consider an observed sub-grid scale transport U_1 moving in one direction, and U_2 in another. To represent the mixing effects on tracers by these transports, we suggest taking the exchange rate U in MOM's cross-land mixing to be the average of the transports $U = (U_1 + U_2)/2$. Cross-land mixing occurs between the user-specified depth levels $k = k_{top}$ and $k = k_{bot}$. If $k_{top} = 1$, then cross-land mixing of volume in the top cell must be considered, in addition to tracer transport, in order to maintain compatibility between volume and tracer budgets.

29.4 Mixing of mass/volume

In a model with a coarse mesh, the Mediterranean is typically land-locked. Hence, the net evaporation experienced over the Mediterranean region will cause the simulated ocean volume in this region to decrease without bound. In a model resolving the Straits of Gibraltar, there is a transfer of mass across the Strait from the Atlantic. This mass transfer creates a change in the height of the free surface.

Our goal is to have a parameterized mass transfer associated just with a difference in the free surface height. That is, if the densities are different yet the free surface heights are equal, then there is no mixing. By transferring masses of water, we must also recognize that the water contains tracer. Hence, mass and tracer mixing must maintain the compatibility mentioned in Section 29.2. In this section, however, we only introduce a basic form for mass transfer. Full compatibility with tracer transfer is achieved in Section 29.5

29.4.1 Instantaneous and complete mixing

To start by considering what form for mixing is appropriate, consider a convective analog whereby a complete mixing of masses $\rho^{(1)} A^{(1)} h^{(1)}$ and $\rho^{(2)} A^{(2)} h^{(2)}$ leaves the final mass per area in both cells given by

$$(\rho h)_{final} = \frac{\rho^{(1)} A^{(1)} h^{(1)} + \rho^{(2)} A^{(2)} h^{(2)}}{A^{(1)} + A^{(2)}}, \quad (29.9)$$

where $A^{(1)}$ and $A^{(2)}$ are the temporally constant horizontal areas of the two grid cells and $h^{(1)}$ and $h^{(2)}$ are their generally time dependent thicknesses.

There are two problems with this mixing. First, it is too rapid and too complete. We prefer a method that allows for some control in the rate of mixing. Second, it changes the mass within a grid cell in cases where the initial masses per area are equal yet the constant horizontal areas of the cells differ.

29.4.2 A finite time incomplete mixing

A finite time and incomplete mixing is analogous to that taken for the tracers in Section 29.3. Here, we consider the time tendencies for the mass per area within a cell

$$\partial_t(\rho^{(1)} h^{(1)}) = \gamma^{(1)} (\rho^{(2)} h^{(2)} - \rho^{(1)} h^{(1)}) \quad (29.10)$$

$$\partial_t(\rho^{(2)} h^{(2)}) = \gamma^{(2)} (\rho^{(1)} h^{(1)} - \rho^{(2)} h^{(2)}), \quad (29.11)$$

where $\gamma^{(1)}$ and $\gamma^{(2)}$ are inverse damping times. This proposed mixing results in a transfer of mass only when the mass per area within the two boxes differs. The total mass of the two-box system is conserved if the following constraint is satisfied

$$\partial_t[(\rho h A)^{(1)} + (\rho h A)^{(2)}] = (A^{(1)} \gamma^{(1)} - A^{(2)} \gamma^{(2)}) (\rho^{(2)} h^{(2)} - \rho^{(1)} h^{(1)}) = 0. \quad (29.12)$$

This relation places a constraint on the inverse damping times $\gamma^{(1)}$ and $\gamma^{(2)}$

$$A^{(1)} \gamma^{(1)} = A^{(2)} \gamma^{(2)} \quad (29.13)$$

which is easily satisfied.

The problem with the mixing prescribed by equations (29.10) and (29.11) is that mixing will ensue in the following two undesirable cases. First, if the densities of the two cells are initially the same $\rho^{(1)} = \rho^{(2)} = \rho$, yet the cells have different thicknesses, then density change is driven solely by the difference in cell thicknesses

$$h^{(1)} \partial_t \rho^{(1)} = \rho \gamma^{(1)} (h^{(2)} - h^{(1)}) \quad (29.14)$$

$$h^{(2)} \partial_t \rho^{(2)} = \rho \gamma^{(2)} (h^{(1)} - h^{(2)}). \quad (29.15)$$

Such is acceptable in our scheme only for the surface ocean grid cell.

Another problem with the mixing prescribed by equations (29.10) and (29.11) is seen by considering the situation whereby two top model grid cells have initially equal thicknesses $h^{(1)} = h^{(2)} = h$ yet different densities. The model grid cell thickness will evolve because of the difference in densities

$$\rho^{(1)} \partial_t h^{(1)} = h \gamma^{(1)} (\rho^{(2)} - \rho^{(1)}) \quad (29.16)$$

$$\rho^{(2)} \partial_t h^{(2)} = h \gamma^{(2)} (\rho^{(1)} - \rho^{(2)}). \quad (29.17)$$

However, as stated at the beginning of this section, we aim to prescribe a mixing process that occurs only when the tracer concentration and/or free surface heights differ. Therefore, we must consider an alternative to equations (29.10) and (29.11)

29.4.3 A finite time incomplete mixing for surface cells

We consider the following prescription for the surface grid cells, in which mixing occurs only when the surface heights differ

$$\partial_t(\rho^{(1)} h^{(1)}) = \gamma^{(1)} \bar{\rho} (h^{(2)} - h^{(1)}) \quad (29.18)$$

$$\partial_t(\rho^{(2)} h^{(2)}) = \gamma^{(2)} \bar{\rho} (h^{(1)} - h^{(2)}). \quad (29.19)$$

When considered over interior model grid cells, then we prescribe no mass transfer. The density factor $\bar{\rho}$ can be given by anything convenient, such as

$$\bar{\rho} = \frac{\rho^{(1)} + \rho^{(2)}}{2}, \quad (29.20)$$

or the even simpler prescription

$$\bar{\rho} = \rho_o. \quad (29.21)$$

29.5 Tracer and mass mixing

The general case of mixing tracers and mass is now considered. The following are the aims of the formulation.

- Total fluid mass in the two boxes is conserved.
- Total tracer mass in the two boxes is conserved.
- In the rigid lid Boussinesq full cell case, the tracer tendency reduces to equations (29.7) and (29.8) used in MOM1.
- Mass is exchanged only between top grid cells, in which case if the tracer concentration in the two boxes is the same yet the mass differs, then mixing of mass will leave the tracer concentrations unchanged.
- The time tendency for the mass exchange in the top cells is proportional to the difference in surface height η_{at} between the cells, rather than the generally smaller difference between the cell thickness dz .

Mixing that satisfies these constraints is given by the following for the surface grid cells with $k = 1$

$$\partial_t(\rho^{(1)} h^{(1)} T^{(1)}) = \left(\frac{2 U \rho_o}{A^{(1)} (H^{(1)} + H^{(2)})} \right) (h^{(2)} T^{(2)} - h^{(1)} T^{(1)}) \quad (29.22)$$

$$\partial_t(\rho^{(2)} h^{(2)} T^{(2)}) = \left(\frac{2 U \rho_o}{A^{(2)} (H^{(1)} + H^{(2)})} \right) (h^{(1)} T^{(1)} - h^{(2)} T^{(2)}) \quad (29.23)$$

$$\partial_t(\rho^{(1)} h^{(1)}) = \left(\frac{2 U \rho_o}{A^{(1)} (H^{(1)} + H^{(2)})} \right) (h^{(2)} - h^{(1)}) \quad (29.24)$$

$$\partial_t(\rho^{(2)} h^{(2)}) = \left(\frac{2 U \rho_o}{A^{(2)} (H^{(1)} + H^{(2)})} \right) (h^{(1)} - h^{(2)}) \quad (29.25)$$

Likewise, for interior cells with $k > 1$, we prescribe

$$\partial_t(\rho^{(1)} h^{(1)} T^{(1)}) = \left(\frac{2 U \overline{\rho_o dz t}}{A^{(2)} (H^{(1)} + H^{(2)})} \right) (T^{(2)} - T^{(1)}) \quad (29.26)$$

$$\partial_t(\rho^{(2)} h^{(2)} T^{(2)}) = \left(\frac{2 U \overline{\rho_o dz t}}{A^{(2)} (H^{(1)} + H^{(2)})} \right) (T^{(1)} - T^{(2)}) \quad (29.27)$$

$$\partial_t(\rho^{(1)} h^{(1)}) = 0 \quad (29.28)$$

$$\partial_t(\rho^{(2)} h^{(2)}) = 0. \quad (29.29)$$

In these equations, H is the depth of a column with a resting ocean surface. For the $k > 1$ equations, $\rho_0 \Delta z$ is the averaged thickness weighted density for the two cells. For the $k = 1$ equations,

$$h(k = 1) = \text{Grd} \Delta z t_{k=1} + \text{eta}_t \quad (29.30)$$

is the thickness of the top cell for the case of a geopotential vertical coordinate. The general Thickness $\Delta z t_{i,j,k=1}$ thickness varies much less rapidly in the horizontal when employing z^* or p^* as the vertical coordinate. In order to employ similar mixing rates for the geopotential model as for the general vertical coordinate models, we prefer the more restricted definition (29.30) of thickness based on the geopotential model. The mass per area equations (29.24) and (29.25) result from the tracer equations (29.22) and (29.23) upon setting the tracer concentrations to a constant, as required for compatible budgets.

29.6 Formulation with multiple depths

We now consider the case where there are multiple boxes in the vertical. We restrict attention to situations where mixing occurs between boxes at the same vertical level, as shown in Figure 29.1.

29.6.1 MOM1 formulation of cross-land tracer mixing

In MOM1, the vertical cells all have time independent thicknesses (i.e., rigid lid geopotential coordinate model), and the fluid is Boussinesq. It is useful to start with this case prior to considering the more general case.

In the full cell rigid lid case, we follow the approach given by equations (29.7) and (29.8), where the relevant volume now becomes that for the respective column. The volumes for the two columns $lx = 1, 2$ are given by

$$V^{(lx)} = A^{(lx)} \sum_{k=kbot}^{ktop} \Delta z t_k = A^{(lx)} H^{(lx)}, \quad (29.31)$$

where

$$A^{(lx)} = \Delta x t_{i,j}^{(lx)} \Delta y t_{i,j}^{(lx)} \quad (29.32)$$

are the generally different horizontal cross-sectional areas of the tracer cells in the two columns, and $H^{(1)} = H^{(2)}$ is the vertical thickness of the two columns. The top and bottom k -levels for the columns are set by $k = ktop$ and $k = kbot$. As mentioned earlier, the formulation here allows for mixing only between boxes that live on the same k -level, so $k = ktop$ and $k = kbot$ are the same for both columns $lx = 1, 2$.

Use of these volumes in equations (29.7) and (29.8) leads to the tracer time tendencies for a particular k -level

$$\partial_t T_k^{(1)} = B^{(1)} (T_k^{(2)} - T_k^{(1)}) \quad (29.33)$$

$$\partial_t T_k^{(2)} = B^{(2)} (T_k^{(1)} - T_k^{(2)}), \quad (29.34)$$

where

$$B^{(lx)} = \frac{U}{V^{(lx)}} \quad (29.35)$$

represents the rate ($B^{(lx)}$ has units of inverse time) at which the two columns participate in the mixing. Conservation of total tracer is maintained between two horizontally adjacent boxes within the two columns. We see such conservation via multiplying the above tendencies by the respective time independent volumes of the two cells, and adding

$$\partial_t (V_k^{(1)} T_k^{(1)} + V_k^{(2)} T_k^{(2)}) = (T_k^{(2)} - T_k^{(1)}) (A^{(1)} B^{(1)} h_k^{(1)} - A^{(2)} B^{(2)} h_k^{(2)}) = 0, \quad (29.36)$$

where

$$\begin{aligned} A^{(1)} B^{(1)} h_k^{(1)} &= U (h_k^{(1)} / H^{(1)}) \\ &= U (h_k^{(2)} / H^{(2)}) \\ &= A^{(2)} B^{(2)} h_k^{(2)} \end{aligned} \quad (29.37)$$

was used.

29.6.2 Generalizing to free surface and non-Boussinesq

We now generalize to the case of time varying grid cells with generalized vertical coordinates. Based on the considerations of Section 29.5 and the form used in MOM1, we write for the general case for a surface grid cell with $k = 1$

$$\partial_t(\rho_k^{(1)} h_k^{(1)} T_k^{(1)}) = \left(\frac{2 U \rho_o}{A^{(1)}(H^{(1)} + H^{(2)})} \right) (h_k^{(2)} T_k^{(2)} - h_k^{(1)} T_k^{(1)}) \quad (29.38)$$

$$\partial_t(\rho_k^{(2)} h_k^{(2)} T_k^{(2)}) = \left(\frac{2 U \rho_o}{A^{(2)}(H^{(1)} + H^{(2)})} \right) (h_k^{(1)} T_k^{(1)} - h_k^{(2)} T_k^{(2)}), \quad (29.39)$$

where again $H^{(1)}$ and $H^{(2)}$ are the generally different static resting depths of the two columns, and

$$h(k = 1) = \text{Grd\%dz}t_{k=1} + \text{eta.t} \quad (29.40)$$

according to our prescription given by equation (29.30). Setting the tracers to uniform constants leads to the transfer of mass per area between two surface cells

$$\partial_t(\rho_k^{(1)} h_k^{(1)}) = \left(\frac{2 U \rho_o}{A^{(1)}(H^{(1)} + H^{(2)})} \right) (h_k^{(2)} - h_k^{(1)}) \quad (29.41)$$

$$\partial_t(\rho_k^{(2)} h_k^{(2)}) = \left(\frac{2 U \rho_o}{A^{(2)}(H^{(1)} + H^{(2)})} \right) (h_k^{(1)} - h_k^{(2)}), \quad (29.42)$$

For interior cells with $k > 1$, we prescribe

$$\partial_t(\rho_k^{(1)} h_k^{(1)} T_k^{(1)}) = \left(\frac{2 U \overline{\text{rho.dzt}_k}}{A^{(1)}(H^{(1)} + H^{(2)})} \right) (T_k^{(2)} - T_k^{(1)}) \quad (29.43)$$

$$\partial_t(\rho_k^{(2)} h_k^{(2)} T_k^{(2)}) = \left(\frac{2 U \overline{\text{rho.dzt}_k}}{A^{(2)}(H^{(1)} + H^{(2)})} \right) (T_k^{(1)} - T_k^{(2)}), \quad (29.44)$$

where

$$\overline{\text{rho.dzt}_k} = \frac{\text{rho.dzt}_k^{(1)} + \text{rho.dzt}_k^{(2)}}{2} \quad (29.45)$$

is the average thickness weighted density of the adjacent cells, and

$$h = \text{Thickness\%dz}t_{i,j,k} \quad (29.46)$$

is the general thickness of the tracer cell. By inspection, for each k -level this formulation conserves total tracer mass and total fluid mass (recall Section 29.4). Setting the tracers to uniform constants leads to a zero transfer of mass per area between two interior cells.

These budgets can be written in a form familiar from other damping processes, in which for $k = 1$ we have

$$\partial_t(\rho^{(1)} h^{(1)} T^{(1)}) = \gamma^{(1)} \rho_o (h^{(2)} T^{(2)} - h^{(1)} T^{(1)}) \quad (29.47)$$

$$\partial_t(\rho^{(2)} h^{(2)} T^{(2)}) = \gamma^{(2)} \rho_o (h^{(1)} T^{(1)} - h^{(2)} T^{(2)}) \quad (29.48)$$

$$\partial_t(\rho^{(1)} h^{(1)}) = \gamma^{(1)} \rho_o (h^{(2)} - h^{(1)}) \quad (29.49)$$

$$\partial_t(\rho^{(2)} h^{(2)}) = \gamma^{(2)} \rho_o (h^{(1)} - h^{(2)}) \quad (29.50)$$

where the depth label k was omitted for brevity, and

$$\gamma^{(1)} = \frac{2 U}{A^{(1)}(H^{(1)} + H^{(2)})} \quad (29.51)$$

$$\gamma^{(2)} = \frac{2 U}{A^{(2)}(H^{(1)} + H^{(2)})} \quad (29.52)$$

defines the damping coefficients. For interior cells, only tracer concentration is mixed, in which case

$$\partial_t(\rho^{(1)} h^{(1)} T^{(1)}) = \gamma^{(1)} \overline{\rho \cdot dz \cdot t} (T^{(2)} - T^{(1)}) \quad (29.53)$$

$$\partial_t(\rho^{(2)} h^{(2)} T^{(2)}) = \gamma^{(2)} \overline{\rho \cdot dz \cdot t} (T^{(1)} - T^{(2)}), \quad (29.54)$$

with $\gamma^{(1)}$ and $\gamma^{(2)}$ as for the surface cell.

The damping coefficients (29.51) and (29.52) are generally time dependent for cases with mixing in the top cell and where the free surface height is included when computing the column thicknesses $H^{(1)}$ and $H^{(2)}$. One may alternatively be motivated to keep the damping coefficients constant in time by setting $H^{(1)}$ and $H^{(2)}$ to be the time independent depth of the respective columns. This choice is appropriate when using cross-land mixing between columns in shallow regions where the free surface height is some nontrivial fraction of the full column depth. MOM generally sets the thicknesses to the time independent depths.

To get a sense for the strength of the mixing, consider the case of a one-degree horizontal grid mesh where the upper thousand meters of the water column is mixed across Gibraltar with $U = 1.75 \times 10^6 \text{ m}^3 \text{ s}^{-1}$, which is a reasonable value. With $H^{(1)} = H^{(2)} \approx 1000 \text{ m}$ we have

$$V^{(1)} \approx V^{(2)} \approx 1.2 \times 10^{13} \text{ m}^3, \quad (29.55)$$

and to the damping coefficient

$$\gamma^{(1)} \approx \gamma^{(2)} = \frac{U}{V} \approx 1.5 \times 10^{-7} \text{ s}^{-1} \approx 77 \text{ days}^{-1}. \quad (29.56)$$

Just as for any other form of mixing, if the damping coefficients are too large, then it is possible for there to be numerical instabilities. MOM provides a check so that no more than one-half of a particular grid cell is mixed per model time step.

29.7 Suppression of B-grid null mode

When mixing the free surface height across an unresolved strait, it has been found essential to mix between two pairs of adjacent columns in order to suppress the checkerboard null mode present when using the B-grid version of MOM (see Chapter 26 of Griffies et al. (2004)). For the Mediterranean example, this means choosing any two adjacent points on each side of Gibraltar and setting the volume transport for each column to $U = (1/2) 1.75 \times 10^6 \text{ m}^3 \text{ s}^{-1}$.

CROSS-LAND INSERTION

Contents

30.1	Introduction	405
30.2	Algorithm details	406
30.3	An example: insertion to three cells in MOM4.0	407
30.4	An example: insertion to just the top cell in MOM4.0	409
30.5	Updates for generalized level coordinates	410

The purpose of this chapter is to present the method used in MOM for adjusting volume/mass and tracers across land separated points, such as across an unresolved Strait of Gibraltar, in a way that aims to reduce the difference between the surface heights. The method is a hybrid between the river discharge process in Chapter 28 and the cross-land mixing process of Chapter 29.

Both the cross-land insertion and river discharge schemes were originally implemented in MOM4.0, in which there is no interior volume sources, and the interior grid cells maintain constant volume. It is only the top model grid cell can modify its volume. This constraint leads to an extra step to conserve volume when moving volume from a surface cell to an interior cell. The result of this constraint is illustrated in Figure 30.1.

The following MOM module is directly connected to the material in this chapter:

ocean_param/sources/ocean_xlandinsert.F90.

30.1 Introduction

In Nature, marginal seas such as the Mediterranean have a nontrivial connection to the World Ocean. Hence, depletion of water in marginal seas by more than a few meters, or over-flowing water, is unlikely in a century scale climate change scenario, even in cases where the net moisture budget over the regions surrounding the marginal sea experience nontrivial changes. That is, the sea levels in marginal seas are roughly in equilibrium with that in the World Ocean.

In many coarse climate models, the connection between marginal seas and the World Ocean is unresolved. The cross-land mixing process discussed in Chapter 29 specifies an *a priori* rate for mixing water masses on the opposite sides of an unresolved straight. This method works fine for simulations where surface forcing is reasonably steady in time. Hence, the sea level in the marginal sea cannot deviate drastically from that in the World Ocean to which it mixes.

Problems arise in a climate change simulation where in general the net moisture balance over regions may experience a trend. Hence, a constant cross-land mixing rate may prove insufficient to ensure that the marginal sea grid cells do not either dry out or become overly full. As MOM cannot numerically run

with a dry top grid cell, it is necessary for numerical purposes to ensure that these cells remain wet. We are therefore motivated to provide a mechanism for the coarse numerical model to retain water levels within enclosed seas that are roughly in equilibrium with the corresponding levels in the adjacent World Ocean.

The cross-land insertion process described in this chapter is a hybrid between the river discharge process in Chapter 28 and the cross-land mixing process of Chapter 29. Cross-land insertion computes the difference in surface height between two grid columns separated by an unresolved straight. The surface height difference defines a volume of seawater that is taken from the thicker cell and inserted into the column with the thinner surface height. The insertion occurs over a depth set by the user, and is realized via an upwind advection scheme just as in the river discharge method. The time scale for the insertion is set according to a user specified delay time.

For cases in practice at GFDL, insertion occurs just from top cell to top cell. In this case, the cross-land insertion process has the same mathematical form as the cross-land mixing process. Importantly, the rate of mixing between the two cells with the cross-land insertion process is proportional to the surface height difference. In contrast, for the cross-land mixing process, the rate of mixing is constant in time.

30.2 Algorithm details

Consider two columns of a Boussinesq fluid. Let the horizontal cross-sectional area of the two columns be A_1 and A_2 , respectively, and let the vertical thickness of their top cells be h_1 and h_2 . The total volume of seawater in the two top cells is

$$V = A_1 h_1 + A_2 h_2. \quad (30.1)$$

The cross-land insertion process must keep this volume constant. In particular, if the two top cells are allowed to freely exchange fluid and reach an equilibrium, then the top cells would each have the same final thickness h given by

$$h = \frac{V}{A_1 + A_2}. \quad (30.2)$$

This result follows from volume conservation, which requires

$$A_1 h_1 + A_2 h_2 = A_1 h + A_2 h. \quad (30.3)$$

We prescribe a mechanism to exchange fluid over a finite time so that the exchange nudges the top cells toward the same thickness h .

To illustrate the algorithm, assume at the initial time that the free surface elevations satisfy $\eta_1 > \eta_2$. Upon full exchange (after an infinite time), the amount of water transferred from the thicker column to the thinner column is

$$(h_1 - h)A_1 = (h - h_2)A_2. \quad (30.4)$$

This exchange leads to the final volume in the two cells

$$h_1 A_1 - (h_1 - h)A_1 = h A_1 \quad (30.5)$$

$$h_2 A_2 + (h - h_2)A_2 = h A_2. \quad (30.6)$$

Now assume that water is removed just from the surface cell from the thicker column and inserted into the adjacent thinner column over its upper few cells. Figure 30.1 illustrates the algorithm.

If the transfer is assumed to occur at a finite rate set by a time constant τ , then one can identify an effective rate of volume that leaves the thicker cell

$$R^{thick} A_1 = A_1 (h_1 - h)/\tau. \quad (30.7)$$

The corresponding rate of volume that enters the thinner cell is

$$R^{thin} A_2 = A_2 (h - h_2)/\tau. \quad (30.8)$$

The volumes rates are the same, but the area normalized rates R^{thick} and R^{thin} differ when the horizontal areas are different. Importantly, the rates of transfer are functions of the difference in surface heights.

This characteristic of the cross-land insertion process is fundamentally distinct from the cross-land mixing process of Chapter 29. It is this property that makes the cross-land insertion process of use for climate change situations.

The area normalized rates R^{thick} and R^{thin} are analogous to the river discharge rate considered in Chapter 28. Consequently, it is anticipated that their magnitudes correspond roughly to that of river runoff, whose values are on the order of a vertical velocity. This observation prescribes a time scale τ on the order of a day. A faster time scale may lead to numerical instability since it will force an overly strong tendency in the surface height equation. As in Chapter 28, fluid from the thicker column is inserted into the adjacent smaller column. Tracer conservation requires that we recognize that transferred seawater has a tracer concentration of the surface cell in the thicker column.

Ignoring all other processes, the free surface and tracer equations for the two columns take the following form. For the thicker column, just the top grid cell is affected

$$\partial_t \eta^{thick} = -R^{thick} \quad (30.9)$$

$$\partial_t (\rho h C)_1^{thick} = -R^{thick} C_1^{thick} \rho_o \quad (30.10)$$

with $\rho \rightarrow \rho_o$ for the Boussinesq case. Note that for the tracer concentration in the Boussinesq case, we have

$$\partial_t C_1^{thick} = C_1^{thick} (-R^{thick} + R^{thick}) = 0. \quad (30.11)$$

This result holds so long as C^{thick} is evaluated to the τ time level when computing the tendency term in the tracer equation. It is the expected result, since by transferring water with a concentration C^{thick} , we should have left behind water of the same concentration (but less total tracer mass).

For the thinner column, all cells within the specified range $1 \leq k \leq kr$ are affected

$$\begin{aligned} \partial_t \eta^{thin} &= R^{thin} \\ \partial_t (h \rho C)_1^{thin} &= R^{thin} [(\delta \rho C)_2^{thin} + \delta_1 \rho_o C_1^{thick}] \\ \partial_t (h \rho C)_k^{thin} &= R^{thin} [(\delta \rho C)_{k+1}^{thin} - (\delta \rho C)_k^{thin} + \delta_k \rho_o C_1^{thick}] \\ \partial_t (h \rho C)_{kr}^{thin} &= R^{thin} [-(\delta \rho C)_{kr}^{thin} + \delta_{kr} \rho_o C_1^{thick}] \end{aligned} \quad (30.12)$$

30.3 An example: insertion to three cells in MOM4.0

It is useful to work through an explicit example to get a sense for how the scheme is expected to work. For this purpose, consider the initial configuration shown in Figure 30.2, in which the thick column has an excess surface height H . For simplicity, we take this excess height to also equal to the thickness of the cells with a resting ocean surface. Assume that the horizontal cross-sectional area of the cells is all the same. Hence, the rate of water transfer, R , is the same for the two columns, and is given by

$$R = H/\tau \quad (30.13)$$

with τ the damping time. Three cells in the vertical into which to transfer the water yields the fraction

$$\delta = 1/3. \quad (30.14)$$

The conservation equations for the top cell in the thick column are

$$\partial_t \eta^{thick} = -R \quad (30.15)$$

$$\partial_t (h C)_1^{thick} = -R C_1^{thick} \quad (30.16)$$

where h is the time dependent thickness of the thick upper cell, and we assumed a Boussinesq fluid for simplicity. Note that the concentration on the right hand side of the tracer equation is evaluated at the

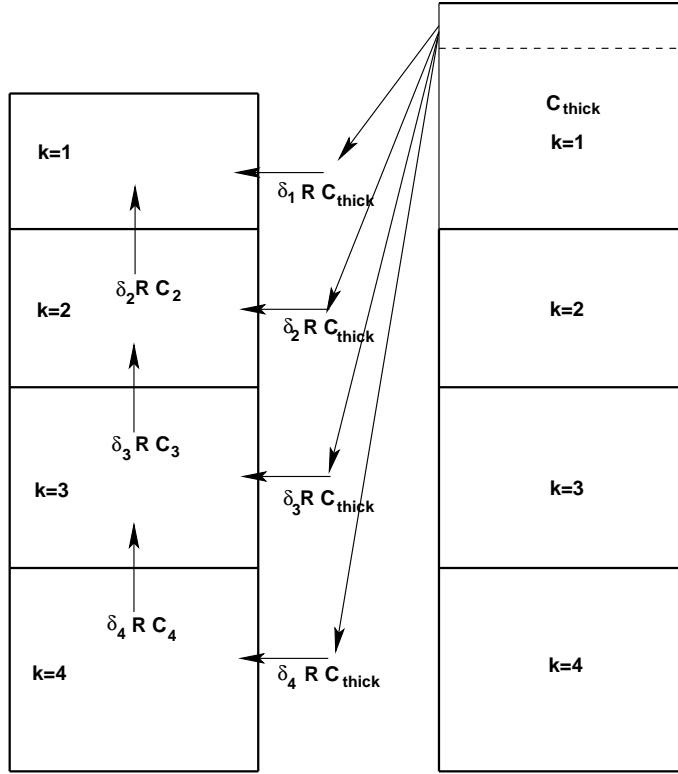


Figure 30.1: Schematic of the cross-land insertion algorithm. An amount of seawater is extracted from the surface cell in a column with greater surface height. This water is inserted into the adjacent column with smaller surface height. As implemented in MOM4.0, Insertion to cells with $k > 1$ requires a bubbling upwards of water in order to conserve volume, since only the top cell can change its volume and MOM4.0 has no interior volume sources or sinks. This figure is directly analogous to Figure 28.1 which illustrates the method used to discharge river water at depth. Note that insertion from the top cell to the top cell is often used in practice. An example is provided in Section 30.4.

present time step, τ . As noted in Section 30.2, the tracer concentration remains unchanged in the top thick cell, since

$$\begin{aligned} h \partial_t C_1^{thick} &= -R C_1^{thick} - C_1^{thick} \partial_t \eta^{thick} \\ &= C_1^{thick} (-R + R) \\ &= 0. \end{aligned} \quad (30.17)$$

Conservation equations for the three cells in the thin column are

$$\partial_t \eta^{thin} = R \quad (30.18)$$

$$\partial_t (h C)_1^{thin} = R \delta (C_2^{thin} + C_1^{thick}) \quad (30.19)$$

$$\partial_t (H C)_2^{thin} = R \delta (C_3^{thin} - C_2^{thin} + C_1^{thick}) \quad (30.20)$$

$$\partial_t (H C)_3^{thin} = R \delta (-C_3^{thin} + C_1^{thick}). \quad (30.21)$$

Using $\partial_t \eta^{thin} = R$ in the first tracer equation leads to

$$h \partial_t C_1^{thin} = R (\delta C_2^{thin} + \delta C_1^{thick} - C_1^{thin}). \quad (30.22)$$

The deep cells have time independent thickness $h = H$. For the surface cell, its thickness is assumed to initially have the same value, $h^{thin}(\tau) = H$. Bringing these results together leads to the three tracer

concentration equations

$$\partial_t C_1^{thin} = \tau^{-1} (\delta C_2^{thin} + \delta C_1^{thick} - C_1^{thin}) \quad (30.23)$$

$$\partial_t C_2^{thin} = \tau^{-1} \delta (C_3^{thin} - C_2^{thin} + C_1^{thick}) \quad (30.24)$$

$$\partial_t C_3^{thin} = \tau^{-1} \delta (-C_3^{thin} + C_1^{thick}). \quad (30.25)$$

Now consider the initial tracer concentration values of

$$C_k^{thin}(\tau) = 24 \quad (30.26)$$

$$C_1^{thick}(\tau) = 36, \quad (30.27)$$

where the units are, for example, psu if the tracer represents salt. The first time step with these initial conditions leads to the following budgets for tracer concentration. The top cell has the budget

$$\begin{aligned} \tau \partial_t C_1^{thin} &= (24\delta + 36\delta - 24) \\ &= -4. \end{aligned} \quad (30.28)$$

The second cell's budget is

$$\begin{aligned} \tau \partial_t C_2^{thin} &= \delta(24 - 24 + 36) \\ &= 12. \end{aligned} \quad (30.29)$$

The third cell's budget is

$$\begin{aligned} \tau \partial_t C_3^{thin} &= \delta(-24 + 36) \\ &= 4. \end{aligned} \quad (30.30)$$

Notice how the surface cell in the thin column has a negative tendency for its concentration, whereas the surface thick cell has its concentration remain unchanged. The surface cell in the thin column has its concentration reduced since its surface height increases by the amount of fluid that is moved from the thick cell. This result may appear counter-intuitive since we are moving water with high tracer concentration from the thick column to the thin column. Additionally, only a fraction of the tracer that left the thick cell is bubbled up to the top of the thin column. Hence, the negative tendency for the top cell in the thin column can change to a positive tendency if the ratio δ becomes large enough. That is, when we insert into fewer cells. The example in Section 30.4 considers this point further.

30.4 An example: insertion to just the top cell in MOM4.0

Consider now the case where the transfer is from the top cell of the thick column to *just* the top cell of the thin column. Here, the surface height tendencies are the same, and the tracer budget for the thick cell is the same. For the thin column, only the top cell is affected. The budget equations are thus given by

$$\partial_t \eta^{thick} = -R \quad (30.31)$$

$$\partial_t (hC)_1^{thick} = -R C_1^{thick} \quad (30.32)$$

$$\partial_t \eta^{thin} = R \quad (30.33)$$

$$\partial_t (hC)_1^{thin} = R C_1^{thick}. \quad (30.34)$$

Converting the tracer equations to tracer concentration equations leads to

$$\partial_t C_1^{thick} = 0 \quad (30.35)$$

$$\partial_t C_1^{thin} = (R/h)(C_1^{thick} - C_1^{thin}). \quad (30.36)$$

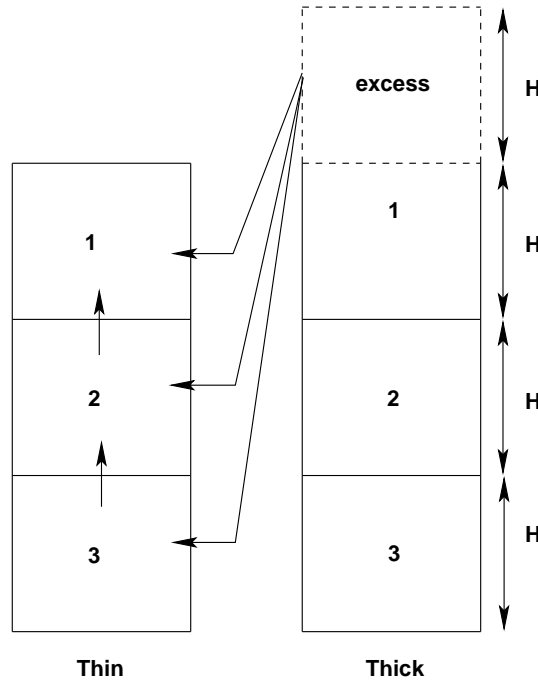


Figure 30.2: Example for how the cross-land insertion method works. In the thicker column, we assume an initial excess of water equal to the amount of water in one of the equally sized interior cells. This excess is transported into the thinner column.

With $C_1^{thick} = 36$ and $C_1^{thin} = 24$, the tendency for C_1^{thin} is $12/\tau$, whereas it was $-4/\tau$ in the 3-cell insertion example discussed in Section 30.3.

In general, recall that the rate of transfer, R , is a function of the difference in surface heights. For larger surface height differences, the rate increases. This example provides a clear illustration of the difference between the cross-land insertion process and the cross-land mixing of Chapter 29. Namely, when mixing from top cell to top cell, the cross-land mixing process does so at a rate that is fixed. For the cross-land insertion process, this rate changes as the surface height between the two columns changes.

30.5 Updates for generalized level coordinates

The generalized level coordinate model code in MOM allows for two modifications relevant for the cross land insertion scheme. First, each grid cell can have a volume or mass source, thus allowing for water to be arbitrarily moved from cell to cell, without requiring an *a priori* means to re-plumb the flow as in Figure 30.1 to conserve mass/volume. Second, each grid cell has a generally time dependent thickness, even in the interior.

We have not generalized the formulation of cross-land insertion in a manner to exploit the added flexibility of MOM. Instead, we have merely updated the algorithm to facilitate use of generalized level coordinates and retain the basics of the MOM4.0 implementation. Given that the scheme is *ad hoc* anyhow, we decided that there was little motivation to present a more rigorous or general formulation.

Chapter 31

THE B-GRID COMPUTATIONAL MODE

Contents

31.1	Checkerboard mode	411
31.2	Filter for sea surface height	412
31.3	Filter for bottom pressure	412

The purpose of this chapter is to detail the methods used in MOM to suppress the B-grid computational mode associated with gravity waves. The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_barotropic.F90

31.1 Checkerboard mode

As discussed by [Mesinger \(1973\)](#), [Killworth et al. \(1991\)](#), [Pacanowski and Griffies \(1999\)](#), [Griffies et al. \(2001\)](#), and Section 12.9 of [Griffies \(2004\)](#), there is a ubiquitous problem with B-grid models due to a null mode present when discretizing inviscid gravity waves. This mode is absent on the C-grid, though there is another mode on the C-grid associated with the Coriolis force ([Adcroft et al., 1999](#)). The B-grid null mode manifests in the velocity field when using a relatively small viscosity. Additionally, it manifests in the surface height or bottom pressure, especially in coarsely represented enclosed or semi-enclosed embayments where waves tend to resonate rather than to propagate. The pattern is stationary¹ and appears as a plus-minus pattern; i.e., as a checkerboard. As there is generally no dissipation in the surface height budget

$$\partial_t \eta = \nabla \cdot \mathbf{U} + q_w, \quad (31.1)$$

suppression of the null mode requires some form of artificial dissipation. An analogous situation exists with the bottom pressure equation in a pressure based vertical coordinate model.

Various methods have been described in the literature (e.g., [Killworth et al., 1991](#); [Griffies et al., 2001](#)) to address this problem. The following constraints guide how we specify the filter.

- For the Boussinesq fluid, the tracer and surface height budgets must remain compatible in the sense defined in Section 10.7. Hence, if a filter is added to the surface height equation, one must correspondingly be added to the tracer equation. Likewise, the non-Boussinesq tracer equation must have a filter added if the bottom pressure equation has a filter.
- The filter should be zero in the case that the surface height is locally a constant, or if the bottom pressure remains locally the same as the reference bottom pressure.

¹Hence the term *null*, thus indicating it has a zero eigenvalue and zero phase speed.

31.2 Filter for sea surface height

The following filter for the surface height in a depth based vertical coordinate model has been found suitable for suppressing noise² in the model of Griffies et al. (2004), and a variant on this form sufficient for geopotential vertical coordinates was described in Section 12.9 of Griffies (2004)

$$\partial_t \eta = [\partial_t \eta]_{\text{no filter}} + \nabla \cdot (A \nabla \eta^*) \delta_{\nabla \eta} \quad (31.2)$$

$$\partial_t (dz t T) = [\partial_t (dz t T)]_{\text{no filter}} + \delta_{k,1} \nabla \cdot [A \nabla (\eta^* T)] \delta_{\nabla \eta}. \quad (31.3)$$

In these equations, $A > 0$ is a diffusivity, $\delta_{\nabla \eta}$ vanishes if the surface height is locally constant, and $\delta_{k,1}$ vanishes for all but the surface grid cell at $k = 1$. The surface height η^* is defined by

$$\eta^* = \eta + |\eta_{\min}| + \eta_{\text{off}} \quad (31.4)$$

where η_{\min} is the global minimum of the surface height, and η_{off} is a nonzero offset. The use of η^* rather than η does not alter the filter acting on the surface height, since $\nabla \eta^* = \nabla \eta$. However, η^* is important for the tracer, since it ensures that $\eta^* T$ has the same sign as T . If we instead diffused ηT , regions where $\eta < 0$ could lead to negative diffusion, which results in tracer extrema and potential instabilities. The alternative $\delta_{k,1} \nabla \cdot [A \nabla (dz t T)]$ is not desirable for z^* and σ vertical coordinates, in which case the bottom topography potentially adds a nontrivial level of smoothing even when the surface height is a uniform constant.

Global conservation of volume and tracer is ensured by using no-flux conditions at the side boundaries. Local conservation is ensured since the tracer and volume equations are compatible, as evidenced by setting the tracer concentration to unity in the tracer equation which, upon vertical integration, recovers the surface height equation (see Section 10.7). Note that a more conventional treatment of the filter in the tracer equation is given by the convergence of the thickness weighted flux $-\eta A \nabla T$. However, $\nabla \cdot (\eta A \nabla T)$ is not compatible with the filter applied to the surface height, and so this alternative approach will lead to local non-conservation.

31.3 Filter for bottom pressure

For pressure based vertical coordinates, we use the following filter

$$\partial_t p_b = [\partial_t p_b]_{\text{no filter}} + \nabla \cdot (A \nabla p_b^*) \delta_{\nabla p_b^*} \quad (31.5)$$

$$\partial_t (\rho dz t T) = [\partial_t (\rho dz t T)]_{\text{no filter}} + \delta_{k,kb} \nabla \cdot [A \nabla (g^{-1} p_b^* T)] \delta_{\nabla p_b^*}, \quad (31.6)$$

where $\delta_{\nabla p_b^*}$ vanishes where ∇p_b^* vanishes. The modified bottom pressure $p_b^* > 0$ is defined by

$$p_b^* = p_b - p_b^{\text{off}} + |\min(p_b - p_b^{\text{off}})| + p_b^{\text{off}}, \quad (31.7)$$

where p_b^{off} is a nonzero offset pressure. The positive pressure p_b^* is defined in a manner analogous to the definition of η^* given by equation (31.4). Note that subtracting out the reference bottom pressure p_b^{off} is useful prior to performing the Laplacian operations. Doing so ensures that the filter does not overly smooth the bottom pressure in regions where its undulations arise from variations in the bottom topography. Such variations are physical, and do not reflect a problem with the B-grid and so should not be filtered. The term $g^{-1} p_b^*$ appearing in the tracer equation acts like an effective mass per area

$$(\rho dz t)^{\text{eff}} \equiv g^{-1} p_b^*, \quad (31.8)$$

which is positive since $p_b^* > 0$. That is, the filter on the bottom pressure equation acts like a mass source in the bottom cell. Hence, tracer must be rearranged in the bottom cell in order to maintain compatibility (see Section 10.7). This filter satisfies the global and local conservation constraints, while adding a level of filtering needed to smooth the bottom pressure.

²For added scale selectivity, it is sometimes useful to employ a biharmonic operator instead of a Laplacian. However, much care should be exercised when using the biharmonic, as it is not positive definite and so can lead to negative tracer concentrations. Therefore, the biharmonic, although present in the MOM codes, is *not recommended* for general use.

Diagnostic capabilities

To rationalize a numerical simulation with MOM, one must perform a suite of analyses to connect the model results to those from theory and observations. This facet of numerical oceanography has long been fundamental to the utility of MOM, since it is distributed with great deal of analysis or diagnostic capabilities. In this part of the manual, we describe some of these features of MOM. The MOM developers encourage contributions of further diagnostic features from the broader MOM community as they may arise from various applications.

A question that is often asked: “Is there a tabulation of all diagnostics available in MOM?” There is unfortunately, no such tabulation. One means of obtaining a listing of diagnostics is through the following Unix command:

```
grep -whr -e "00\\|\\(register_diag_field[:blank:]]\\{0,\\}\\)" . | sed -e "s/^[/]*register_diag_field[:blank:]]\\{0,\\}(/" | uniq > diag_field
```


METHODS FOR DIAGNOSING MASS TRANSPORT

Contents

32.1	Brief on notation	416
32.2	Meridional-overturning streamfunction	416
32.2.1	Summary of mass conservation for a finite region	416
32.2.2	Zonally integrated mass transport	416
32.2.3	Mass transport streamfunction	417
32.2.4	Transport beneath an arbitrary surface	419
32.2.5	Transport from GM90	420
32.2.5.1	Boussinesq geopotential coordinates	420
32.2.5.2	Non-Boussinesq pressure coordinates	420
32.2.6	Diagnosing the streamfunction in Ferret from MOM output	421
32.3	Mass transport through tracer cell faces	421
32.3.1	Tracer cell mass transport from parameterized mesoscale and submesoscale eddies	422
32.3.2	Summary of the mass transport diagnostics	423
32.3.3	A caveat about masking for the GM and submesoscale streamfunctions	424
32.4	Vertically integrated transport	425
32.4.1	Summary of the rigid lid case	425
32.4.2	General case of divergent flow in non-Boussinesq fluids	425
32.4.3	Vertically integrated transport from eddy parameterizations	426

The purpose of this chapter is to present the various methods available in MOM to diagnose mass transport. We give particular attention to the mathematical formulation of streamfunctions commonly used to summarize the overturning circulation as well as the vertically integrated circulation. This chapter is updated from that presented in the MOM4.0 manual of [Griffies et al. \(2004\)](#).

The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_barotropic.F90
ocean_diag
ocean_param/neutral
ocean_param/mixing/ocean_submesoscale.F90
```

32.1 Brief on notation

In this chapter, we employ generalized orthogonal coordinates, in which the horizontal grid increments are given by

$$dx = h_1 d\xi^1 \quad (32.1)$$

$$dy = h_2 d\xi^2. \quad (32.2)$$

The stretching functions h_1 and h_2 are generally dependent on the horizontal position on the sphere, but independent of vertical position and independent of time. In spherical coordinates,

$$dx = R \cos \phi d\lambda \quad (32.3)$$

$$dy = R d\phi, \quad (32.4)$$

with R the earth's radius, λ the longitude, and ϕ the latitude.

32.2 Meridional-overturning streamfunction

The meridional overturning streamfunction is commonly used to diagnose features of the thermohaline circulation. The purpose of this section is to formulate an expression for this streamfunction, and in turn to highlight its limitations.

32.2.1 Summary of mass conservation for a finite region

In Section 2.6, we developed the conservation equations for scalar fields over a finite size region, such as a model grid cell. In particular, the mass budget for a grid cell (equation (2.155)) is written

$$\partial_t(dz\rho) = dz\rho S^{(M)} - \nabla_s \cdot (dz\rho \mathbf{u}) - (\rho w^{(s)})_{s=s_{k-1}} + (\rho w^{(s)})_{s=s_k}. \quad (32.5)$$

In the following, we neglect mass sources $S^{(M)}$ for brevity, but they may be easily reintroduced if present as part of a model's subgrid scale parameterization.¹ As noted by equation (2.156), the divergence operator acting on the flux of mass per unit horizontal area takes the form

$$\nabla_s \cdot (dz\rho \mathbf{u}) = \frac{1}{dy} \frac{\partial}{\partial x} (dy dz \rho u) + \frac{1}{dx} \frac{\partial}{\partial y} (dx dz \rho v), \quad (32.6)$$

which leads to the expanded form of mass conservation

$$\boxed{\begin{aligned} \partial_t(dx dy dz \rho) = & -dx \frac{\partial}{\partial x} (dy dz \rho u) - dy \frac{\partial}{\partial y} (dx dz \rho v) \\ & - (dx dy \rho w^{(s)})_{s=s_{k-1}} + (dx dy \rho w^{(s)})_{s=s_k}. \end{aligned}} \quad (32.7)$$

This result follows by assuming the horizontal area $dx dy$ of a grid cell is independent of time and depth.

32.2.2 Zonally integrated mass transport

We now consider the integrated mass transports

$$\mathcal{V} = \int_{\xi_a^1}^{\xi_b^1} dx (v \rho dz) \quad (32.8)$$

$$\mathcal{W} = \int_{\xi_a^1}^{\xi_b^1} dx (w \rho dy), \quad (32.9)$$

¹For example, the cross land scheme of Chapter 29 and the cross-land insertion scheme of Chapter 30 introduce mass source and sink terms.

which provide the meridional and vertical transport of mass (kg/sec) integrated along a line of constant generalized zonal coordinate ξ^1 . Integration endpoints ξ_a^1 and ξ_b^1 are assumed to be at land-sea boundaries, where u vanishes, or over a periodic domain. As in the spherical coordinate case, we refer to \mathcal{V} as the generalized meridional transport, though the generalized coordinates need not be aligned with the geographical latitude/longitude coordinates.

Now consider the meridional derivative of the meridional transport

$$\begin{aligned}\partial_y \mathcal{V} &= \frac{\partial}{\partial y} \left(\int_{\xi_a^1}^{\xi_b^1} dx (v \rho dz) \right) \\ &= \frac{1}{h_2 \partial \xi^2} \left(\int_{\xi_a^1}^{\xi_b^1} dx (v \rho dz) \right) \\ &= \frac{1}{h_2} \int_{\xi_a^1}^{\xi_b^1} \frac{\partial}{\partial \xi^2} (dx v \rho dz).\end{aligned}\tag{32.10}$$

In the last step, we allowed for the partial derivative operator $\partial/\partial \xi^2$ to commute with integration over paths with constant generalized zonal coordinate ξ^1 . This result follows since the generalized horizontal coordinates (ξ^1, ξ^2) are independent. Correspondingly, we may introduce the increment $d\xi^2$ inside and outside of the integral to render

$$\partial_y \mathcal{V} = \frac{1}{dy} \int_{\xi_a^1}^{\xi_b^1} dy \frac{\partial}{\partial y} (dx v \rho dz).\tag{32.11}$$

We now use the mass conservation equation (32.7) to find

$$\partial_y \mathcal{V} = \frac{1}{dy} \int_{\xi_a^1}^{\xi_b^1} \left(-\partial_t (dx dy dz \rho) - (dx dy \rho w^{(s)})_{s=s_{k-1}} + (dx dy \rho w^{(s)})_{s=s_k} \right),\tag{32.12}$$

where we cancelled the zonal transport term $u \rho dy dz$, since it either vanishes at the zonal boundaries (ξ_b^1, ξ_a^1) , or exhibits periodicity. Introducing the vertical transport (32.9) yields

$$\boxed{dy \partial_y \mathcal{V} + \delta_k \mathcal{W} = - \int_{\xi_a^1}^{\xi_b^1} \partial_t (dx dy dz \rho)}\tag{32.13}$$

with

$$\delta_k \mathcal{W} = \mathcal{W}_{k-1} - \mathcal{W}_k\tag{32.14}$$

the vertical finite increment of the vertical transport.

Equation (32.13) is a direct result of the mass conservation equation (32.7) applied to a zonal integral. It says that the divergence of the zonally integrated meridional and vertical transport equals to the zonally integrated tendency of mass. This result is the basis for our next step, which is to define a streamfunction.

32.2.3 Mass transport streamfunction

A streamfunction can be defined for any non-divergent transport. For the zonally integrated mass transport, equation (32.13) says there is zero divergence if the zonal integral of the time tendency of mass

vanishes. Zero time tendency arises if the top and bottom grid cell boundaries are set according to constant hydrostatic pressure surfaces, so that $\rho dz = -g^{-1} dp$. In this case, the mass within a cell is constant in time. Analogously, there is also a zero divergence of the volume transport for a Boussinesq fluid using geopotential vertical coordinates, in which case the volume of fluid in a cell is constant. In more general cases, however, there is no guarantee that the mass within a cell is constant. Additionally, in the presence of water crossing the ocean boundaries, mass will change, again leading to a nonzero divergence. Even with these caveats, the streamfunction defined below provides a very useful measure of the zonally integrated mass transport, thus motivating its near ubiquitous use as a model diagnostic.

In the case of a non-divergent zonally integrated mass transport, a mass transport streamfunction Ψ can be introduced according to the following specifications²

$$\mathcal{V} = -\delta_k \Psi \quad (32.15)$$

$$\mathcal{W} = \Psi_{,y}. \quad (32.16)$$

The streamfunction has dimensions mass/time, just as the meridional and vertical transports \mathcal{V} and \mathcal{W} . The typical oceanographic dimension for these transports is Sverdrup, where $1 \text{ Sv} = 10^9 \text{ kg sec}^{-1}$.

To derive a diagnostic expression for the streamfunction, we may start from either of the expressions (32.15) or (32.16). Integration and the use of boundary conditions then leads to expressions for Ψ , with the two expressions equivalent if the zonally integrated mass transport is non-divergent. In many simulations, the meridional transport \mathcal{V} is more easily computed than the vertical transport \mathcal{W} , making the specification $\mathcal{V} = -\delta_k \Psi$ the most common starting point. From a finite volume perspective, the relation $\mathcal{V} = -\delta_k \Psi$ says that the discretized streamfunction should be computed at the top and bottom interfaces of grid cells, so that its vertical difference across the cells then leads to the meridional transport through the cell's vertical side walls (see Figure 32.1). Correspondingly, the streamfunction is horizontally co-located with the meridional transport \mathcal{V} .³

The relation $\mathcal{V} = -\delta_k \Psi$ remains valid if we modify Ψ by any function of horizontal position, since the vertical difference eliminates the arbitrary function.⁴ We choose to exploit this ambiguity by specifying the arbitrary function so that the streamfunction has a zero value at the ocean bottom. We are motivated to take this choice since for most oceanographic purposes, there is no mass transport considered between the liquid ocean and solid earth. Consequently, the solid earth boundary condition is time independent, and for convenience we specify that it vanishes.

Let us now develop the streamfunction, starting from the ocean bottom using the definition $\mathcal{V} = -\delta_k \Psi$. For the top surface of the bottom-most cell with $k = k_{mt}$, we have

$$\Psi_{k_{mt}-1} = 0 - \mathcal{V}_{k_{mt}}, \quad (32.17)$$

where $0 = \Psi_{k_{mt}}$ is inserted as a place-holder for the next iteration, and $\mathcal{V}_{k_{mt}}$ is the meridional transport leaving the vertical side walls within the bottom-most cell. For the next cell up in the column, we have

$$\begin{aligned} \Psi_{k_{mt}-2} &= \Psi_{k_{mt}-1} - \mathcal{V}_{k_{mt}-1} \\ &= -\mathcal{V}_{k_{mt}} - \mathcal{V}_{k_{mt}-1}. \end{aligned} \quad (32.18)$$

Induction leads to the result

$$\begin{aligned} \Psi_K &= - \sum_{k=K+1}^{k_{mt}} \mathcal{V}_k \\ &= - \sum_{k=K+1}^{k_{mt}} \left(\int_{x_a}^{x_b} dx (v \rho dz) \right)_k, \end{aligned} \quad (32.19)$$

²This definition of Ψ has an associated sign convention, with the opposite convention just as valid mathematically, but chosen less frequently in practice.

³We are led to an alternative grid placement for Ψ if starting from the relation $\mathcal{W} = \Psi_{,y}$.

⁴This ambiguity represents a *gauge symmetry*, which can be exploited in whatever manner is most convenient.

with a continuous expression given by

$$\Psi(y, z, t) = - \int_{x_a}^{x_b} dx \int_{-H}^z v \rho dz'. \quad (32.20)$$

In the continuous expression, it is important to perform the vertical integral first, since the bottom topography $z = -H(x, y)$ is a function of the horizontal position.

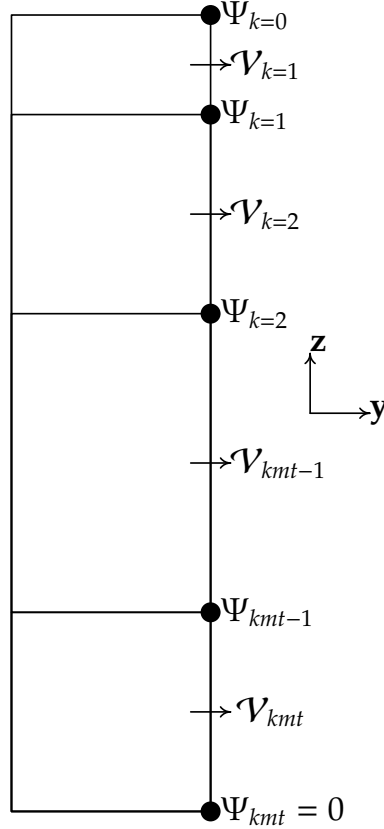


Figure 32.1: This figure illustrates the relation between the meridional-overturning streamfunction Ψ and the meridional transport \mathcal{V} , for the particular case of $kmt = 4$ vertical grid cells. The streamfunction is evaluated on the interfaces between the vertical cells, whereas the transport measures the mass leaving the cell in the meridional direction. Note the relatively thin bottom cell can arise from the use of bottom partial step representation of topography as used in MOM.

32.2.4 Transport beneath an arbitrary surface

As shown in Section 40.9 of [Pacanowski and Griffies \(1999\)](#), we can extend the above considerations to the case of generalized vertical coordinates. In this case, we are concerned with the meridional transport of fluid beneath some generalized vertical coordinate surface. It is a straightforward matter to extend the definition of the overturning streamfunction to this case, where

$$\Psi(y, s, t) = - \int_{x_a}^{x_b} dx \int_{s(-H)}^{s(z)} v \rho dz', \quad (32.21)$$

with $s = s(x, y, z, t)$ the generalized vertical coordinate (see [Griffies \(2004\)](#) for details). Surfaces that are physically of interest include various potential density surfaces, which are especially relevant when the flow is adiabatic. See Section 40.9 of [Pacanowski and Griffies \(1999\)](#) for more discussion.

32.2.5 Transport from GM90

The parameterization of [Gent and McWilliams \(1990\)](#) and [Gent et al. \(1995\)](#) provides a volume transport in addition to the resolved scale Eulerian mean transport. The total meridional-overturning streamfunction takes the form

$$\Psi^{(tot)}(y, z, t) = - \int_{x_a}^{x_b} dx \int_{-H}^z dz' \rho (v + v^{gm}). \quad (32.22)$$

We consider now special cases for the eddy-induced velocity v^{gm} .

32.2.5.1 Boussinesq geopotential coordinates

In the Boussinesq case with geopotential vertical coordinates, the meridional eddy-induced velocity is given by

$$v^{gm} = -\partial_z (\kappa S_y) \quad (32.23)$$

with $S_y = -\partial_y \rho / \partial_z \rho$ the neutral slope in the y -direction and $\kappa > 0$ a kinematic diffusivity. Performing the vertical integral on the GM90 piece leads to

$$\Psi^{(tot)}(y, z, t) = \Psi(y, z, t) + \Psi^{gm}(y, z, t) \quad (32.24)$$

where

$$\Psi^{gm}(y, z, t) = \rho_0 \int_{x_a}^{x_b} dx (\kappa S_y) \quad (32.25)$$

with $\kappa S_y = 0$ at $z = -H$, and ρ_0 the reference density for the Boussinesq fluid. Hence, the [Gent et al. \(1995\)](#) parameterization adds a contribution that scales linearly with basin size, isopycnal slope, and diffusivity

$$\Psi^{gm} \sim \rho_0 L S \kappa. \quad (32.26)$$

As an example, let $\kappa = 10^3 \text{m}^2 \text{s}^{-1}$, $S = 10^{-3}$, and $L = 10^7 \text{m}$, which yields $\mathcal{T} \approx 10 \text{Sv}$. Such transport can represent a nontrivial addition to that from the resolved scale velocity field.

32.2.5.2 Non-Boussinesq pressure coordinates

In the hydrostatic non-Boussinesq case with pressure vertical coordinates, the meridional eddy-induced velocity is given by

$$v^{gm} = \partial_p (g \rho \kappa S_y) \quad (32.27)$$

with $S_y = -\partial_y \rho / \partial_z \rho$ the neutral slope in the y -direction relative to constant pressure surfaces. Performing the vertical integral for the GM90 streamfunction leads to

$$\Psi^{gm}(y, z, t) = \int_{x_a}^{x_b} dx (\rho \kappa S_y). \quad (32.28)$$

32.2.6 Diagnosing the streamfunction in Ferret from MOM output

In MOM, there are two key diagnostics computed on-line that should be saved in order to determine an accurate expression for the streamfunction:

$$\text{ty_trans} = dx(v\rho dz) \quad (32.29a)$$

$$\text{ty_trans_gm} = dx(\rho\kappa S_y) \quad (32.29b)$$

Computing the Eulerian streamfunction (32.20) requires the following operations

$$\begin{aligned} \Psi(y, z, t) &= - \int_{x_a}^{x_b} dx \int_{-H}^z v \rho dz' \\ &= - \int_{x_a}^{x_b} dx \int_{-H}^{\eta} v \rho dz' + \int_{x_a}^{x_b} dx \int_z^{\eta} v \rho dz' \\ &= -\text{ty_trans}[i=@\text{sum}, k=@\text{sum}] + \text{ty_trans}[i=@\text{sum}, k=@\text{rsum}]. \end{aligned} \quad (32.30)$$

We must compute the streamfunction in this manner since Ferret's relative sum $k=@\text{rsum}$ starts from an assumed zero value at the surface and integrates downward, whereas the streamfunction has a zero boundary condition on the ocean bottom. Hence, without subtracting the term $\text{ty_trans}[i=@\text{sum}, k=@\text{sum}]$, the diagnosed streamfunction will incorrectly have nonzero values at the bottom. A nonzero value for $\text{ty_trans}[i=@\text{sum}, k=@\text{sum}]$ arises from net vertically integrated mass transport through a section. For a rigid lid model, this net transport vanishes. However, for a model with real water fluxes, this net transport will generally be nonzero.

The GM-streamfunction is simpler to compute, whereby

$$\Psi^{\text{gm}}(y, z, t) = \text{ty_trans_gm}[i=@\text{sum}]. \quad (32.31)$$

There is no vertical sum, since the vertical integral has already been performed analytically (equations (32.25) and (32.28)). In particular, $\Psi^{\text{gm}}(y, z, t)$ in the ocean surface cell equals $\text{ty_trans_gm}[i=@\text{sum}, k=1]$. Since the vertical integral of the GM-streamfunction vanishes, $\text{ty_trans_gm}[i=@\text{sum}, k=0]$ is zero, by definition, although this level is not explicitly saved in the output.

In a similar manner, the streamfunction from a skew flux implementation of the submesoscale eddy parameterization of Fox-Kemper et al. (2008b), Fox-Kemper et al. (2008a) and Fox-Kemper et al. (2011) (see Chapter 24) is computed as

$$\Psi^{\text{submeso}}(y, z, t) = \text{ty_trans_submeso}[i=@\text{sum}]. \quad (32.32)$$

There is no vertical sum required, since the vertical integral has already been performed analytically just as for the GM-streamfunction (see Section 24.3.2). Furthermore, as for the GM-streamfunction, the submesoscale transport vanishes when integrated over the full ocean depth, so that $\text{ty_trans_submeso}[i=@\text{sum}, k=0]$ is zero, by definition, although this level is not explicitly saved in the output.

32.3 Mass transport through tracer cell faces

MOM has diagnostics for the following three components to the mass transport from resolved flow passing through the tracer cell

$$\text{tx_trans} = u \rho dz dy \quad \text{tracer cell zonal face} \quad (32.33a)$$

$$\text{ty_trans} = v \rho dz dx \quad \text{tracer cell meridional face} \quad (32.33b)$$

$$\text{tz_trans} = w \rho dx dy \quad \text{tracer cell bottom face.} \quad (32.33c)$$

We note in these equations where the mass transport is centred on the tracer cell face. These fields are saved in units of either mass-Sverdrup (10^9 kg s^{-1} or mass transport kg s^{-1}).

32.3.1 Tracer cell mass transport from parameterized mesoscale and submesoscale eddies

MOM5 generally employs the skew flux approach of [Griffies \(1998\)](#), which means we generally do not diagnose the mass transport crossing a cell face associated with subgrid scale parameterizations. Instead, we diagnose a vertically integrated streamfunction for use in mapping overturning circulation. We diagnose a similar overturning streamfunction for the submesoscale scheme from [Fox-Kemper et al. \(2011\)](#). For many purposes, it is useful to diagnose the mass transport crossing through a cell face, which requires taking a vertical difference as per the schematic shown in Figure 32.1. We detail this diagnostic calculation here.

The overturning diagnostic fields saved for the mesoscale parameterization schemes (e.g., [Gent et al. \(1995\)](#)) are the following

$$\text{tx_trans_gm} \quad (32.34a)$$

$$\text{ty_trans_gm} \quad (32.34b)$$

whereas for the [Fox-Kemper et al. \(2011\)](#) scheme we use

$$\text{tx_trans_submeso} \quad (32.35a)$$

$$\text{ty_trans_submeso} \quad (32.35b)$$

Each of these arrays extends from $k = 1$ to $k = nk$, with nk the total number of vertical grid cells. For a vertical level k , the zonal fields are located at the bottom of a tracer cell and at the zonal face, whereas the meridional fields are located at the bottom of a tracer cell and at the meridional face (see Figure 32.1). The $k = 0$ field vanishes, so it is not diagnosed. Likewise, bottom cell (at $k = kmt$) transport vanishes, though it is written in the diagnosed field.

To develop the algorithm to diagnose the eddy-induced velocity, we note the [Gent et al. \(1995\)](#) meridional overturning diagnostic is given by

$$\text{ty_trans_gm} = \rho \, dx \, \kappa \, S^{(y)}, \quad (32.36)$$

where κ is the diffusivity and $S^{(y)}$ is the meridional slope of the neutral direction. The corresponding meridional eddy-induced velocity is

$$v^{\text{gm}} = -\partial_z (\kappa \, S^{(y)}). \quad (32.37)$$

A diagnostic calculation of the meridional eddy-induced velocity is thus given by

$$v^{\text{gm}}[k] = -\left(\frac{\text{ty_trans_gm}[k-1] - \text{ty_trans_gm}[k]}{\text{dxnt} \, \text{rho_dzt}} \right), \quad (32.38)$$

where $v^{\text{gm}}[k]$ is centred on the north face of the tracer cell, and dxnt is the zonal width (in meters) of the tracer grid cell at the northern face. More simply, we diagnose a meridional mass transport by taking the vertical difference

$$(v^{\text{gm}} \rho \, dx \, dz)[k] = -(\text{ty_trans_gm}[k-1] - \text{ty_trans_gm}[k]). \quad (32.39)$$

This transport corresponds directly to the mass transport component ty_trans diagnosed for the resolved velocity (equation (32.33b)).

We extend the above diagnostic results to the zonal component as well as to the submesoscale scheme. A summary of these diagnostic results for the eddy-induced velocity components is the following.

$$u^{\text{gm}}[k] = -\left(\frac{\text{tx_trans_gm}[k-1] - \text{tx_trans_gm}[k]}{\text{dyet} \, \text{rho_dzt}} \right) \quad (32.40a)$$

$$v^{\text{gm}}[k] = -\left(\frac{\text{ty_trans_gm}[k-1] - \text{ty_trans_gm}[k]}{\text{dxnt} \, \text{rho_dzt}} \right) \quad (32.40b)$$

$$u^{\text{submeso}}[k] = -\left(\frac{\text{tx_trans_submeso}[k-1] - \text{tx_trans_submeso}[k]}{\text{dyet} \, \text{rho_dzt}} \right) \quad (32.40c)$$

$$v^{\text{submeso}}[k] = -\left(\frac{\text{ty_trans_submeso}[k-1] - \text{ty_trans_submeso}[k]}{\text{dxnt} \, \text{rho_dzt}} \right), \quad (32.40d)$$

where dy_{et} is the meridional width (in meters) of the tracer cell at its eastern face. The corresponding mass transport is given by

$$(u^{gm} \rho dy dz)[k] = -(tx_trans_gm[k-1] - tx_trans_gm[k]) \quad (32.41a)$$

$$(v^{gm} \rho dx dz)[k] = -(ty_trans_gm[k-1] - ty_trans_gm[k]) \quad (32.41b)$$

$$(u^{submeso} \rho dy dz)[k] = -(tx_trans_submeso[k-1] - tx_trans_submeso[k]) \quad (32.41c)$$

$$(v^{submeso} \rho dx dz)[k] = -(ty_trans_submeso[k-1] - ty_trans_submeso[k]). \quad (32.41d)$$

As for the resolved mass transport diagnosed in tx_trans and ty_trans , these parameterized mass transports are centered on the corresponding zonal and meridional tracer cell face.

The vertical component to the parameterized mass transport is derived through continuity, in which the mass entering a cell from the parameterized scheme equals the mass leaving a cell. We start from the ocean bottom and move upwards in a column to derive the following expressions

$$(w^{gm} \rho dx dy)[i, j, k-1] = (w^{gm} \rho dx dy)[i, j, k] + (u^{gm} \rho dy dz)[i-1, j, k] - (u^{gm} \rho dy dz)[i, j, k] \quad (32.42a)$$

$$+ (v^{gm} \rho dx dz)[i, j-1, k] - (v^{gm} \rho dx dz)[i, j, k]$$

$$(w^{submeso} \rho dx dy)[i, j, k-1] = (w^{submeso} \rho dx dy)[i, j, k] + (u^{submeso} \rho dy dz)[i-1, j, k] - (u^{submeso} \rho dy dz)[i, j, k] \quad (32.42b)$$

$$+ (v^{submeso} \rho dx dz)[i, j-1, k] - (v^{submeso} \rho dx dz)[i, j, k]$$

The vertical transport from subgrid scale transport vanishes for the bottom of the bottom-most cell, and for the surface of the ocean surface cell.

32.3.2 Summary of the mass transport diagnostics

We now summarize the mass transport diagnostics available from the MOM CORE-II simulation. All transports are diagnosed in the respective face of the tracer cell

$$\text{zonal transport}[i] = \text{x-transport on east face of tracer cell} \quad (32.43a)$$

$$\text{meridional transport}[j] = \text{y-transport on north face of tracer cell} \quad (32.43b)$$

$$\text{vertical transport}[k] = \text{z-transport on bottom face of tracer cell.} \quad (32.43c)$$

Each diagnostic field has units of mass-Sverdrup (10^9 kg s^{-1}). As the model is configured using the Boussinesq kinematics, division by the constant reference density

$$\rho_o = 1035 \text{ kg m}^{-3} \quad (32.44)$$

brings the mass transports into volume-Sverdrup units ($10^6 \text{ m}^3 \text{ s}^{-1}$).

The following summarizes the diagnostics needed for the resolved mass transport through a tracer cell face

$$tx_trans = u \rho dz dy \quad (32.45a)$$

$$ty_trans = v \rho dz dx \quad (32.45b)$$

$$tz_trans = w \rho dx dy \quad (32.45c)$$

For the GM advective mass transport, we diagnose the advective mass transports

$$tx_trans_gm_adv = (u^{gm} \rho dy dz)[i, j, k] \quad (32.46a)$$

$$ty_trans_gm_adv = (v^{gm} \rho dx dz)[i, j, k] \quad (32.46b)$$

$$tz_trans_gm_adv = (w^{gm} \rho dx dy)[i, j, k] \quad (32.46c)$$

using the following algorithm

$$tx_trans_gm_adv[i, j, k] = -tx_trans_gm[i, j, k - 1] + tx_trans_gm[i, j, k] \quad (32.47a)$$

$$ty_trans_gm_adv[i, j, k] = -ty_trans_gm[i, j, k - 1] + ty_trans_gm[i, j, k] \quad (32.47b)$$

$$\begin{aligned} tz_trans_gm_adv[i, j, k - 1] &= tz_trans_gm_adv[i, j, k] \\ &\quad + tx_trans_gm_adv[i - 1, j, k] - tx_trans_gm_adv[i, j, k] \\ &\quad + ty_trans_gm_adv[i, j - 1, k] - ty_trans_gm_adv[i, j, k]. \end{aligned} \quad (32.47c)$$

For the vertical transport, we set the bottom boundary condition

$$tz_trans_gm_adv[i, j, k = kbot] = 0. \quad (32.48)$$

The top boundary condition

$$tz_trans_gm_adv[i, j, k = 0] = 0, \quad (32.49)$$

should be checked by determining whether

$$\begin{aligned} 0 \approx & tz_trans_gm_adv[i, j, k = 1] + tx_trans_gm_adv[i - 1, j, k = 1] - tx_trans_gm_adv[i, j, k = 1] \\ & + ty_trans_gm_adv[i, j - 1, k = 1] - ty_trans_gm_adv[i, j, k = 1]. \end{aligned} \quad (32.50)$$

For the parameterized submesoscale advective mass transport, we diagnose the advective mass transports

$$tx_trans_submeso_adv = (u^{submeso} \rho dy dz)[i, j, k] \quad (32.51a)$$

$$ty_trans_submeso_adv = (v^{submeso} \rho dx dz)[i, j, k] \quad (32.51b)$$

$$tz_trans_submeso_adv = (w^{submeso} \rho dx dy)[i, j, k] \quad (32.51c)$$

using the following algorithm

$$tx_trans_submeso_adv[i, j, k] = -tx_trans_submeso[i, j, k - 1] + tx_trans_submeso[i, j, k] \quad (32.52a)$$

$$ty_trans_submeso_adv[i, j, k] = -ty_trans_submeso[i, j, k - 1] + ty_trans_submeso[i, j, k] \quad (32.52b)$$

$$\begin{aligned} tz_trans_submeso_adv[i, j, k - 1] &= tz_trans_submeso_adv[i, j, k] \\ &\quad + tx_trans_submeso_adv[i - 1, j, k] - tx_trans_submeso_adv[i, j, k] \\ &\quad + ty_trans_submeso_adv[i, j - 1, k] - ty_trans_submeso_adv[i, j, k]. \end{aligned} \quad (32.52c)$$

For the vertical transport, we set the bottom boundary condition

$$tz_trans_submeso_adv[i, j, k = kmxl] = 0, \quad (32.53)$$

where $kmxl$ is the cell at the mixed layer base. The top boundary condition

$$tz_trans_submeso_adv[i, j, k = 0] = 0, \quad (32.54)$$

should be checked by determining whether

$$\begin{aligned} 0 \approx & tz_trans_submeso_adv[i, j, k = 1] + tx_trans_submeso_adv[i - 1, j, k = 1] - tx_trans_submeso_adv[i, j, k = 1] \\ & + ty_trans_submeso_adv[i, j - 1, k = 1] - ty_trans_submeso_adv[i, j, k = 1]. \end{aligned} \quad (32.55)$$

32.3.3 A caveat about masking for the GM and submesoscale streamfunctions

As shown in Figure 32.1, the discrete streamfunction $\Psi_{i,j,k}$ sits on the bottom corner of a tracer cell. As mentioned in the caption to this figure, $\Psi_{i,j,k}$ vanishes for points where there is land just beneath either adjacent ocean column. Unfortunately, the diagnostics in MOM5 do not properly account for the masking required to set the parameterized mesoscale or submesoscale streamfunctions to zero in such cases

where there is land at $k = kmt$ in one column but not in the other. To properly account for this mistake requires the offline masking to be applied to the following streamfunctions

$$tx_trans_gm[i, j, k] \rightarrow tx_trans_gm[i, j, k] * tmask[i, j, k + 1] * tmask[i + 1, j, k + 1] \quad (32.56a)$$

$$ty_trans_gm[i, j, k] \rightarrow ty_trans_gm[i, j, k] * tmask[i, j, k + 1] * tmask[i, j + 1, k + 1] \quad (32.56b)$$

$$tx_trans_submeso[i, j, k] \rightarrow tx_trans_submeso[i, j, k] * tmask[i, j, k + 1] * tmask[i + 1, j, k + 1] \quad (32.56c)$$

$$ty_trans_submeso[i, j, k] \rightarrow ty_trans_submeso[i, j, k] * tmask[i, j, k + 1] * tmask[i, j + 1, k + 1]. \quad (32.56d)$$

32.4 Vertically integrated transport

When the vertically integrated transport is non-divergent, it may be described by a streamfunction, which is often termed the *barotropic streamfunction*. This situation generally holds for a Boussinesq rigid lid model in the absence of surface water fluxes. However, MOM is no longer a rigid lid model, so one requires both a streamfunction and stream potential to describe the vertically integrated transport. Nonetheless, since for many purposes the vertically integrated transport is nearly non-divergent on annual and longer time scales, even in the presence of surface water fluxes, the *quasi*-streamfunction defined in the following is quite useful as an approximation to the vertically integrated transport.

32.4.1 Summary of the rigid lid case

With the rigid lid method of [Bryan \(1969\)](#), it is assumed that the vertically integrated velocity in a Boussinesq fluid is non-divergent $\nabla \cdot \mathbf{U} = 0$. Hence, it can be described via a scalar streamfunction $\mathbf{U} = \hat{\mathbf{z}} \wedge \nabla \tilde{\psi}$. In this case, the vertically integrated advective mass transport between two points is given by

$$T_{ab} = \rho_0 \int_a^b dl \, \hat{\mathbf{n}} \cdot \int_{-H}^0 dz \, \mathbf{u} = \rho_0 (\tilde{\psi}_a - \tilde{\psi}_b), \quad (32.57)$$

where dl is the line element along any path connecting the points a and b , and $\hat{\mathbf{n}}$ is a unit vector pointing perpendicular to the path in a rightward direction when facing the direction of integration. As written, T_{ab} has units of mass per time, and so it represents a mass transport ([Bryan, 1969](#)). Therefore, the difference between the streamfunction at two points represents the vertically integrated mass transport between the two points. It is for this reason that the streamfunction is sometimes called the *mass transport streamfunction*.

32.4.2 General case of divergent flow in non-Boussinesq fluids

The vertically integrated horizontal mass transport

$$\mathbf{U}^\rho = \int_{-H}^{\eta} \rho \mathbf{u} \quad (32.58)$$

generally has a non-zero divergence due to fluctuations of mass within the vertical column, as seen by the column integrated mass balance (equation (2.21))

$$\partial_t \left(\int_{-H}^{\eta} dz \rho \right) + \nabla \cdot \mathbf{U}^\rho = Q_w + \int_{-H}^{\eta} dz \rho \mathcal{S}^{(M)}. \quad (32.59)$$

As for the overturning transport, we ignore mass sources in the following, thus considering

$$\nabla \cdot \mathbf{U}^\rho = -\partial_t \left(\int_{-H}^{\eta} dz \rho \right) + Q_w. \quad (32.60)$$

In either case, the presence of a nonzero divergence requires the introduction of a streamfunction $\tilde{\psi}$ and a velocity potential χ

$$\mathbf{U}^\rho = \hat{\mathbf{z}} \wedge \nabla \tilde{\psi} + \nabla \chi. \quad (32.61)$$

Only for a Boussinesq rigid-lid model with zero fresh water flux will χ vanish. Hence, to compute the precise vertically integrated mass transport passing between two points, a direct evaluation of the integral

$$T_{ab} = \int_a^b dl \, \hat{\mathbf{n}} \cdot \mathbf{U}^\rho \quad (32.62)$$

is given. Although accurate and complete, this integral does not readily provide a horizontal map of transport, and so it loses much of the appeal associated with the transport streamfunction used with a rigid lid.

However, for many practical situations, maps of the function

$$\psi(x, y, t) = - \int_{y_0}^y dy' U^\rho(x, y', t) \quad (32.63)$$

are quite useful, where the lower limit y_0 is taken at the southern boundary of the domain, generally given by a solid wall for ocean climate models. By its definition, the meridional derivative of ψ yields the zonal mass transport

$$\psi_{,y} = U^\rho. \quad (32.64)$$

The zonal derivative, however, does not yield the meridional mass transport due to the divergent nature of the vertically integrated flow. It is for this reason that we denote ψ a *quasi-streamfunction*. Notably, for many cases, especially with long time averages, the divergence is small, thus allowing ψ to present a good indication of the path and intensity of the vertically integrated mass transport.

By construction, ψ reduces to the transport streamfunction in the case of a rigid lid where $\nabla \cdot \mathbf{U} = 0$. However, this is not a unique choice and alternatives do exist. For example,

$$\psi^*(x, y, t) = \psi(x_0, y, t) + \int_{x_0}^x dx' V^\rho(x', y, t), \quad (32.65)$$

gives

$$V^\rho = \psi^*_{,x}. \quad (32.66)$$

ψ^* has the advantage that zonal derivatives give the exact meridional transport, yet the meridional derivative in general deviates from the zonal transport. Comparing maps of ψ and ψ^* reveals the degree to which the vertically integrated mass transport is non-divergent. For most purposes of climate modeling at GFDL, we map the streamfunction $\psi = - \int_{y_0}^y dy' U^\rho$, as deviations from ψ^* are modest for most applications.

32.4.3 Vertically integrated transport from eddy parameterizations

As mentioned above, the vertically integrated transport from both the [Gent et al. \(1995\)](#) mesoscale eddy closure and the submesoscale scheme of [Fox-Kemper et al. \(2008b\)](#) vanish. Hence, the transport from these schemes add zero to the net vertically integrated transport through any ocean column.

KINETIC ENERGY DIAGNOSTICS

Contents

33.1	Formulation of kinetic energy diagnostics	427
----------------------	---	-----

The purpose of this chapter is to present the formulation of various kinetic energy diagnostics available in MOM.

The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_velocity_diag.F90

33.1 Formulation of kinetic energy diagnostics

The kinetic energy of the fluid in a grid cell for a discrete hydrostatic fluid is determined by the horizontal components

$$E_{\text{kinetic}} = \frac{1}{2} \rho \, dV \, \mathbf{u} \cdot \mathbf{u} \quad (33.1)$$

where $M = \rho \, dV$ is the mass of fluid in the grid cell. Note that the *in situ* density ρ reduces in the Boussinesq case to the constant Boussinesq reference density ρ_0 . The horizontal velocity vector is written as

$$\mathbf{u} = (u, v) \quad (33.2)$$

The SI unit for kinetic energy is Joule.

We often find it useful to split the kinetic energy into that determined by the barotropic, or vertically averaged flow, and that determined by the baroclinic flow. For this purpose, define the vertically averaged velocity according to

$$\bar{\mathbf{u}}^z = \left(\frac{\sum_{k=1}^{nk} \mathbf{u} \, \rho \, dz}{\sum_{k=1}^{nk} \rho \, dz} \right). \quad (33.3)$$

The deviation from this vertical average approximates the baroclinic velocity

$$\hat{\mathbf{u}} = \mathbf{u} - \bar{\mathbf{u}}^z. \quad (33.4)$$

We thus introduce the baroclinic kinetic energy

$$E_{\text{clinic}} = \frac{1}{2} \rho \, dV \, \hat{\mathbf{u}} \cdot \hat{\mathbf{u}} \quad (33.5)$$

and the barotropic kinetic energy

$$E_{\text{tropic}} = \frac{1}{2} \bar{\mathbf{u}}^z \cdot \bar{\mathbf{u}}^z \left(\sum_{k=1}^{nk} \rho dV \right). \quad (33.6)$$

Note that it is only when vertically integrating that we have the identity

$$\sum_{k=1}^{nk} E_{\text{kinetic}} = E_{\text{tropic}} + \sum_{k=1}^{nk} E_{\text{clinic}}. \quad (33.7)$$

EFFECTIVE DIANEUTRAL DIFFUSIVITY

Contents

34.1	Potential energy and APE in Boussinesq fluids	430
34.2	Effective dianeutral mixing	431
34.2.1	Global effective dianeutral diffusivity	431
34.2.2	Finite difference approximation	432
34.2.3	Relevant vertical stratification range	433
34.2.4	A useful test case	433
34.2.5	Computational precision	433
34.2.6	Negative κ_{eff}	434
34.2.7	A comment on convection	434
34.2.8	The experimental design	434
34.3	Modifications for time dependent cell thicknesses	434
34.4	An example with vertical density gradients	435
34.4.1	Evolution of the unsorted state	436
34.4.2	Evolution of the sorted state	437
34.4.3	Caveat about weakly stratified regions	440
34.5	An example with vertical and horizontal gradients	440
34.5.1	Vertical diffusion	440
34.5.1.1	Evolution of the unsorted state	440
34.5.1.2	Evolution of the sorted state	441
34.5.2	Horizontal diffusion	444
34.5.2.1	Evolution of the unsorted state	444
34.5.2.2	Evolution of the sorted state	444

The purpose of this chapter is to detail a method to quantify water mass mixing in MOM without detailed knowledge of the numerical transport scheme. The method is restricted to experiments configured with the following:

- Boussinesq fluid;
- linear free surface, so that the thickness of a grid cell remains constant in time; alternatively, to the use of z^* vertical coordinate, where each cell has a time independent spacing in z^* -space;
- flat bottom ocean;
- linear equation of state;

- each grid cell has the same volume in $x - y - s$ space, with s the general vertical coordinate;
- zero buoyancy forcing.

Relaxing some of these assumptions is possible, yet not implemented. Momentum forcing via winds is allowed. Much of the fundamentals in this chapter are guided by the work of [Winters et al. \(1995\)](#) and [Winters and D'Asaro \(1995\)](#). [Griffies et al. \(2000b\)](#) applied these methods to various idealized model configurations in a rigid lid version of MOM3.

We assume the linear equation of state for an incompressible fluid is written in the form

$$\rho = \rho_0(1 - \alpha \theta), \quad (34.1)$$

where θ is potential temperature, ρ_0 is a constant density associated with the Boussinesq approximation, and α is a constant thermal expansion coefficient. The system is open to momentum fluxes yet closed to buoyancy fluxes.

The following MOM module is directly connected to the material in this chapter:

ocean_diag/ocean_tracer_diag.F90

34.1 Potential energy and APE in Boussinesq fluids

The purpose of this section is to introduce the notion of a sorted density profile in the context of potential energy and available potential energy (APE). This profile is of particular relevance when considering the effective mixing occurring throughout a column of sorted fluid in Section 34.2.

Potential energy of the ocean is given by

$$E_p = \int dV \rho \mathcal{P}, \quad (34.2)$$

where

$$\mathcal{P} = g z \quad (34.3)$$

is the potential energy per mass of a fluid parcel, g is the acceleration of gravity, z is the vertical position of a fluid parcel, and $\rho dV = \rho dx dy dz$ is the parcel mass.

Available potential energy (APE) is the difference between the potential energy of the fluid in its natural state, and the potential energy of a corresponding stably stratified *reference* state. The reference state is reached by adiabatically rearranging the fluid to a state of minimum potential energy, which is a state that contains zero horizontal gradients. This rearrangement, or *sorting*, provides a non-local mapping between the unsorted fluid density and the sorted density

$$\rho(\mathbf{x}, t) = \rho(z^*(\mathbf{x}, t), t). \quad (34.4)$$

The sorting map determines a vertical position field $z^*(\mathbf{x}, t)$ which is the vertical height in the sorted state occupied by a parcel at (\mathbf{x}, t) in the unsorted state. Due to the monotonic arrangement of density in the sorted state, $z^*(\mathbf{x}, t)$ is a monotonic function of density $\rho(\mathbf{x}, t)$.

It is convenient to set the origin of the vertical coordinate at the ocean bottom so to keep potential energy of the unsorted state non-negative. This convention also allows for $z^*(\mathbf{x}, t)$ to be defined as a monotonically decreasing function of density. That is,

$$\rho(\mathbf{x}_1, t) < \rho(\mathbf{x}_2, t) \Rightarrow z^*(\mathbf{x}_1, t) > z^*(\mathbf{x}_2, t). \quad (34.5)$$

Conservation of volume in a flat bottom ocean implies that the sorted fluid state has the same vertical extent as the unsorted fluid, which renders

$$0 \leq z, z^* \leq H, \quad (34.6)$$

where H is the ocean depth.

In the following, it proves convenient to denote the density profile in the sorted reference state using the symbols

$$\rho(z^*, t) = \rho_{ref}(z = z^*, t). \quad (34.7)$$

Given this notation, the non-local sorting map between the unsorted and sorted fluid states provides the equivalence

$$\rho(\mathbf{x}, t) = \rho(z^*(\mathbf{x}, t), t) = \rho_{ref}(z = z^*, t). \quad (34.8)$$

In turn, potential energy for the sorted fluid state can be written in two equivalent manners

$$E_{ref} = g \int dV z \rho_{ref}(z, t) \quad (34.9)$$

$$= g \int dV z^*(\mathbf{x}, t) \rho(\mathbf{x}, t). \quad (34.10)$$

Equation (34.9) represents an integral over the sorted fluid state, in which the density of this state is a function only of the depth. The horizontal area integral is thus trivial to perform. Equation (34.10) represents an integral over the unsorted fluid state, where the density $\rho(\mathbf{x}, t)$ of an unsorted parcel is weighted by the vertical position $z^*(\mathbf{x}, t)$ that the parcel occupies in the sorted state. It follows that the APE can be written in two equivalent ways

$$E_{APE} = g \int dV z [\rho(\mathbf{x}, t) - \rho_{ref}(z, t)] \quad (34.11)$$

$$= g \int dV \rho(\mathbf{x}, t) [z - z^*(\mathbf{x}, t)]. \quad (34.12)$$

34.2 Effective dianeutral mixing

In this section we formulate a method to empirically quantify the effects on water masses arising from various simulated tracer transport processes. A similar application was advocated by [Winters et al. \(1995\)](#) and [Winters and D'Asaro \(1995\)](#) for the purpose of diagnosing mixing in direct numerical simulations of unstable fluid flows. Their focus was on physically motivated mixing such as that occurring with breaking waves. The main focus here is on spurious mixing due to numerical errors. The procedure is identical, however, in that for each case, one considers the evolution of the reference density profile, $\rho_{ref}(z, t)$, in a fluid system closed to buoyancy fluxes

$$\partial_t \rho_{ref} = \partial_{z^*} (\kappa_{eff} \partial_{z^*} \rho_{ref}). \quad (34.13)$$

Again, in this equation z^* is the vertical position in the sorted fluid state. Therefore, constant z^* surfaces represent constant density surfaces in the unsorted state. As such, the effective diffusivity κ_{eff} summarizes the total amount of mixing across constant density surfaces. If the simulation does not change the water mass distribution, then $D\rho/Dt = 0$, the sorted reference density is static $\partial_t \rho_{ref} = 0$, and the effective diffusivity is zero. In turn, any temporal change in the reference density represents changes in the water mass distribution. These changes are the result of dianeutral mixing, and so have an associated non-zero $\kappa_{eff}(z^*, t)$. This is the basic idea that is pursued in the following sections.

34.2.1 Global effective dianeutral diffusivity

In addition to the diffusivity $\kappa_{eff}(z^*, t)$, which is local in density space, it is useful to garner a summary of the overall dianeutral mixing occurring in an ocean model. A vertical integral of $\kappa_{eff}(z^*, t)$ would provide such information. A quicker computation of a global effective dianeutral diffusivity can be obtained by inverting the variance equation for the sorted density

$$\partial_t \int dV \rho_{ref}^2 = -2 \int dV \kappa_{eff} (\partial_{z^*} \rho_{ref})^2. \quad (34.14)$$

This result, derived for a closed fluid system, suggests the introduction of a global effective diffusivity

$$\kappa_{global}(t) = - \left(\frac{\partial_t \int dV \rho_{ref}^2}{2 \int dV (\partial_z \rho_{ref})^2} \right). \quad (34.15)$$

This diffusivity provides one number that can be used to represent the total amount of dianeutral diffusion acting over the full model domain. It vanishes when the simulation is adiabatic, as does the effective diffusivity $\kappa_{eff}(z^*, t)$. However it is generally different from the vertical average of $\kappa_{eff}(z^*, t)$.

34.2.2 Finite difference approximation

In the following, assume that the discrete sorted density is equally spaced in the vertical with a separation Δz^* , and let the vertical coordinate increase upwards from zero at the flat bottom ocean floor. Note that in general, $\Delta z^* \ll \Delta z$, where Δz is the ocean model's grid spacing. The reason is that *all* the $N_x \times N_y \times N_z$ grid points in the ocean model are sorted into the reference vertical profile, which has a vertical range over the same extent as the ocean model: $0 \leq z, z^* \leq H$. As a consequence, the vertical resolution of the sorted profile is $N_x \times N_y$ times finer than the N_z points resolving the profile at a particular horizontal position in the unsorted state.

On the discrete lattice, the vertical diffusive flux of the sorted density

$$F^{z^*}(z^*, t) = -\kappa_{eff}(z^*, t) \partial_{z^*} \rho_{ref}(z^*, t) \quad (34.16)$$

is naturally defined at the top face of the density cell whose center is at z^* . As such, the diffusion operator at the lattice point z^* , which is constructed as the convergence of the diffusive flux across a density grid cell, takes the discrete form

$$-(\partial_{z^*} F^{z^*})(z^*, t) \approx - \left(\frac{F^{z^*}(z^*, t - \Delta t) - F^{z^*}(z^* - \Delta z^*, t - \Delta t)}{\Delta z^*} \right). \quad (34.17)$$

The time lag is necessary to provide for a stable discretization of the diffusion equation. The discretization of the flux is given by

$$\begin{aligned} F^{z^*}(z^*, t) &= -\kappa_{eff}(z^*, t) \partial_{z^*} \rho_{ref}(z^*, t) \\ &\approx -\kappa_{eff}(z^*, t) \left(\frac{\rho_{ref}(z^* + \Delta z^*, t) - \rho_{ref}(z^*, t)}{\Delta z^*} \right). \end{aligned} \quad (34.18)$$

Since the flux is located at the top face of the density grid cell whose center is at the position z^* , the effective diffusivity is located at this face as well. Each of these difference operators is consistent with those used in MOM when discretizing the diffusion equation for the unsorted fluid.

As with the unsorted tendency, the time derivative in the effective diffusion equation can be approximated using a leap-frog differencing:

$$\partial_t \rho_{ref}(z^*, t) \approx \frac{\rho_{ref}(z^*, t + \Delta t) - \rho_{ref}(z^*, t - \Delta t)}{2\Delta t}. \quad (34.19)$$

Piecing these results together yields the expression for the vertical flux at the top of the density cell $z^* + \Delta z^*$

$$F^{z^*}(z^*, t - \Delta t) = F^{z^*}(z^* - \Delta z^*, t - \Delta t) - \frac{\Delta z^*}{2\Delta t} [\rho_{ref}(z^*, t + \Delta t) - \rho_{ref}(z^*, t - \Delta t)]. \quad (34.20)$$

This flux can be determined starting from the ocean bottom, where it vanishes, and working upwards. Without surface buoyancy fluxes, it also vanishes at the top of the water column, resulting in conservation of $\int dz^* \rho_{ref}(z^*, t)$. After diagnosing the flux from the tendency, the effective diffusivity can be diagnosed from

$$\kappa_{eff}(z^*, t) = -F^{z^*}(z^*, t) \left(\frac{\Delta z^*}{\rho_{ref}(z^* + \Delta z^*, t) - \rho_{ref}(z^*, t)} \right). \quad (34.21)$$

The issues of what to do when the density gradient becomes small, as in weakly stratified regions, is discussed in Sections 34.2.3 and 34.2.5.

34.2.3 Relevant vertical stratification range

In the stratified portions of the upper ocean, periods $2\pi/N$ for buoyancy oscillations are roughly 10-30 minutes, smaller in the pycnocline, and in the deep ocean periods are roughly 5-6 hours (see pages 55-56 of [Pickard and Emery \(1990\)](#)). The squared buoyancy frequency for the sorted reference state is given by

$$\begin{aligned} N_*^2 &= -\frac{g}{\rho_o} \frac{d\rho_{ref}}{dz^*} \\ &= -\frac{g}{1000\rho_o} \frac{d\sigma_{ref}}{dz^*}, \end{aligned} \quad (34.22)$$

where $\sigma_{ref} = 1000(\rho_{ref} - 1)$ is the sigma value for the sorted density $\rho_{ref}(g/cm^3)$. Working with σ_{ref} is desirable for accuracy reasons. The observed range in buoyancy periods provides a range over the sorted vertical profile's stratification for which a calculation of the model's effective diffusivity will be performed:

$$\frac{d\rho_{ref}}{dz^*} = -\frac{1.035g/cm^3}{980cm/sec^2} \frac{4\pi^2}{T^2}, \quad (34.23)$$

where $T(sec)$ is the period. With $1 \times 60secs < T < 6 \times 60 \times 60secs$ defining the period range, the corresponding vertical density gradient range is

$$10^{-10}g/cm^4 \leq \left| \frac{d\rho_{ref}}{dz} \right| \leq 10^{-5}g/cm^4, \quad (34.24)$$

and the corresponding range for the sigma gradient is

$$10^{-7}g/cm^4 \leq \left| \frac{d\sigma_{ref}}{dz} \right| \leq 10^{-2}g/cm^4. \quad (34.25)$$

34.2.4 A useful test case

When coding the effective diffusivity algorithm, it has been found useful to compare results with those from a different approach. Here, we horizontally average (i.e., homogenize) the density field along a particular depth surface. In a model with stable stratification, rigid lid, flat bottom, no-flux boundary, potential density evolution takes the form

$$\partial_t \langle \rho \rangle^{x,y} = -\partial_z \langle w \rho \rangle^{x,y} + \partial_z \langle \kappa \rho_{,z} \rangle^{x,y} \quad (34.26)$$

where κ is a vertical diffusivity and the angled-brackets indicate horizontally averaged quantities. With zero advection, evolution occurs solely via vertical diffusion. Hence, backing out an effective diffusivity for this horizontally homogenized system yields κ , regardless the horizontal/vertical stratification. It turns out that this algorithm is far simpler to implement numerically, since it does not require sorting nor interpolation to a prespecified sorted coordinate z^* . Its results are in turn more robust. Yet, importantly, they are relevant only for the case of no-advection, which is not so interesting in general yet serves as a good check for specific cases.

34.2.5 Computational precision

Models run with pure horizontal and/or vertical diffusion theoretically show $\kappa_{eff} \geq 0$ (see [Winters and D'Asaro \(1995\)](#) and [Griffies et al. \(2000b\)](#)). However, if the stratification range given by equations (34.24) or (34.25) is violated by more than roughly an order of magnitude, then spurious values of κ_{eff} tend to arise. These spurious values include unreasonably large values for κ_{eff} in regions of very low stratification, and negative values in regions of very large stratification. However, within the range given by equations (34.24) or (34.25), the computation yields reasonable values. For stratification outside this range, κ_{eff} is arbitrarily set to zero.

Another point to consider is that the stratification of ρ_{ref} shows much fine-scale step-like structure. Computing an effective diffusivity based on such a profile will in turn show lots of noise. Averaging over

the fine scales is therefore necessary to garner robust answers. That is, the spurious mixing diagnostic is smoother when having coarser vertical resolution. An objective means of averaging is to average ρ_{ref} vertically onto the same vertical grid used by the forward model in computing the unsorted density state. If this vertical stratification is itself very fine, then spurious values of κ_{eff} may still result, again due to not enough points of ρ_{ref} averaged into a single layer.

34.2.6 Negative κ_{eff}

Those advection schemes which contain dispersion, such as centered differenced advection, have leading order error terms that are not second order, but rather third order differential operators. Hence, the diagnosis of κ_{eff} for these schemes will likely to contain a fair amount of negative values. In turn, negative κ_{eff} may be interpreted as a sign of dispersion errors, which can create or destroy water masses. Upon introducing convection into the model, much of these *undershoots* and *overshoots* created by dispersion are rapidly mixed. In turn, the resulting κ_{eff} should become positive upon introducing convection.

Another source of negative κ_{eff} apparently can arise simply due to the finite sampling time and discrete grid, even in the case of pure diffusion. For example, if there is a mixing event, and if this event is under-sampled in time, it is possible that the sorted state may have density appear in a non-local manner. Such mixing events will lead to negative κ_{eff} . The ability to realize such values for κ_{eff} motivates a sampling time Δt equal to time step used to evolve the unsorted density.

34.2.7 A comment on convection

Although the relaxation experiments allow for a focus on adiabatic physics, in a z-coordinate model there is no guarantee that an experiment will remain vertically stable, especially if running with a nonzero wind stress. If convective adjustment is then allowed, water mass mixing will occur. Hence, the experiments which focus on advection must remove convective adjustment. In turn, the presence of convection is actually quite an important element in determining the effective amount of spurious water mass mixing occurring in the model. The reason is that certain advection schemes, through dispersion errors, introduce unstable water which is then mixed-out through convection. After determining the effective diffusivity from the pure advection experiments, it is appropriate to then allow convection to occur and to compare the amount of convection appearing with the various advection discretizations.

34.2.8 The experimental design

The framework developed in this section applies most readily to an ocean model with a linear equation of state run without any buoyancy forcing. Since the model is to be run with zero buoyancy forcing, it is necessary to spin-up to some interesting state and perform various relaxation experiments. An interesting alternative is to run with zero buoyancy forcing but nonzero wind forcing, such as in a wind driven gyre.

As a test of the implementation for the algorithm to compute κ_{eff} , it is useful to run a set of tests with pure horizontal and vertical diffusion; no advection or convection. These experiments are necessary to establish a baseline for later comparison. After being satisfied, a set of relaxation experiments should be run with advection and/or other transport processes enabled.

34.3 Modifications for time dependent cell thicknesses

There is presently no formulation of this diagnostic for the general case of a time dependent cell thickness. The problem is that the one-dimensional effective diffusion equation, written as

$$\partial_t(\Delta z^* \rho_{ref}) = F_n^{z^*} - F_{n+1}^{z^*} \quad (34.27)$$

to account for time dependent thicknesses, no longer satisfies the compatibility condition of [Griffies et al. \(2001\)](#). That is, for cases with all fluxes vanishing in the unsorted state, there is no guarantee that fluxes likewise vanish in the sorted state, since the evolution of Δz^* is no longer tied properly to its neighbor.

34.4 An example with vertical density gradients

It is useful to present some examples which can be readily worked through by hand. These examples highlight many of the points raised in the previous discussion, and provide guidance for interpreting the three-dimensional MOM results. Each of these examples considers the dynamics of the unsorted and sorted density fields when the unsorted field is affected by vertical and horizontal diffusion. For simplicity, we assume the thickness of all grid cells remains time independent. Extensions to the more general case were discussed in Section 34.3. We also assume a leap frog time stepping, though the analysis follows trivially for a forward time step, with $\tau - \Delta\tau$ converted to τ , and $2\Delta\tau$ converted to $\Delta\tau$.

The first example considers the initial density field shown in Figure 34.1. There are a total of $N = N_x N_y N_z = (4 \times 1) \times 3$ grid cells in this two-dimensional example. The density field has zero baroclinicity. So the question is: How does this state, and the corresponding sorted density state, evolve under the effects of vertical diffusion? Note the grid dimensions for the two states are related through

$$\Delta z = 4 \Delta z^*, \quad (34.28)$$

where z is the vertical coordinate for the unsorted state, and z^* is the vertical coordinate for the sorted state. For the following, it is convenient to define this state as that at time $(t - \Delta t)$. The potential energies of the unsorted and sorted states are easily computed to be

$$E_p(t - \Delta t) = 56 \rho_o g \Delta z V \quad (34.29)$$

$$E_{ref}(t - \Delta t) = 56 \rho_o g \Delta z V \quad (34.30)$$

$$E_{APE}(t - \Delta t) = 0 \quad (34.31)$$

where V is the volume of the grid cells, and ρ_o is the density scale. The zero APE is due to the absence of horizontal density gradients.

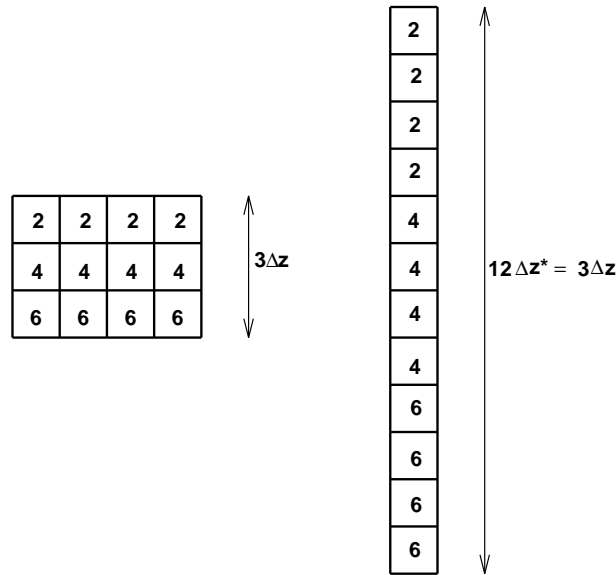


Figure 34.1: The initial density field for the first example. The number in each box represents the density, given in units of ρ_o . The left panel shows the density $\rho(x, z, t - \Delta t)$ in the unsorted fluid state, and the right panel shows the density $\rho_{ref}(z^*, t - \Delta t)$ in the sorted state. Note that the vertical scale $\Delta z^* = \Delta z/4$ for the sorted state has been expanded for purposes of display.

34.4.1 Evolution of the unsorted state

Evolution of the unsorted density is given by the discrete equation

$$\rho(x, z, t + \Delta t) = \rho(x, z, t - \Delta t) - \left(\frac{2\Delta t}{\Delta z} \right) [F^z(x, z, t - \Delta t) - F^z(x, z - \Delta z, t - \Delta t)], \quad (34.32)$$

where the vertical diffusive flux is given by

$$\begin{aligned} F^z(x, z, t) &= -\kappa \delta_z \rho(x, z, t) \\ &\approx -\kappa \left(\frac{\rho(x, z + \Delta z, t) - \rho(x, z, t)}{\Delta z} \right). \end{aligned} \quad (34.33)$$

$F^z(x, z, t)$ is defined at the top face of the density grid cell whose center has position (x, z) . In the following, it is useful to introduce the dimensionless quantity

$$\delta_{(v)} = 2\kappa \Delta t / (\Delta z)^2. \quad (34.34)$$

This number arises from the chosen discretization of the diffusion equation. For linear stability of the discretization, $\delta_{(v)} < 1$ must be maintained.

The top panel of Figure 34.2 shows the vertical diffusive flux through the cell faces at time $t - \Delta t$, and the bottom panel shows the resulting density $\rho(x, z, t + \Delta t)$. Density in the middle row does not change, whereas the upper row density increases and the lower row density decreases. The potential energy of this state is

$$E_p(t + \Delta t) = \rho_o g \Delta z V (56 + 8\delta_{(v)}). \quad (34.35)$$

This increase in potential energy is a result of the raised center of mass arising from the vertical diffusive fluxes.

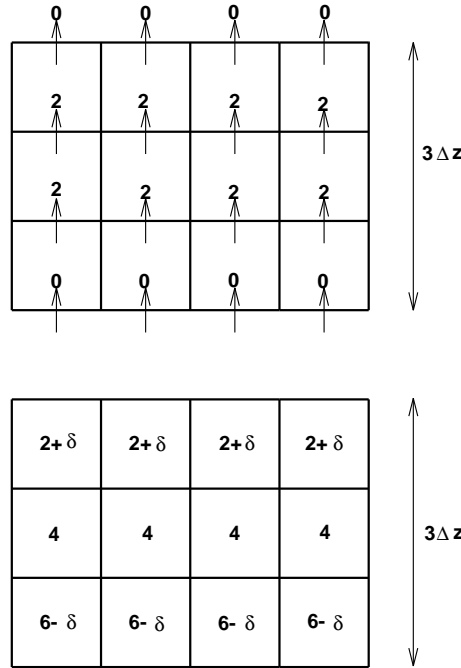


Figure 34.2: Top panel: The vertical diffusive flux $F^z(x, z, t - \Delta t)$, in units of $\rho_o \kappa / \Delta z$, passing through the faces of the unsorted density grid cells. Bottom panel: The unsorted density field $\rho(x, z, t + \Delta t)$, in units of ρ_o , where the dimensionless increment δ is given by $\delta = 2\delta_{(v)} = 4\kappa \Delta t / (\Delta z)^2$. This density field results from the vertical convergence of the flux $F^z(x, z, t - \Delta t)$.

34.4.2 Evolution of the sorted state

Corresponding to the evolution of the unsorted density, there is an evolution of the sorted density

$$\rho_{ref}(z^*, t + \Delta t) = \rho_{ref}(z^*, t - \Delta t) - \left(\frac{2\Delta t}{\Delta z^*} \right) [F^{z^*}(z^*, t - \Delta t) - F^{z^*}(z^* - \Delta z^*, t - \Delta t)]. \quad (34.36)$$

The dianeutral diffusive flux is

$$\begin{aligned} F^{z^*}(z^*, t) &= -\kappa_{eff}(z^*, t) \delta_{z^*} \rho_{ref}(z^*, t) \\ &\approx -\kappa_{eff}(z^*, t) \left(\frac{\rho_{ref}(z^* + \Delta z^*, t) - \rho_{ref}(z^*, t)}{\Delta z^*} \right), \end{aligned} \quad (34.37)$$

where $\rho_{ref}(z^*, t)$ is the sorted state's density. $F^{z^*}(z^*, t)$ is defined at the top face of the sorted density grid cell whose center has height z^* . Given the time tendency for the sorted state, the flux is diagnosed through

$$F^{z^*}(z^*, t - \Delta t) = F^{z^*}(z^* - \Delta z^*, t - \Delta t) - \left(\frac{\Delta z^*}{2\Delta t} \right) [\rho_{ref}(z^*, t + \Delta t) - \rho_{ref}(z^*, t - \Delta t)]. \quad (34.38)$$

The left panel of Figure 34.3 shows the sorted density field $\rho_{ref}(z^*, t + \Delta t)$, and the second panel shows the diagnosed vertical diffusive flux $F^{z^*}(z^*, t - \Delta t)$. The third panel shows the vertical density gradient $[\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)]/\Delta z^*$. The fourth panel shows the effective diffusivity $\kappa_{eff}(z^*, t - \Delta t)$, which is diagnosed from the relation

$$\kappa_{eff}(z^*, t - \Delta t) = -F^{z^*}(z^*, t - \Delta t) \left(\frac{\Delta z^*}{\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)} \right). \quad (34.39)$$

The units for $\kappa_{eff}(z^*, t - \Delta t)$ are $(\Delta z^*)^2/\Delta t$. Hence, a value for $\kappa_{eff}(z^*, t - \Delta t)$ of 2δ in Figure 34.3 indicates a dimensional value of

$$\begin{aligned} \kappa_{eff}(z^*, t - \Delta t) &= 2\delta \frac{(\Delta z^*)^2}{\Delta t} \\ &= \kappa \frac{4\Delta t}{(\Delta z)^2} \frac{(\Delta z^*)^2}{\Delta t} \\ &= \kappa/4. \end{aligned} \quad (34.40)$$

This example illustrates a problem with unstratified parts of the sorted profile. As evident from Figures 34.1 and 34.3, the 12 sorted boxes are actually three larger homogeneous boxes, and so the calculation should compute fluxes and diffusivities for these three boxes rather than for the 12 boxes. Figure 34.4 shows such a combined system, where there are three boxes each of height Δz comprising the sorted state. Repeating the previous calculation for this configuration recovers the expected $\kappa_{eff} = \kappa$ on the two interior interfaces. Note that there is no *ad hoc* setting to zero certain values of κ_{eff} associated with unstratified portions of the profile.

As a final note, the potential energy of the sorted state at time $t + \Delta t$ is

$$E_{ref}(t + \Delta t) = \rho_o g \Delta z V (56 + 16\delta_{(v)}), \quad (34.41)$$

which is higher than the initial potential energy as a result of the raised center of mass. The APE remains unchanged

$$E_{APE}(t + \Delta t) = 0, \quad (34.42)$$

as it should since there remains zero baroclinicity in the final state.

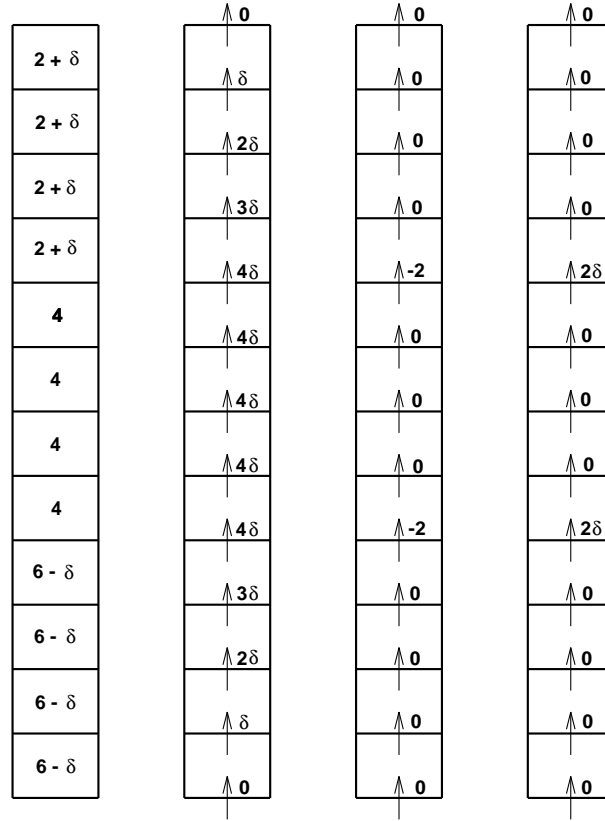


Figure 34.3: First panel (left): The sorted density field $\rho_{ref}(z^*, t + \Delta t)$, in units of ρ_o . Second panel: The vertical diffusive flux $F^{z^*}(z^*, t - \Delta t)$, in units of $\rho_o \Delta z^*/(2\Delta t)$, passing through the faces of the sorted density grid cells. Third panel: The vertical density gradient $[\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)]/\Delta z^*$ in units of $\rho_o/\Delta z^*$. Fourth panel: The effective diffusivity $\kappa_{eff}(z^*, t - \Delta t)$ in units of $(\Delta z^*)^2/\Delta t$.

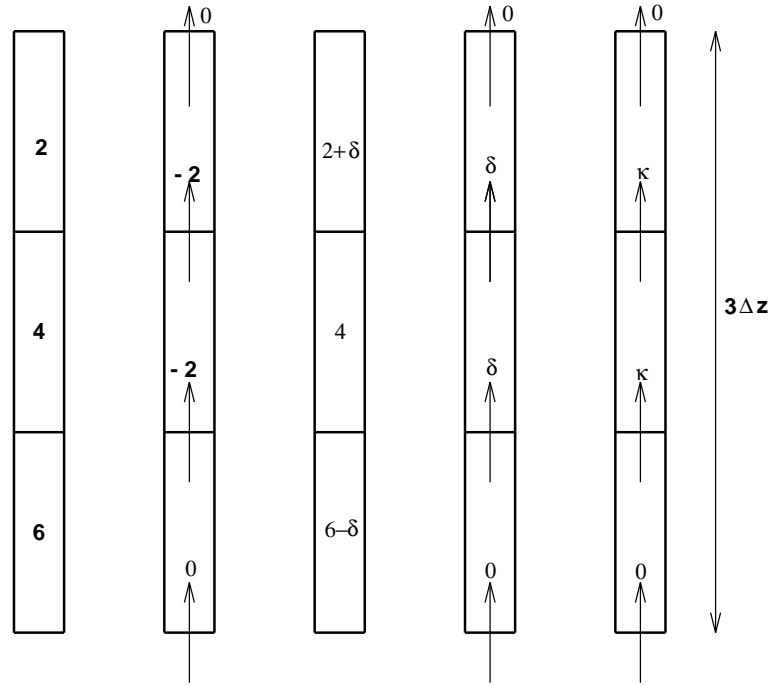


Figure 34.4: First panel (far left): The initial density field $\rho_{ref}(z^*, t - \Delta t)$, consisting of the combination of the three groups of four homogeneous cells. The values are given in units of ρ_o . In this recombined arrangement, $\Delta z^* = \Delta z$. Second panel: The vertical density gradient $[\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)]/\Delta z^*$, in units of $\rho_o/\Delta z^*$. Third panel: The density $\rho_{ref}(z^*, t + \Delta t)$ in units of ρ_o . Fourth panel: The diffusive flux $F^z(z^*, t - \Delta t)$. Fifth panel (far right): The effective diffusivity $\kappa_{eff}(z^*, t - \Delta t)$.

34.4.3 Caveat about weakly stratified regions

Note that in this example, the same set of boxes are perfectly homogenized at each time step. As such, it is straightforward to combine the boxes in order to derive their effective diffusivities. In general, this simple situation will not be true, and so the effective height of the combined boxes will differ. Furthermore, most cases of homogenization are approximate (Sections 34.2.3 and 34.2.5), which introduces even more time dependence to the interfaces between effectively homogeneous boxes. In order to compute an effective diffusivity, however, our algorithm needs to evaluate all quantities at the same depth level z^* . Time dependent z^* is problematical.

The current example suggests that one possible way to account for homogenization is to count the number of nearly homogeneous boxes occurring in a particular section of the sorted column. When the first interface is reached that has a nontrivial stratification, then the effective diffusivity computed for this interface is multiplied by the number of trailing boxes which are homogeneous. This trick works for the example just considered ($\kappa/4 \times 4 = \kappa$). However, in the example considered in Section 34.5.2, it leads to an effective diffusivity which can be larger than the horizontal diffusivity. Such is not possible, and so one is led to reject the proposed patch. A clean way to proceed is to try to resolve as best as possible the stratification within the sorted fluid state. For those regions which are simply too weakly stratified, it must be recognized that the computed effective diffusivity might be smaller than a more refined computation.

An alternative approach is to average the sorted density field onto the discrete levels realized in the unsorted state. Indeed, this resolution of the sorted state is arguably that which is relevant for diagnosing the effective diffusivity. This is the approach taken with the MOM experiments documented in Griffies et al. (2000b).

34.5 An example with vertical and horizontal gradients

This example considers the initial unsorted density configuration is shown in Figure 34.5. There are three rows of four boxes stacked on top of one another, and there are both vertical and horizontal density gradients. Also shown is the corresponding sorted state. As with the example in Section 34.4, the grid dimensions for the two states are related through $\Delta z = 4\Delta z^*$, where z is the vertical coordinate for the unsorted state, and z^* is the vertical coordinate for the sorted state. The potential energies are

$$E_p(t - \Delta t) = 110\rho_o g \Delta z V \quad (34.43)$$

$$E_{ref}(t - \Delta t) = 98\rho_o g \Delta z V \quad (34.44)$$

$$E_{APE}(t - \Delta t) = 12\rho_o g \Delta z V, \quad (34.45)$$

where V is the volume of the boxes and ρ_o is the density scale.

34.5.1 Vertical diffusion

Consider first just vertical diffusion acting on the unsorted state. The vertical diffusivity κ acting on the unsorted state is assumed to be uniform and constant.

34.5.1.1 Evolution of the unsorted state

Evolution of the unsorted density is given by the discrete equation

$$\rho(x, z, t + \Delta t) = \rho(x, z, t - \Delta t) - \left(\frac{2\Delta t}{\Delta z} \right) [F^z(x, z, t - \Delta t) - F^z(x, z - \Delta z, t - \Delta t)], \quad (34.46)$$

where the vertical diffusive flux is given by

$$\begin{aligned} F^z(x, z, t) &= -\kappa \delta_z \rho(x, z, t) \\ &\approx -\kappa \left(\frac{\rho(x, z + \Delta z, t) - \rho(x, z, t)}{\Delta z} \right). \end{aligned} \quad (34.47)$$

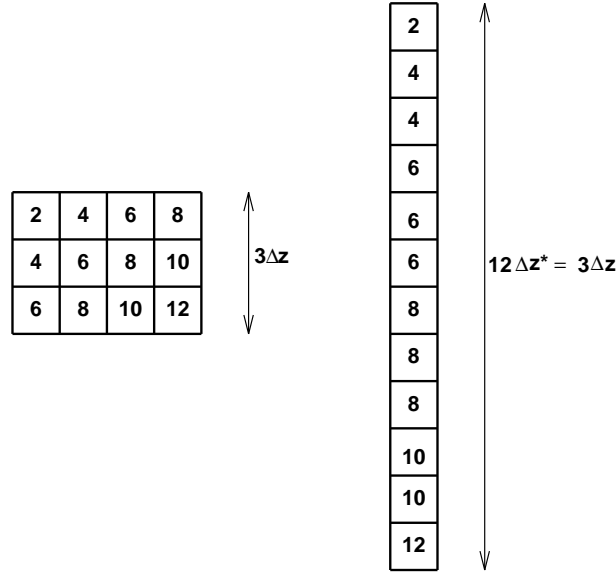


Figure 34.5: The initial density field for the horizontal and vertical diffusion examples. The number in each box represents the density, given in units of ρ_o . The left panel shows the density $\rho(x, z, t - \Delta t)$ in the unsorted fluid state, and the right panel shows the density $\rho_{ref}(z^*, t - \Delta t)$ in the sorted state. Note that the vertical scale $\Delta z^* = \Delta z/4$ for the sorted state has been expanded for purposes of display.

$F^z(x, z, t)$ is defined at the top face of the density grid cell whose center has height z . The top panel of Figure 34.6 shows the vertical diffusive flux through these faces at time $t - \Delta t$, and the bottom panel shows the resulting density field $\rho(x, z, t + \Delta t)$. Density in the middle row does not change, whereas the upper row density increases and the lower row density decreases. The potential energy of this state is

$$E_p(t + \Delta t) = \rho_o g \Delta z V (110 + 16 \delta_{(v)}), \quad (34.48)$$

which is higher than the initial potential energy as a result of the raised center of mass.

34.5.1.2 Evolution of the sorted state

Corresponding to the evolution of the unsorted density, there is an evolution of the sorted density which is given by

$$\rho_{ref}(z^*, t + \Delta t) = \rho_{ref}(z^*, t - \Delta t) - \left(\frac{2\Delta t}{\Delta z^*} \right) [F^{z^*}(z^*, t - \Delta t) - F^{z^*}(z^* - \Delta z, t - \Delta t)]. \quad (34.49)$$

The dianeutral diffusive flux is given by

$$\begin{aligned} F^{z^*}(z^*, t) &= -\kappa_{eff}(z^*, t) \delta_{z^*} \rho_{ref}(z^*, t) \\ &\approx -\kappa_{eff}(z^*, t) \left(\frac{\rho_{ref}(z^* + \Delta z^*, t) - \rho_{ref}(z^*, t)}{\Delta z^*} \right), \end{aligned} \quad (34.50)$$

where $\rho_{ref}(z^*, t)$ is the sorted state's density. $F^{z^*}(z^*, t)$ is defined at the top face of the sorted density grid cell whose center has height z^* . Given the time tendency for the sorted state, the flux is diagnosed through

$$F^{z^*}(z^*, t - \Delta t) = F^{z^*}(z^* - \Delta z^*, t - \Delta t) - \left(\frac{\Delta z^*}{2\Delta t} \right) [\rho_{ref}(z^*, t + \Delta t) - \rho_{ref}(z^*, t - \Delta t)]. \quad (34.51)$$

The left panel of Figure 34.7 shows the sorted density field $\rho_{ref}(z^*, t + \Delta t)$, and the second panel shows the diagnosed vertical diffusive flux $F^{z^*}(z^*, t - \Delta t)$. The third panel shows the vertical density gradient

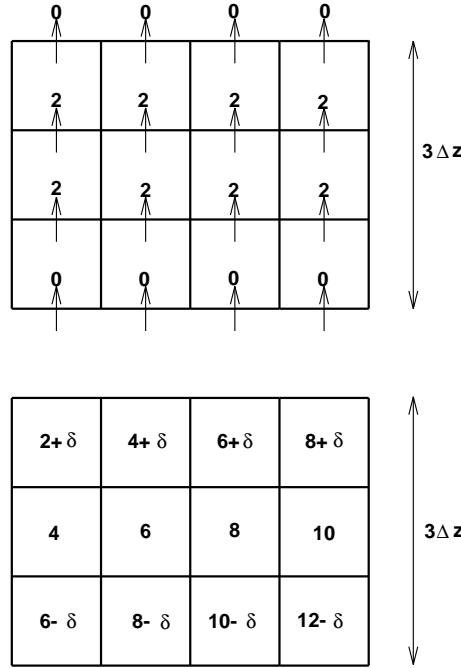


Figure 34.6: Top panel: The vertical diffusive flux $F^z(x, z, t - \Delta t)$, in units of $\rho_o \kappa / \Delta z$, passing through the faces of the unsorted density grid cells. Bottom panel: The unsorted density field $\rho(x, z, t + \Delta t)$, in units of ρ_o , where $\delta = 2\delta_{(v)} = 4\kappa \Delta t / (\Delta z)^2$. This is the density field resulting from the vertical convergence of the flux $F^z(x, z, t - \Delta t)$. The potential energy of this field is $E_p(t + \Delta t) = \rho_o g \Delta z V (110 + 16\delta_{(v)})$.

$[\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)] / \Delta z^*$. Note the regions of zero stratification. The fourth panel shows the effective diffusivity $\kappa_{eff}(z^*, t - \Delta t)$, which is diagnosed from the relation

$$\kappa_{eff}(z^*, t - \Delta t) = -F^{z^*}(z^*, t - \Delta t) \left(\frac{\Delta z^*}{\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)} \right). \quad (34.52)$$

The units for $\kappa_{eff}(z^*, t - \Delta t)$ are $(\Delta z^*)^2 / \Delta t$. In addition, consistent with the discussion in Section 34.4.2, the effective diffusivity for the interfaces on top of unstratified water are multiplied by the number of unstratified boxes. A value for $\kappa_{eff}(z^*, t - \Delta t)$ of δ in Figure 34.7 indicates a dimensional value of

$$\begin{aligned} \kappa_{eff}(z^*, t - \Delta t) &= \delta \frac{(\Delta z^*)^2}{\Delta t} \\ &= \kappa \frac{4\Delta t}{(\Delta z)^2} \frac{(\Delta z^*)^2}{\Delta t} \\ &= \kappa/4. \end{aligned} \quad (34.53)$$

As a final note, the potential energy of the sorted state at time $t + \Delta t$ is

$$E_{ref}(t + \Delta t) = \rho_o g \Delta z V (98 + 7\delta_{(v)}), \quad (34.54)$$

which is higher than the initial potential energy as a result of the raised center of mass. The APE is therefore given by

$$E_{APE}(t + \Delta t) = \rho_o g \Delta z V (12 + 11\delta_{(v)}), \quad (34.55)$$

which is larger than $E_{APE}(t - \Delta t)$ given in equation (34.45).

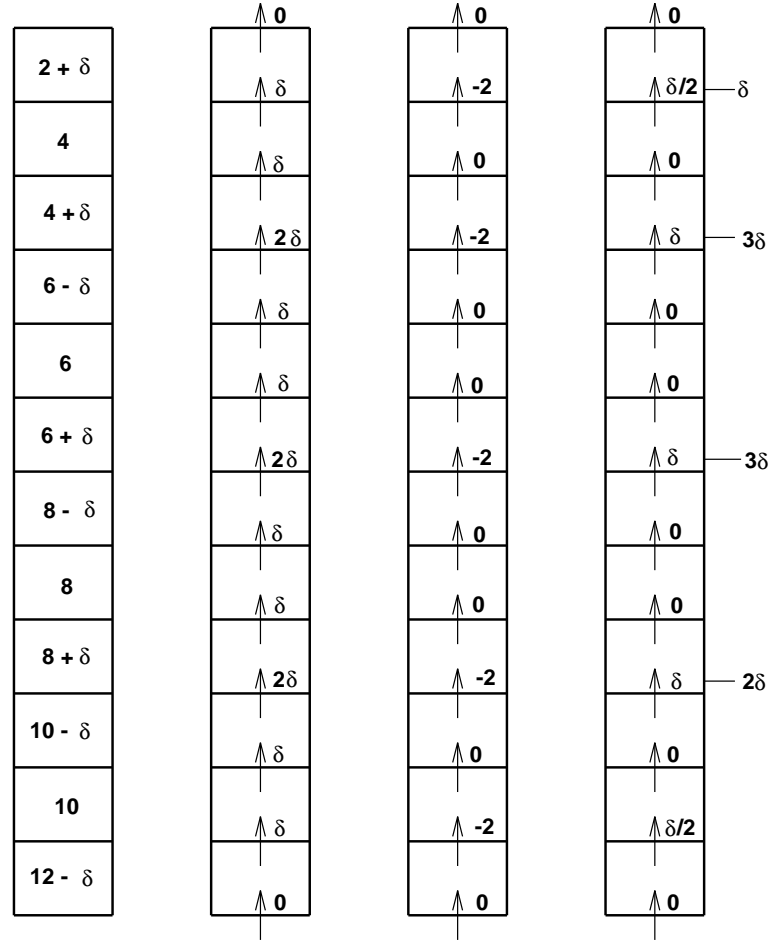


Figure 34.7: Left panel: The sorted density field $\rho_{ref}(z^*, t + \Delta t)$, in units of ρ_o . Second panel: The vertical diffusive flux $F^{z^*}(z^*, t - \Delta t)$, in units of $\rho_o \Delta z^* / \Delta t$, passing through the faces of the sorted density grid cells. Third panel: The vertical density gradient $[\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)] / \Delta z^*$ in units of $\rho_o / \Delta z^*$. Fourth panel: The effective diffusivity $\kappa_{eff}(z^*, t - \Delta t)$ in units of $(\Delta z^*)^2 / \Delta t$. The four κ_{eff} values which are on top of unstratified portions of the $\rho_{ref}(z^*, t - \Delta t)$ profile have been multiplied by the number of unstratified boxes which lie directly beneath it.

34.5.2 Horizontal diffusion

Consider now just horizontal diffusion acting on the unsorted state. The horizontal diffusivity A acting on the unsorted state is assumed to be uniform and constant.

34.5.2.1 Evolution of the unsorted state

Evolution of the unsorted density is given by the discrete equation

$$\rho(x, z, t + \Delta t) = \rho(x, z, t - \Delta t) - \left(\frac{2\Delta t}{\Delta x} \right) [F^x(x, z, t - \Delta t) - F^x(x - \Delta x, z, t - \Delta t)], \quad (34.56)$$

where the horizontal diffusive flux is given by

$$\begin{aligned} F^x(x, z, t) &= -A \delta_x \rho(x, z, t) \\ &\approx -A \left(\frac{\rho(x + \Delta x, z, t) - \rho(x, z, t)}{\Delta x} \right). \end{aligned} \quad (34.57)$$

$F^x(x, z, t)$ is defined at the east face of the density grid cell whose center has position (x, z) . The top panel of Figure 34.8 shows the horizontal diffusive flux through these faces at time $t - \Delta t$, and the bottom panel shows the resulting density field $\rho(x, z, t + \Delta t)$. The potential energy of this state is the same as the initial potential energy, since the horizontal fluxes are parallel to the geopotential

$$E_p(t + \Delta t) = E_p(t - \Delta t) = 110 \rho_o g \Delta z V. \quad (34.58)$$

34.5.2.2 Evolution of the sorted state

Corresponding to the evolution of the unsorted density, there is an evolution of the sorted density which is given by

$$\rho_{ref}(z^*, t + \Delta t) = \rho_{ref}(z^*, t - \Delta t) - \left(\frac{2\Delta t}{\Delta z^*} \right) [F^{z^*}(z^*, t - \Delta t) - F^{z^*}(z^* - \Delta z, t - \Delta t)]. \quad (34.59)$$

The dianeutral diffusive flux is given by

$$\begin{aligned} F^{z^*}(z^*, t) &= -\kappa_{eff}(z^*, t) \delta_{z^*} \rho_{ref}(z^*, t) \\ &\approx -\kappa_{eff}(z^*, t) \left(\frac{\rho_{ref}(z^* + \Delta z^*, t) - \rho_{ref}(z^*, t)}{\Delta z^*} \right), \end{aligned} \quad (34.60)$$

where $\rho_{ref}(z^*, t)$ is the sorted state's density. $F^{z^*}(z^*, t)$ is defined at the top face of the sorted density grid cell whose center has height z^* . Given the time tendency for the sorted state, the flux is diagnosed through

$$F^{z^*}(z^*, t - \Delta t) = F^{z^*}(z^* - \Delta z^*, t - \Delta t) - \left(\frac{\Delta z^*}{2\Delta t} \right) [\rho_{ref}(z^*, t + \Delta t) - \rho_{ref}(z^*, t - \Delta t)]. \quad (34.61)$$

The left panel of Figure 34.5.2.2 shows the sorted density field $\rho_{ref}(z^*, t + \Delta t)$, and the second panel shows the diagnosed vertical diffusive flux $F^{z^*}(z^*, t - \Delta t)$. The third panel shows the vertical density gradient $[\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)]/\Delta z^*$. Note the regions of zero stratification. The fourth panel shows the effective diffusivity $\kappa_{eff}(z^*, t - \Delta t)$, which is diagnosed from the relation

$$\kappa_{eff}(z^*, t - \Delta t) = -F^{z^*}(z^*, t - \Delta t) \left(\frac{\Delta z^*}{\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)} \right). \quad (34.62)$$

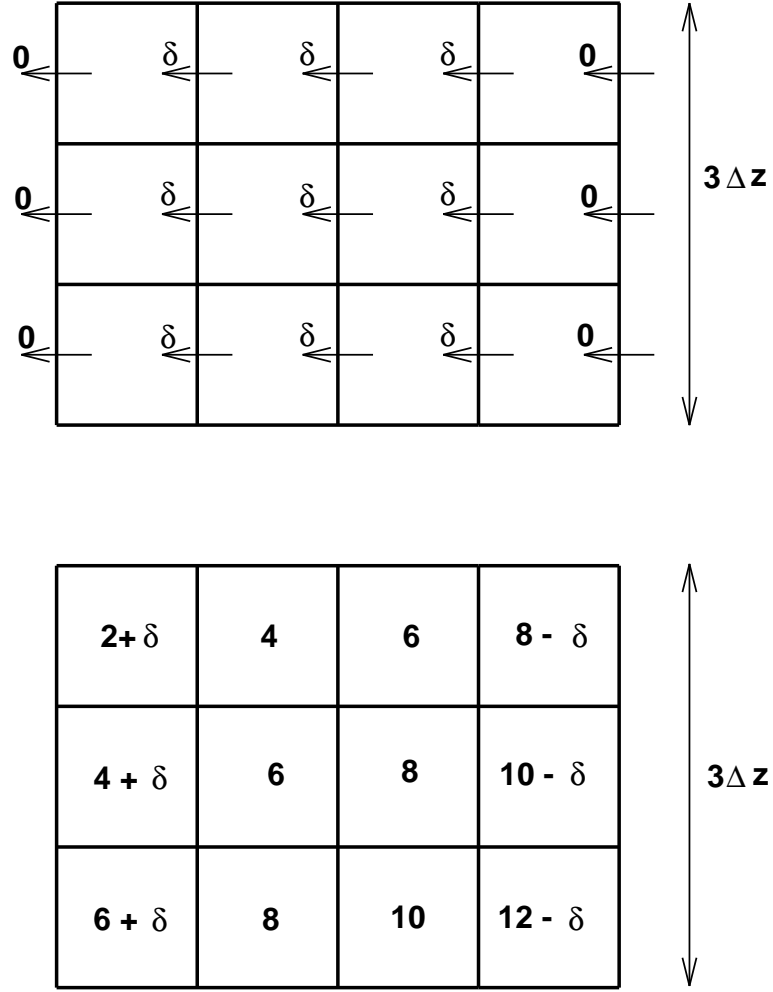


Figure 34.8: Top panel: The horizontal diffusive flux $F^x(x, z, t - \Delta t)$, in units of $\rho_o A / \Delta x$, passing through the faces of the unsorted density grid cells. Bottom panel: The unsorted density field $\rho(x, z, t + \Delta t)$, in units of ρ_o , where $\delta = 2\delta_{(h)} = 4A\Delta t / (\Delta x)^2$. This is the density field resulting from the vertical convergence of the flux $F^x(x, z, t - \Delta t)$. The potential energy of this field is $E_p(t + \Delta t) = 110\rho_o g \Delta z V$.

The units for $\kappa_{eff}(z^*, t - \Delta t)$ are $(\Delta z^*)^2/\Delta t$. For example, a value for $\kappa_{eff}(z^*, t - \Delta t)$ of $3\delta/2$ in Figure 34.5.2.2 indicates a dimensional value of

$$\begin{aligned}\kappa_{eff}(z^*, t - \Delta t) &= (3\delta/2) \frac{(\Delta z^*)^2}{\Delta t} \\ &= 6A \frac{\Delta t}{(\Delta x)^2} \frac{(\Delta z^*)^2}{\Delta t} \\ &= 6A \left(\frac{\Delta z^*}{\Delta x} \right)^2.\end{aligned}\tag{34.63}$$

For the special case of $\Delta x = \Delta z = 4\Delta z^*$, the effective diffusivity is $3A/8$. Note that if the patch proposed in Section 34.4.3 is used, then the $3\delta/2$ diffusivity would become $9\delta/2$, leading to the possibility for an effective diffusivity of $9A/8$, which is impossible.

As a final note, the potential energy of the sorted state at time $t + \Delta t$ is

$$E_{ref}(t + \Delta t) = \rho_o g \Delta z V (98 + 19\delta_{(h)}/2),\tag{34.64}$$

which is higher than the initial potential energy as a result of the raised center of mass. The APE is given by

$$E_{APE}(t + \Delta t) = \rho_o g \Delta z V (12 - 19\delta_{(h)}/2),\tag{34.65}$$

which is smaller than $E_{APE}(t - \Delta t)$ given in equation (34.45).

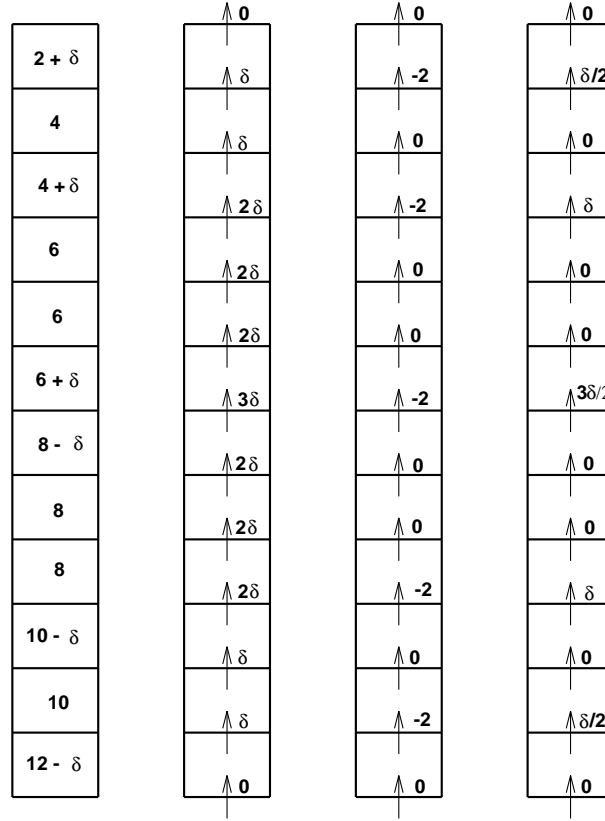


Figure 34.9: First panel (far left): The sorted density field $\rho_{ref}(z^*, t + \Delta t)$ in units of ρ_o . Second panel: The vertical diffusive flux $F^{z^*}(z^*, t - \Delta t)$, in units of $\rho_o \Delta z^* / \Delta t$, passing through the faces of the sorted density grid cells. Third panel: The vertical density gradient $[\rho_{ref}(z^* + \Delta z^*, t - \Delta t) - \rho_{ref}(z^*, t - \Delta t)] / \Delta z^*$ in units of $\rho_o / \Delta z^*$. Fourth panel: The effective diffusivity $\kappa_{eff}(z^*, t - \Delta t)$ in units of $(\Delta z^*)^2 / \Delta t$. The four κ_{eff} values which are on top of unstratified portions of the $\rho_{ref}(z^*, t - \Delta t)$ profile have been multiplied by the number of unstratified boxes which lie directly beneath it.

SPURIOUS DISSIPATION FROM NUMERICAL ADVECTION

Contents

35.1	Formulation of the method for Boussinesq fluid	449
35.2	Formulation for MOM	451
35.3	Comparing to physical mixing	453

The purpose of this chapter is to detail a method to locally quantify the level of dissipation (either positive or negative) associated with discretization errors in numerical tracer advection. This method was introduced by [Burchard and Rennau \(2008\)](#), and it provides a valuable complement to the effective diapycnal diffusivity diagnostic detailed in Chapter 34. In particular, the [Burchard and Rennau \(2008\)](#) method can provide a local quantification of the dissipation for any three dimensional model simulation. It cannot, however, generally translate that mixing into a diapycnal diffusivity, since there is no knowledge of neutral directions built into the diagnostic.

The following MOM module is directly connected to the material in this chapter:

`ocean_tracers/ocean_tracer_advect.F90`

35.1 Formulation of the method for Boussinesq fluid

Consider a continuous Boussinesq fluid in the absence of mixing, subgrid-scale fluxes, or sources, in which case the tracer concentration is affected only by advection

$$\partial_t C = -\nabla \cdot (\mathbf{v} C). \quad (35.1)$$

Likewise, the squared tracer concentration (indeed, the tracer concentration raised to any power) satisfies the same equation

$$\partial_t C^2 = -\nabla \cdot (\mathbf{v} C^2). \quad (35.2)$$

Now consider a space-time discretization of the tracer concentration equation ([35.1](#))

$$C^{n+1} = C^n - \mathcal{D}(C) \quad (35.3)$$

where we assume an explicit two-time level update of the tracer concentration onto discrete time levels $\tau_n = \tau_0 + n \Delta \tau$. This is the preferred time stepping method in MOM (see Chapter 11). The discrete operator

$$\mathcal{D}(C) \approx \Delta \tau \nabla \cdot (\mathbf{v} C) \quad (35.4)$$

symbolizes one of the many possible methods used to discretize tracer advection. For later purposes, it is useful to define the right hand side of equation (35.3) as the operator $\mathcal{A}(C)$, so that

$$\mathcal{A}(C) = C^n - \mathcal{D}(C). \quad (35.5)$$

Given the discrete expression (35.3), we can form the square of the updated tracer concentration

$$\begin{aligned} [(C^{n+1})^2]_{(a)} &= [\mathcal{A}(C)]^2 \\ &= (C^n)^2 + [\mathcal{D}(C)]^2 - 2C^n \mathcal{D}(C). \end{aligned} \quad (35.6)$$

Additionally, another means for computing the updated squared tracer concentration is to discretize the continuous equation for C^2 , given by equation (35.2), in which the updated squared tracer concentration is given by

$$[(C^{n+1})^2]_{(b)} = (C^n)^2 - \mathcal{D}(C^2). \quad (35.7)$$

Following the definition of the operator $\mathcal{A}(C)$ in equation (35.5), we have

$$[(C^{n+1})^2]_{(b)} = \mathcal{A}(C^2). \quad (35.8)$$

The fundamental question that this diagnostic asks is how well the two approximations for $(C^{n+1})^2$ agree. To answer this question, we simply take the difference

$$\begin{aligned} [(C^{n+1})^2]_{(a)} - [(C^{n+1})^2]_{(b)} &= [\mathcal{A}(C)]^2 - \mathcal{A}(C^2) \\ &= [\mathcal{D}(C)]^2 - 2C^n \mathcal{D}(C) + \mathcal{D}(C^2). \end{aligned} \quad (35.9)$$

A nonzero value for this difference results from nonzero spurious mixing or unmixing due to truncation errors in the advection scheme. Computing this difference requires an evaluation of the advection operator on both the tracer concentration and the squared tracer concentration.

To associate mixing with a particular sign of the difference (35.9), we consider the special case of one-dimensional advection with a constant advection velocity $u > 0$, discretized with first order upstream spatial differences on a uniform grid with spacing Δx . In this case

$$\begin{aligned} [(C^{n+1})^2]_{(a)} &\equiv \mathcal{A}(C)^2 \\ &= (1 - \gamma)^2 (C_i^n)^2 + \gamma^2 (C_{i-1}^n)^2 + 2\gamma(1 - \gamma) C_i^n C_{i-1}^n \end{aligned} \quad (35.10)$$

and

$$\begin{aligned} [(C^{n+1})^2]_{(b)} &\equiv \mathcal{A}(C^2) \\ &= (1 - \gamma)(C_i^n)^2 + \gamma(C_{i-1}^n)^2, \end{aligned} \quad (35.11)$$

where $\gamma = u \Delta \tau / \Delta x$ is the Courant number, and the discrete advection equation is stable so long as $0 \leq \gamma \leq 1$. With this chosen discretization, the difference (35.9) takes the form

$$\begin{aligned} \frac{[\mathcal{A}(C)]^2 - \mathcal{A}(C^2)}{\Delta \tau} &= -\left(\frac{\gamma(1 - \gamma)}{\Delta \tau}\right) (C_i^n - C_{i-1}^n)^2 \\ &= -\left(\frac{\gamma(1 - \gamma)(\Delta x)^2}{\Delta \tau}\right) \left(\frac{C_i^n - C_{i-1}^n}{\Delta x}\right)^2 \\ &\approx -\kappa_{\text{eff}} (\partial_x C)^2 \\ &\leq 0, \end{aligned} \quad (35.12)$$

where we identified $(C_i - C_{i-1})/\Delta x$ as a discrete approximation to $\partial_x C$, and defined an effective diffusivity

$$\kappa_{\text{eff}} = \left(\frac{\gamma(1 - \gamma)(\Delta x)^2}{\Delta \tau}\right). \quad (35.13)$$

Hence, the difference between the two approximations to the updated squared tracer concentration takes the form of a discrete dissipation of tracer variance, thus exemplifying the well known dissipative property of first order upwind advection. The effective diffusivity that sets the scale of this dissipation vanishes when the Courant number is either zero, which is the trivial case of no advective transport, or unity, in which case the full contents of cell $i - 1$ are transported into cell i over a single time step.

Burchard and Rennau (2008) take the previous result as motivation to define a numerically induced dissipation rate for any advection scheme. We follow their definition, yet introduce a division by $(\Delta\tau)^{-2}$ rather than their use of $\Delta\tau$

$$\begin{aligned}\Sigma &\equiv \frac{\mathcal{A}(C^2) - [\mathcal{A}(C)]^2}{(\Delta\tau)^2} \\ &= \frac{2C^n \mathcal{D}(C) - [\mathcal{D}(C)]^2 - \mathcal{D}(C^2)}{(\Delta\tau)^2}.\end{aligned}\tag{35.14}$$

The extra $\Delta\tau$ factor is motivated by dimensional arguments given in the discussion following the more general result given by equation (35.29). Again motivated by the one-dimensional upwind advection case, Burchard and Rennau (2008) propose the following identifications

$$\Sigma > 0 \Rightarrow \text{positive dissipation through mixing} \tag{35.15}$$

$$\Sigma < 0 \Rightarrow \text{negative dissipation through unmixing} \tag{35.16}$$

$$\Sigma = 0 \Rightarrow \text{zero dissipation.} \tag{35.17}$$

Regardless whether these identifications are rigorous, they are suggestive and allow one to stratify different advection schemes according to their values of Σ . In general, a key goal of an advection scheme is to have Σ from advection much smaller than the corresponding physically induced dissipation from subgrid scale mixing.

35.2 Formulation for MOM

The previous formulation was based on finite differences applied to a Boussinesq fluid. MOM is based on a finite volume formulation of the non-Boussinesq fluid. In this section, we generalize the previous considerations to that appropriate for MOM.

For a non-Boussinesq fluid, mass conservation takes the form

$$\partial_t \rho = -\nabla \cdot (\rho \mathbf{v}). \tag{35.18}$$

Likewise, tracer conservation in the absence of subgrid scale fluxes is given by

$$\partial_t (\rho C) = -\nabla \cdot (\rho \mathbf{v} C), \tag{35.19}$$

and the corresponding budget for squared tracer concentration is

$$\partial_t (\rho C^2) = -\nabla \cdot (\rho \mathbf{v} C^2). \tag{35.20}$$

MOM time steps a finite volume tracer budget for a non-Boussinesq fluid using generalized level coordinates, in which case the budget for tracer mass per horizontal area is time stepped. As detailed in Section 2.6, this approach leads to the following tracer advection equation for an interior grid cell

$$\partial_t (C \rho dz) = -\nabla_s \cdot (\mathbf{u} C \rho dz) - (\rho w^{(s)} C)_{s=s_{k-1}} + (\rho w^{(s)} C)_{s=s_k}, \tag{35.21}$$

with $w^{(s)}$ the dia-surface velocity component. Following the notation from Section 35.1, we discretize this budget as

$$\begin{aligned}(C \rho dz)^{n+1} &= (C \rho dz)^n - \mathcal{E}(C) \\ &\equiv \mathcal{B}(C),\end{aligned}\tag{35.22}$$

where the discrete advection operator is written

$$\mathcal{E}(C)/\Delta\tau \approx \nabla_s \cdot (\mathbf{u} C \rho dz) + (\rho w^{(s)} C)_{s=s_{k-1}} - (\rho w^{(s)} C)_{s=s_k}. \quad (35.23)$$

Likewise, the discretized budget for the mass weighted squared tracer concentration is

$$\begin{aligned} (C^2 \rho dz)^{n+1} &= (C^2 \rho dz)^n - \mathcal{E}(C^2) \\ &\equiv \mathcal{B}(C^2), \end{aligned} \quad (35.24)$$

where $\mathcal{B}(C^2)$ has dimensions of squared tracer concentration times mass per area.

We follow the steps considered in Section 35.1 to derive an operator that identifies the dissipation due to truncation errors with tracer advection. For this purpose, we consider the square of equation (35.22) as one approximation to the updated squared mass per area of a tracer

$$\begin{aligned} [(C \rho dz)^{n+1}]_{(a)}^2 &= [\mathcal{B}(C)]^2 \\ &= [(C \rho dz)^n]^2 + [\mathcal{E}(C)]^2 - 2\mathcal{E}(C)(C \rho dz)^n. \end{aligned} \quad (35.25)$$

An alternative approximation is obtained from equation (35.24), in which

$$\begin{aligned} [(C \rho dz)^{n+1}]_{(b)}^2 &= (\rho dz)^{n+1} \mathcal{B}(C^2) \\ &= (\rho dz)^{n+1} (\rho dz)^n (C^n)^2 - (\rho dz)^{n+1} \mathcal{E}(C^2). \end{aligned} \quad (35.26)$$

The difference between these two approximations is given by

$$\begin{aligned} [\mathcal{B}(C)]^2 - (\rho dz)^{n+1} \mathcal{B}(C^2) &= (C^2 \rho dz)^n [(\rho dz)^n - (\rho dz)^{n+1}] \\ &\quad + [\mathcal{E}(C)]^2 - 2(C \rho dz)^n \mathcal{E}(C) + (\rho dz)^{n+1} \mathcal{E}(C^2). \end{aligned} \quad (35.27)$$

The first term arises from time tendencies in the mass per area of a grid cell

$$\frac{(\rho dz)^n - (\rho dz)^{n+1}}{\Delta\tau} \approx -\partial_t(\rho dz), \quad (35.28)$$

and this term has nothing to do with errors in the advection scheme. Notably, it vanishes in two special cases:

- Boussinesq fluid with z -coordinates, in which case the cell thickness dz is time independent. The only exception is the top level, where the thickness changes due to time tendencies in the free surface height.
- Non-Boussinesq fluid with pressure coordinates, in which case the cell mass per area ρdz is time independent. The only exception is the bottom cell, where the mass per area changes due to time tendencies in the bottom pressure.

To remove these effects from temporal changes in mass per area of a grid cell, we define the generalized advection dissipation operator

$$\Sigma_{\text{general}} \equiv \frac{2(C \rho dz)^n \mathcal{E}(C) - [\mathcal{E}(C)]^2 - (\rho dz)^{n+1} \mathcal{E}(C^2)}{(\Delta\tau)^2}. \quad (35.29)$$

This definition corresponds to the operator Σ defined by equation (35.14). Even though we have ignored the $\partial_t(\rho dz)$ term, the operator (35.29) can still be nonzero even if the advection operators are perfect, as occurs when ρdz has horizontal spatial variations or time variations. Hence, this operator provides insight towards the advection truncation errors only for the special cases listed in the above two bullet points. Indeed, if the mass per area is the same temporal constant on a k -level (as for geopotential coordinates below the surface cell, and pressure coordinates above the bottom cell), we have

$$\Sigma_{\text{general}} = (\rho dz) \Sigma \quad \text{if } \nabla_s(\rho dz) = 0, \quad \text{and } \partial_t(\rho dz) = 0, \quad (35.30)$$

where the dissipation operator Σ is defined by equation (35.14).

The extra $\Delta\tau$ factor in the denominator of (35.29), relative to Burchard and Rennau (2008), provides sensible units for the tracer dissipation. In particular, for tracer $C = C_p \theta$ being the heat capacity times potential temperature, Σ_{general} has dimensions of $(\text{Watt}/\text{m}^2)^2$. For $C = 1000 S$ the mass of salt per mass of seawater, Σ_{general} has dimensions of $(\text{kg m}^{-2} \text{s}^{-1})^2$.

35.3 Comparing to physical mixing

As a means to gauge the levels of Σ_{general} , we may compare it to tracer dissipation arising from physically motivated subgrid scale mixing processes. For this purpose, we introduce a subgrid scale flux \mathbf{J} , so that the tracer concentration and squared tracer concentration satisfy the following equations

$$\partial_t(\rho C) + \nabla \cdot (\rho \mathbf{v} C) = -\nabla \cdot \mathbf{J} \quad (35.31)$$

$$\partial_t(\rho C^2) + \nabla \cdot (\rho \mathbf{v} C^2) = -C \nabla \cdot \mathbf{J}. \quad (35.32)$$

For the squared tracer, write the right hand side in the following manner

$$\begin{aligned} \partial_t(\rho C^2) + \nabla \cdot (\rho \mathbf{v} C^2) &= -C \nabla \cdot \mathbf{J} \\ &= -\nabla \cdot (C \mathbf{J}) + \nabla C \cdot \mathbf{J}. \end{aligned} \quad (35.33)$$

Considering a tensor formulation for the subgrid scale tracer flux

$$\mathbf{J} = -\rho \mathbf{K} \cdot \nabla C \quad (35.34)$$

leads to

$$\partial_t(\rho C^2) + \nabla \cdot (\rho \mathbf{v} C^2) = -\nabla \cdot (C \mathbf{J}) - \rho \nabla C \cdot \mathbf{K} \cdot \nabla C. \quad (35.35)$$

For a symmetric diffusion tensor,

$$\rho \nabla C \cdot \mathbf{K} \cdot \nabla C > 0, \quad (35.36)$$

in which case the mass weighted squared tracer concentration is dissipated by the sink $-\rho \nabla C \cdot \mathbf{K} \cdot \nabla C$. For the special case of vertical diffusion with diffusivity $\kappa > 0$,

$$\rho \nabla C \cdot \mathbf{K} \cdot \nabla C = \rho \kappa (\partial_z C)^2. \quad (35.37)$$

A finite volume formulation of the squared tracer equation, focusing just on the dissipation from vertical diffusion, leads to

$$(C^2 \rho dz)^{n+1} = (C^2 \rho dz)^n - \Delta\tau [\rho dz \kappa (\partial_z C)^2]^n, \quad (35.38)$$

which leads to

$$[(C \rho dz)^{n+1}]^2 = (\rho dz)^{n+1} (C^2 \rho dz)^n - \Delta\tau (\rho dz)^{n+1} [\rho dz \kappa (\partial_z C)^2]^n. \quad (35.39)$$

We are thus led to identify the dissipation operator for vertical diffusion

$$\Sigma_{\text{vert-diff}} = (\rho dz)^{n+1} (\rho dz)^n \frac{[\kappa (\partial_z C)^2]^n}{\Delta\tau}. \quad (35.40)$$

The dissipation operator $\Sigma_{\text{vert-diff}}$ has the same dimensions as Σ_{general} defined for advection in equation (35.29). Importantly, Σ_{general} accounts for dissipation from the three dimensional advection operator, whereas $\Sigma_{\text{vert-diff}}$ accounts for dissipation just from vertical diffusion. That is, the operator Σ_{general} is unable to generally isolate advection induced dissipation associated any particular direction, with the most physically relevant direction being the dianeutral. Furthermore, many of the more promising advection operators are three dimensional, and so we cannot isolate any one of the directions to ascribe a particular effective diffusivity.

Nonetheless, the following are two notable cases where the dissipation operators from advection and diffusion can be directly compared.

- One-dimensional advection-diffusion, which is a rather trivial case, but very useful for prototype development;
- A three-dimensional simulation with just advection and vertical diffusion, using a linear equation of state with density directly proportional to temperature; in this case, the dissipation operators Σ_{general} and $\Sigma_{\text{vert-diff}}$ for temperature are directly comparable.

DIANEUTRAL TRANSPORT AND ASSOCIATED BUDGETS

Contents

36.1	Introduction to the diagnostic methods	458
36.1.1	Two analysis methods	458
36.1.2	A comment on terminology	459
36.2	Density layer mass budgets and watermass formation	459
36.2.1	Mass budget within a layer	461
36.2.2	Watermass transformation and formation	461
36.2.3	Accumulating the formation from the bottom to an arbitrary density	462
36.2.4	Meridional overturning streamfunction	463
36.3	Pieces required to locally compute dianeutral transport	464
36.3.1	Neutral tangent plane and neutral density	464
36.3.2	The dia-surface velocity component	465
36.3.3	Resolved and parameterized tracer advection	469
36.4	The dianeutral transport	470
36.5	Layer calculation of the watermass transformation $\mathcal{G}(\gamma)$	472
36.5.1	An expression for $\mathcal{G}(\gamma)$ via Leibniz's Rule	472
36.5.2	Neutral density versus locally referenced potential density	473
36.5.3	The discrete approximation	474
36.6	Kinematic method to compute the material time derivative	475
36.6.1	Principles of the kinematic method	475
36.6.2	Finite volume considerations	475
36.6.3	Distinguishing material evolution from local time evolution	477
36.7	Process method to compute the material time derivative	477
36.7.1	Neutral diffusion, cabbeling, and thermobaricity	477
36.7.2	Dianeutral diffusion	479
36.7.3	Sources	479
36.8	Finite volume estimate of the advective-form material time derivative	480
36.8.1	A transport theorem for grid cells	480
36.8.2	Tracer and mass budgets for an interior grid cell	482
36.8.2.1	Semi-discrete flux-form expression	482
36.8.2.2	Advective-form expression	483
36.8.3	Material time derivative of locally referenced potential density for an interior cell	483

36.8.4	Tracer and mass budgets for a bottom grid cell	483
36.8.5	Material time derivative of locally referenced potential density for a bottom cell	484
36.8.6	Tracer and mass budgets for a surface grid cell	485
36.8.6.1	Kinematic formulation	485
36.8.6.2	Process formulation and the boundary layer	486
36.8.6.3	Advective form	487
36.8.6.4	Comments	487
36.8.7	Material time derivative of locally referenced potential density for a surface cell	488
36.8.7.1	Surface buoyancy forcing	489
36.8.7.2	Further comments on surface buoyancy fluxes	489
36.9	Comments on the MOM diagnostic calculation	489
36.9.1	Sampling	489
36.9.2	Accounting for time-explicit and time-implicit processes	490
36.9.2.1	Explicit plus implicit time stepping	490
36.9.2.2	What if we diagnosed vertical processes in series?	490
36.9.2.3	<i>A priori</i> and <i>a posteriori</i> diagnostics	491
36.9.3	Splitting physics into flux convergence plus thermodynamic source	492
36.9.4	Cabbeling, thermobaricity, and neutral diffusion	492
36.9.5	Concerning spurious dianeutral transport	493
36.10	Kinematic method diagnosed in MOM	493
36.10.1	Eulerian time tendency	495
36.10.2	Advection by resolved flow	496
36.10.3	Gent-McWilliams transport	496
36.10.4	Submesoscale transport	497
36.10.5	Precipitation minus evaporation: flux-form	497
36.10.6	Precipitation minus evaporation: advective-form	498
36.10.7	Liquid plus solid river runoff: flux-form	498
36.10.8	Liquid plus solid river runoff: advective-form	499
36.10.9	Liquid river runoff: flux-form	499
36.10.10	Liquid river runoff: advective-form	500
36.10.11	Solid calving land ice: flux-form	500
36.10.12	Solid calving runoff: advective-form	501
36.10.13	Summary of the kinematic method	501
36.10.13.1	Material time derivative	501
36.10.13.2	Dianeutral transport from ω -diagnostic	502
36.10.13.3	Dianeutral transport from t -form diagnostics	502
36.11	Process method diagnosed in MOM	502
36.11.1	Boundary fluxes of heat and salt through the vertical mixing operator	504
36.11.1.1	Surface boundary heat fluxes	504
36.11.1.2	Surface boundary salt fluxes	504
36.11.1.3	Net surface boundary heat and salt fluxes	505
36.11.1.4	Bottom boundary heat flux	505
36.11.1.5	Penetrative shortwave radiation	505
36.11.2	Boundary fluxes of buoyancy arising from mass transport	505
36.11.2.1	Precipitation minus evaporation	506
36.11.2.2	Liquid plus solid river runoff	507
36.11.2.3	Liquid river runoff	508
36.11.2.4	Solid calving runoff	509
36.11.3	Vertical mixing processes	509
36.11.3.1	Vertical diffusion and boundary fluxes	509

36.11.3.2	Vertical diffusion from net vertical diffusivity	510
36.11.3.3	Vertical diffusion from static background vertical diffusivity	511
36.11.3.4	Vertical diffusion from internal tide mixing vertical diffusivity	511
36.11.3.5	Vertical diffusion from coastal tide mixing vertical diffusivity	512
36.11.3.6	Vertical diffusion from leewave induced vertical diffusivity	513
36.11.3.7	Vertical diffusion from K33-implicit	514
36.11.3.8	Vertical diffusion from dianeutral mixing plus K33-implicit	514
36.11.3.9	Vertical mixing from convective adjustment schemes	515
36.11.3.10	Nonlocal KPP transport	515
36.11.3.1	Diagnostic checks for vertical processes	516
36.11.4	Neutral diffusion	516
36.11.4.1	Neutral diffusion operator: time-explicit piece	516
36.11.4.2	Cabelling in the ocean interior	517
36.11.4.3	Thermobaricity in the ocean interior	517
36.11.4.4	Diagnostic checks for neutral diffusion	517
36.11.5	Submesoscale horizontal diffusion	517
36.11.6	Quasi-physical parameterizations of overflow and marginal sea exchange	518
36.11.6.1	Over-exchange scheme	519
36.11.6.2	Overflow scheme	519
36.11.6.3	Overflow scheme from NCAR	520
36.11.6.4	Mixdownslope scheme	520
36.11.6.5	Sigma diffusion scheme	521
36.11.6.6	Cross land mixing scheme	521
36.11.6.7	Cross land insertion scheme	522
36.11.7	Miscellaneous schemes	522
36.11.7.1	Frazil heating of ocean liquid	522
36.11.7.2	Free surface or bottom pressure smoothing	523
36.11.8	Summary of the process method for the ESM2M ocean	523
36.11.8.1	Material time derivative	523
36.11.8.2	Dianeutral transport from wdian diagnostics	524
36.11.8.3	Dianeutral transport from tform diagnostics	524
36.12	Budget for locally referenced potential density	525
36.13	Diagnosing mass budgets for density layers	526
36.13.1	Time tendency for layer mass, $\mathcal{M}(\gamma)$	526
36.13.1.1	Time averaging the time tendency	526
36.13.1.2	Noise in the tendency of binned mass	527
36.13.1.3	A smoothed mass tendency	527
36.13.1.4	Estimating layer mass via interpolation	528
36.13.1.5	Binning the mass of a tracer cell to neutral density layers	529
36.13.1.6	Binning the time tendency for the mass per area (not recommended)	529
36.13.1.7	Regarding water fluxes and vertical coordinates	529
36.13.2	Surface mass transport, $\mathcal{E}(\gamma)$	530
36.13.3	Overturning streamfunction, $\Psi^+(\gamma) = -\int_{\gamma}^{\gamma_b} \nu^+$	531
36.13.4	watermass transformation, $\mathcal{G}(\gamma)$	533
36.14	Inferring transformation from surface buoyancy fluxes	534
36.14.1	Density forcing associated with surface water fluxes	534
36.14.2	Density forcing associated with surface heat and salt fluxes	535
36.15	Specifying the density classes for layer diagnostics	537
36.15.1	Online calculation of neutral density	537
36.15.2	Defining the density bins	537

36.15.3	Convention for the binning	538
36.15.4	Binning versus remapping	538
36.16	Known limitations	539
36.16.1	Disagreements between <code>wdian</code> and <code>tform</code>	539
36.16.2	Inserting river water into the ocean	539
36.16.3	Difficulty closing the mass budgets	539

The purpose of this chapter is to detail diagnostic methods available online in MOM to compute the rate that seawater is transported across locally defined potential density surfaces, along with budgets for locally referenced potential density. There are two general methods detailed: (A) The *layer method* based on extensions of the [Walín \(1982\)](#) isopycnal mass budget approach, with this method fundamentally based on balances within density layers. (B) A *local method* that calculates the dianeutral transport at each model grid cell, with this transport mapped either in the native model coordinates or binned to density layers. Each method in turn has an equivalent kinematic formulation and a process formulation. The layer and direct methods are identical in the continuum, but differ in practice due to numerical truncation.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_sbc.F90
ocean_core/ocean_bbc.F90
ocean_tracers/ocean_tracer_advect.F90
```

as well as various other modules associated with parameterized physical processes.

Caveat

Section 36.16 summarizes the many known limitations of this diagnostic scheme. Ongoing research is aimed at resolving these limitations.

36.1 Introduction to the diagnostic methods

The purpose of this chapter is to detail two diagnostic methods available in MOM to compute the mass flux crossing density surfaces. That is, we develop a diagnostic methodology that identifies and quantifies the boundary and interior physical processes contributing to watermass transformation occurring in a simulation. Quantifying and mapping such *dianeutral mass transport*, identifying the processes giving rise to such transport, and quantifying the associated budgets for locally referenced potential density, is a central piece required to answer oceanographic questions regarding the transformation of fluid between density classes.

36.1.1 Two analysis methods

To deductively obtain information about density transformation from observations is nearly impossible, given the need to have full information about ocean mixing processes. However, there is a powerful inferential method, first introduced by [Walín \(1982\)](#), that offers many useful conclusions based on surface buoyancy and mass fluxes, along with a steady state assumption and mass balance within density layers (or volume balance for Boussinesq fluids). The [Walín \(1982\)](#) method has been further explored, generalized, and applied in such studies as [Tziperman \(1986\)](#), [Speer and Tziperman \(1992\)](#), [Marshall et al. \(1999\)](#), [Viúdez \(2000\)](#), [Large and Nurser \(2001\)](#), [Maze et al. \(2009\)](#), [Iudicone et al. \(2008\)](#), and [Downes et al. \(2011\)](#). Given the far more detailed information available in an ocean model, we apply this method throughout the ocean column, thus allowing for deductive statements to be made about the causes of simulated watermass transformation from the ocean surface to bottom. We refer the [Walín \(1982\)](#) based approach as the *layer method*.

The layer method requires the specification of layer bounding surfaces that have a simply connected topology. However, [McDougall and Jackett \(1988\)](#) showed that there is no simply connected surface in the ocean whose tangents are equal to neutral directions, with neutral directions those most relevant for orienting mixing processes ([McDougall, 1987a](#)). Approximate simply connected surface are available, such as the neutral density γ^n of [Jackett and McDougall \(1997\)](#). Neutral density is useful in certain parts of the World Ocean, especially when considering limited geographical and/or density ranges. However, isosurfaces of neutral density are a poor approximation to neutral directions when water properties deviate from the climatology used to define the γ^n . Additionally, γ^n generally has troubles in the Southern Ocean, where the effects of the nonlinear equation of state are substantial. [Ludicone et al. \(2008\)](#) encountered these issues in their analysis of the Southern Ocean, and presented a partial solution whose generality and robustness are not well established.

A second limitation in practice with the layer method is that it requires the binning of transports into density classes. Binning level model information to layer space can introduce a nontrivial level of noise to the diagnostic, with this noise a function of the subjectively defined density classes. Such noise can often be overcome with sufficient sampling through spatial and temporal averaging. But it remains a definite concern for any diagnostic method aiming to be quantitatively robust.

As a means to add robustness to the diagnosed watermass transformation, and to avoid some of the limitations of the layer approach, we introduce a second analysis method. The *direct* method for diagnosing dianeutral transport computes the dianeutral transport at each point in the generalized level coordinate ocean model, rather than through formulating a layer mass balance as in the layer method. Consequently, the method is more “direct” than the layer method. Furthermore, the direct method allows one to map the dianeutral transport on the native model grid, thus avoiding problems with density binning. Nonetheless, to directly compare to the layer method, and to thus provide a quantitative measure of watermass transformation according to density classes, one may also bin according to density layers.

For those interested solely in applications of the two analyses methods, the differences in their formulation are far less interesting than the fact that two methods exist, and that they agree in the continuum. In practice, differences arise from discretization errors. Such errors are ubiquitous in numerical modelling, so that having two methods offers added robustness to any results, especially results that aim at quantitative precision. This chapter exposes the mathematical formulation as well as the numerical implementation details required to garner a full understanding of the MOM watermass transformation diagnostic

36.1.2 A comment on terminology

Consistent with the usage in the [Walín \(1982\)](#) inspired watermass transformation literature, we consider a “watermass” to be specified by its density class. The transformation of a watermass is thus associated with processes leading to a material change in the locally referenced potential density. However, a more general definition of watermass is found [Sve \(????\)](#), where on page 141 they state the following.

Watermasses can be classified on the basis of their temperature-salinity characteristics, but density cannot be used for classification because two watermasses of different temperatures and salinities may have the same density.

Changes in density arise through changes in temperature and salinity. Hence, changes in the “watermass” as defined in the density-based methods lead to watermass changes in the sense of [Sve \(????\)](#). However, the converse does not generally hold. Namely, there are physically interesting changes in temperature and salinity that leave density unchanged. A more general watermass transformation equation, that incorporates both density changing and density compensated effects, has been discussed by [McDougall \(1987b\)](#), and further explored by [Zika et al. \(2010\)](#). Nonetheless, for the purposes of the present chapter, we follow the [Walín \(1982\)](#) inspired usage, in which watermasses are altered by material changes to density.

36.2 Density layer mass budgets and watermass formation

The purpose of this section is to introduce certain of the ideas used for density based watermass transformation analysis. This discussion sets the theoretical stage for the diagnostic method detailed in later

sections. We then return in Section 36.13 to summarize the diagnostic calculation of the layer mass budget as available in MOM.

Figure 36.1 illustrates the geometry considered when formulating the mass budget for a layer of fluid labelled by a density variable γ , bounded above and below by densities $\gamma - \delta\gamma/2$ and $\gamma + \delta\gamma/2$, respectively. We may choose an arbitrary region to establish the formalism, but our particular example of the Southern Ocean is of great interest within the research community (e.g., [Iudicone et al., 2008](#)), so we consider it to help ground the formalism in a particular application.

As for discrete isopycnal models, properties within a chosen layer are assumed to be uniform. We specify a watermass according to the locally referenced potential density, γ . However, since surfaces of constant γ exhibit a nontrivial topology in an ocean with a realistic equation of state ([McDougall and Jackett, 1988](#)), we will ultimately consider the case of neutral density, following the ideas of [Iudicone et al. \(2008\)](#). For the main portion of the formulation, we discount the limitations associated with locally referenced potential density. We return to neutral density in Section 36.5.2.

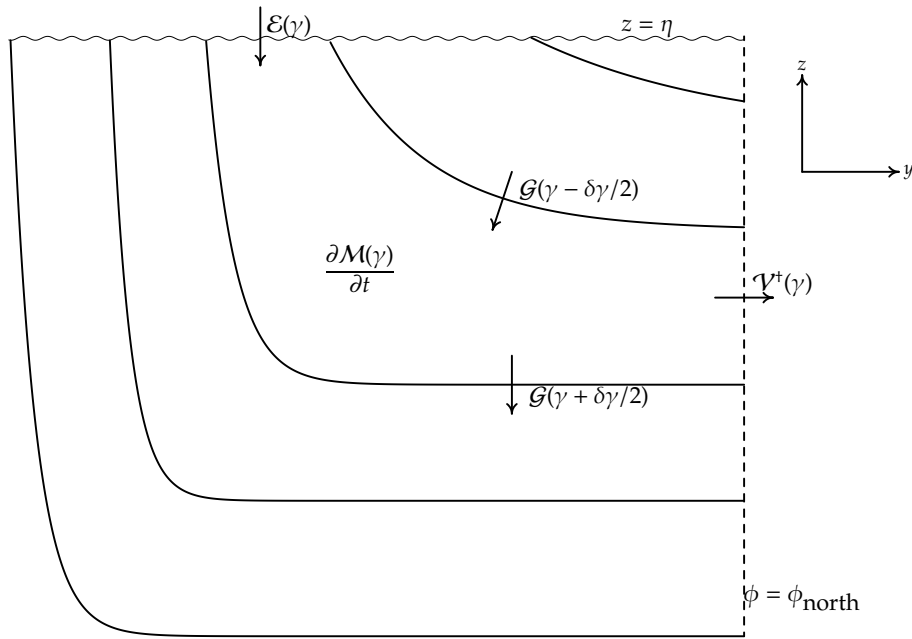


Figure 36.1: Illustration of seawater layers, oriented here according to that in the Southern Ocean where dense water outcrops to the south. We formulate a mass budget for a layer of fluid, shown here labelled by locally referenced potential density γ with layer boundaries at densities $\gamma \pm \delta\gamma/2$. The mass of this layer, $M(\gamma)$, will evolve according to mass fluxes (with units of kg s^{-1}) crossing the layer boundaries. At the ocean surface, precipitation minus evaporation plus runoff and ice melt contribute to a mass flux $\mathcal{E}(\gamma)$, with this flux considered positive for water entering the layer. At the northern boundary, water can cross the boundary at a rate $\mathcal{V}^+(\gamma)$, with $\mathcal{V}^+(\gamma) > 0$ taken for flow moving northward thus depleting mass in the layer. The transport $\mathcal{V}^+(\gamma)$ is computed as the residual mean transport, which includes the mean transport explicitly represented by the model advection, plus the quasi-Stokes transport often parameterized according to [Gent et al. \(1995\)](#) for the mesoscale or [Fox-Kemper et al. \(2008b\)](#) for the submesoscale. Finally, water can cross the layer boundaries at a rate $\mathcal{G}(\gamma - \delta\gamma/2)$ from the lighter layer, and $\mathcal{G}(\gamma + \delta\gamma/2)$ into the denser layer. We choose signs so that $\mathcal{G}(\gamma \pm \delta\gamma/2) > 0$ for water flowing from a light layer to a denser layer; i.e., for water flowing downwards in a stably stratified fluid. Convergence of fluid transport across surfaces of constant $\gamma \pm \delta\gamma/2$, associated with a nonzero transport $\mathcal{G}(\gamma \pm \delta\gamma/2)$, will inflate or deflate the constant γ layer, which in turn leads to motion of the bounding surfaces $\gamma \pm \delta\gamma/2$.

36.2.1 Mass budget within a layer

Following the conventions defined by Figure 36.1, the mass budget for a layer of fluid specified by γ is given by

$$\frac{\partial \mathcal{M}(\gamma)}{\partial t} = \mathcal{E}(\gamma) - \mathcal{V}^\dagger(\gamma) + \mathcal{G}(\gamma - \delta\gamma/2) - \mathcal{G}(\gamma + \delta\gamma/2). \quad (36.1)$$

All terms on the right hand side denote the integrated mass transports (units of mass per time) across the bounding interfaces of a particular density layer. The budget is posed over a fluid region whose bounding surfaces are dynamic. For example, a layer accumulating mass will inflate, with inflation associated with the motion of the surfaces. Each term in the mass budget (36.1) is straightforward to conceptualize; they are merely transports across the layer interfaces, or tendencies associated with transients. Nonetheless, the diagnostic calculation of each term requires care, with special care given to calculating the accumulated dianeutral mass transport, \mathcal{G} , also termed here the *watermass transformation*. Much of the formalism developed in this chapter concerns methods to diagnose \mathcal{G} , and in particular how to partition its contributions according to physical processes.

36.2.2 Watermass transformation and formation

We introduce some of the terminology of watermass transformation literature (e.g., Large and Nurser, 2001) by defining the *watermass formation rate*, $\mathcal{F}(\gamma)\delta\gamma$, given by

$$\mathcal{F}(\gamma)\delta\gamma \equiv \frac{\partial \mathcal{M}(\gamma)}{\partial t} + \mathcal{V}^\dagger(\gamma). \quad (36.2)$$

The quantity $\mathcal{F}(\gamma)$ is termed the *formation rate per unit density*. It measures the mass per time per density class of water formed within the density interval

$$\delta\gamma > 0. \quad (36.3)$$

Formation is associated with temporal changes in the mass of a layer, often termed the drift or secular term, or through mass entering or exiting through the latitudinal domain boundary. In the steady state, formation of water within a particular density class is balanced by export of water outside of the chosen domain. That is,

$$\mathcal{F}(\gamma)\delta\gamma = \mathcal{V}^\dagger(\gamma) \quad \text{steady state.} \quad (36.4)$$

Therefore, in the steady state, measurements of mass transport within a particular density class that is exiting a domain, $\mathcal{V}^\dagger(\gamma)$, is a direct reflection of formation of water with this density somewhere inside the domain. Conversely, if there is no formation of water within a particular density class, then there can be no export of that water through the domain boundaries. These statements are rather trivial conceptually. However, it is important to keep the language clear, and to precisely associate the language with a mathematical expression. Confusion can arise without such care with the language and mathematics.

As seen from the mass budget (36.1), watermass formation arises from the input of water through the ocean surface, or passage of water across the boundaries of the layer

$$\mathcal{F}(\gamma)\delta\gamma = \mathcal{E}(\gamma) - \delta\gamma \left(\frac{\partial \mathcal{G}}{\partial \gamma} \right). \quad (36.5)$$

In this equation, we introduced the following approximation for the convergence of mass crossing γ surfaces

$$-\frac{\partial \mathcal{G}}{\partial \gamma} \approx - \left(\frac{\mathcal{G}(\gamma + \delta\gamma/2) - \mathcal{G}(\gamma - \delta\gamma/2)}{\delta\gamma} \right). \quad (36.6)$$

We term the cross layer transport $\mathcal{G}(\gamma)$ the rate of *watermass transformation* from one density class to another. Through the relation (36.5), there is a nonzero watermass formation rate $\mathcal{F}(\gamma)\delta\gamma$ for a particular density class so long as there is a nonzero mass flux entering the ocean through the upper boundary, as measured by $\mathcal{E}(\gamma)$, or a nonzero convergence of the watermass transformation into the density layer, as measured by $-\partial\mathcal{G}/\partial\gamma$.

36.2.3 Accumulating the formation from the bottom to an arbitrary density

Rather than considering the watermass formation rate for a single density class, $\mathcal{F}(\gamma)\delta\gamma$, it is generally more convenient, and less noisy when diagnosing the formation in a level model, to perform a vertical integral from the ocean bottom up to a particular density class in a manner that is directly analogous to the treatment of an overturning streamfunction. We thus refer to the vertically integrated watermass formation rate as the *watermass formation streamfunction*

$$\Phi(\gamma) \equiv \int_{\gamma}^{\gamma_b} \mathcal{F}(\gamma) \delta\gamma. \quad (36.7)$$

The integration limits start from an arbitrary density γ and then increase ($\delta\gamma > 0$) to the ocean bottom, γ_b , with $\gamma_b > \gamma$ for a stably stratified fluid. Note that $\Phi(\gamma)$ is in fact a streamfunction only in the steady state case where mass in the layer is time independent. We nonetheless use the common “streamfunction” terminology even in the time dependent case.

It follows from the definitions of watermass formation rate (36.2) and (36.5) that the formation streamfunction can be written in the following equivalent manners

$$\begin{aligned} \Phi(\gamma) &\equiv \int_{\gamma}^{\gamma_b} \mathcal{F}(\gamma) \delta\gamma. \\ &= \int_{\gamma}^{\gamma_b} \left(\mathcal{V}^{\dagger} + \frac{\partial \mathcal{M}}{\partial t} \right) \\ &= \int_{\gamma}^{\gamma_b} \left(-\delta\gamma \left(\frac{\partial \mathcal{G}}{\partial \gamma} \right) + \mathcal{E}(\gamma) \right) \\ &= \mathcal{G}(\gamma) - \mathcal{G}(\gamma_b) + \int_{\gamma}^{\gamma_b} \mathcal{E}. \end{aligned} \quad (36.8)$$

In order to compute the mass tendency for the layer, $\partial \mathcal{M} / \partial t$, the mass flux through the surface, \mathcal{E} , and the transformation $\mathcal{G}(\gamma) - \mathcal{G}(\gamma_b)$, we must sum the grid-point contributions for each γ layer both zonally around the domain, and meridionally from the southern boundary (e.g., the Antarctic continent) northward to the chosen latitude. This step is assumed in the following discussion.

The accumulated formation equation (36.8) results just from considerations of mass balance for a region with density greater than γ . Consequently, it has some very general implications. To help better understand the implications, and to further solidify how the language of watermass transformation maps to the mathematics, consider the following special cases.

- $\mathcal{E} = 0$ and $\mathcal{G}(\gamma_b) = 0$: Consider the case with zero mass flux through the ocean surface, so that $\mathcal{E} = 0$, and zero geothermal heating, so that $\mathcal{G}(\gamma_b) = 0$. The accumulated formation equation (36.8) reduces to

$$\Phi(\gamma) = \mathcal{G}(\gamma) \quad \text{when } \mathcal{E} = 0 \text{ and } \mathcal{G}(\gamma_b) = 0. \quad (36.9)$$

Hence, the formation of water with density greater than γ (left hand side) equals to the rate that water transfers across the γ surface (right hand side). Reference to Figure 36.1 makes this relation manifest. This situation may be familiar to those having studied watermass transformation in simulations with virtual salt fluxes, in which $\mathcal{E} = 0$, as well as with zero geothermal heating. Hence, formation equals transformation in this special case.

- $\partial \mathcal{M} / \partial t = 0$: Consider now the steady state, in which the mass within each density layer remains constant with time. Such may be a useful approximation when time averaging over many years. In

this case, the accumulated formation equation (36.8) reduces to

$$\Phi(\gamma) = \int_{\gamma}^{\gamma_b} \mathcal{V}^{\dagger} \quad \text{when } \partial\mathcal{M}/\partial t = 0. \quad (36.10)$$

Thus, in the steady state, the net formation of water with density greater than γ equals to the water leaving through the northern boundary with density greater than γ . Again, reference to Figure 36.1 makes this result quite clear. Further use of the accumulated formation equation (36.8) leads to

$$\int_{\gamma}^{\gamma_b} \mathcal{V}^{\dagger} = \mathcal{G}(\gamma) - \mathcal{G}(\gamma_b) + \int_{\gamma}^{\gamma_b} \mathcal{E}(\gamma) \quad \text{when } \partial\mathcal{M}/\partial t = 0. \quad (36.11)$$

Hence, in the steady state, mass of water leaving through the northern boundary with density greater than γ (left hand side), is balanced by water converging into the density layer across the density interfaces, plus the mass of water crossing the ocean surface into this region (right hand side).

- $\mathcal{E} = 0$, $\mathcal{G}(\gamma_b) = 0$, and $\partial\mathcal{M}/\partial t = 0$: The simplest case is the steady state with zero surface mass flux and zero geothermal heating. The accumulated formation equation (36.8) then reduces to

$$\int_{\gamma}^{\gamma_b} \mathcal{V}^{\dagger} = \mathcal{G}(\gamma) \quad \text{when } \mathcal{E} = 0, \mathcal{G}(\gamma_b) = 0, \text{ and } \partial\mathcal{M}/\partial t = 0. \quad (36.12)$$

Steady state simulations with no surface water fluxes, no geothermal heating, and no irreversible processes creating cross-density transport have zero residual mean circulation. As a corollary, for an ocean with zero interior dianeutral transport and zero boundary water fluxes, steady state circulation is restricted to density classes that intersect boundary regions with nonzero mixing. Conversely, if there is a steady state circulation for regions denser than a particular density class γ , and there is no surface mass flux $\mathcal{E}(\gamma)$ nor geothermal heating impacting this particular density, then we can infer the presence of a nonzero cross density transport $\mathcal{G}(\gamma)$.

36.2.4 Meridional overturning streamfunction

The overturning streamfunction is generally defined according to

$$\Psi^{\dagger}(y, \gamma) = - \int dx \int_{z=-H}^{z=z(\gamma)} \rho v^{\dagger}(x, y, z) dz, \quad (36.13)$$

where $z = -H(x, y)$ is the depth at the ocean bottom, $z = z(\gamma)$ is the depth of the γ surface, and v^{\dagger} is the meridional component of the residual mean velocity. The zonal integration occurs over a specified periodic or closed domain, such as the full longitudinal extent of the Southern Ocean or the region between two continents such as in the North Atlantic. $\Psi^{\dagger}(y, \gamma)$ provides a means to display the mass flux of fluid moving across a latitude line from the ocean bottom up to a chosen density layer γ_b .

The minus sign in equation (36.13) represents the most common convention used for defining an overturning streamfunction. Consider a stably stratified case in which there is a net northward movement of mass for water denser than γ . That is, $\mathcal{V}^{\dagger}(y) > 0$ in these regions, meaning that mass is leaving the northern boundary of the domain (Figure 36.1). In this case, the streamfunction $\Psi^{\dagger}(y, \gamma)$ is negative. Conversely, if water denser than γ is accumulating into the region, so that $\mathcal{V}^{\dagger}(y) < 0$ for the dense layers, then the streamfunction is positive. From this definition, we have the following relation

$$\Psi^{\dagger}(y, \gamma) = - \int_{\gamma}^{\gamma_b} \mathcal{V}^{\dagger}(y). \quad (36.14)$$

This minus sign is important to keep in mind when performing watermass transformation diagnostics.

36.3 Pieces required to locally compute dianeutral transport

We are concerned with processes that contribute to transport across a locally defined potential density surface. That is, we wish to compute the *dianeutral transport*, which in turn can be integrated over a density layer to yield $\mathcal{G}(\gamma)$ appearing in the layer mass equation (36.1). As shown in this section, such transport is directly related to material changes in the locally referenced potential density. The following represents a summary of such processes.

- Boundary (surface and bottom) fluxes of buoyancy, including penetrative shortwave radiation;
- Dianeutral mixing, as typically parameterized by vertical diffusion, is the canonical example of a process contributing to interior dianeutral transport.
- Nonlocal dianeutral mixing, which appears in certain boundary layer schemes such as KPP [Large et al. \(1994\)](#), can also be an important source for dianeutral transport.
- Neutral diffusion coupled to the ocean's nontrivial equilibrium thermodynamics, which is captured by the seawater equation of state ([IOC et al., 2010](#)), leads to cabbeling, thermobaricity, and halobaricity ([McDougall, 1987b](#)).

In the remainder of this section, we introduce the three pieces of a framework used to formulate the dianeutral transport.

36.3.1 Neutral tangent plane and neutral density

Under an infinitesimal displacement $d\mathbf{x}$, the *in situ* density changes according to

$$d\rho = \rho d\mathbf{x} \cdot \left(-\alpha \nabla \Theta + \beta \nabla S + \frac{1}{\rho c_s^2} \nabla p \right) \quad (36.15)$$

where the thermal expansion coefficient is

$$\alpha = -\frac{1}{\rho} \left(\frac{\partial \rho}{\partial \Theta} \right), \quad (36.16)$$

the haline contraction coefficient is

$$\beta = \frac{1}{\rho} \left(\frac{\partial \rho}{\partial S} \right), \quad (36.17)$$

the inverse squared sound speed is

$$c_s^{-2} = \left(\frac{\partial \rho}{\partial p} \right), \quad (36.18)$$

S is the salinity, and Θ is the conservative temperature. Under adiabatic and isohaline motions, the density change is associated just with pressure changes

$$(d\rho)_{\text{adiabatic/isohaline}} = \rho d\mathbf{x} \cdot \left(\frac{1}{\rho c_s^2} \nabla p \right). \quad (36.19)$$

Therefore, if we consider an adiabatic and isohaline displacement of a fluid parcel, the difference in density between the parcel and the surrounding environment is given by

$$\begin{aligned} d\rho - (d\rho)_{\text{adiabatic/isohaline}} &= \rho d\mathbf{x} \cdot (-\alpha \nabla \Theta + \beta \nabla S) \\ &= \rho d\mathbf{x} \cdot \hat{\mathbf{n}}(\gamma) |-\alpha \nabla \Theta + \beta \nabla S|, \end{aligned} \quad (36.20)$$

where the *dianeutral unit vector* is defined by

$$\begin{aligned} \hat{\mathbf{n}}(\gamma) &\equiv \frac{\rho_{,\Theta} \nabla \Theta + \rho_{,S} \nabla S}{|\rho_{,\Theta} \nabla \Theta + \rho_{,S} \nabla S|} \\ &= \frac{-\alpha \nabla \Theta + \beta \nabla S}{|-\alpha \nabla \Theta + \beta \nabla S|}, \end{aligned} \quad (36.21)$$

with the shorthand notation

$$\rho_{,\theta} = \frac{\partial \rho}{\partial \Theta} \quad (36.22)$$

$$\rho_{,s} = \frac{\partial \rho}{\partial S}. \quad (36.23)$$

At each point in the fluid, the accumulation of displacements that are orthogonal to $\hat{\mathbf{n}}(\gamma)$ define the *neutral tangent plane*.

Our goal is to quantify the rate that fluid moves across the neutral tangent plane, with such motion termed the *dianeutral mass transport*. Additionally, we wish to realize this goal regardless the vertical stratification, as measured by the squared buoyancy frequency

$$N^2 = -g \left(\alpha \frac{\partial \Theta}{\partial z} - \beta \frac{\partial S}{\partial z} \right). \quad (36.24)$$

We are able to realize this goal locally at each grid point in a level coordinate model, which is a powerful aspect of the diagnostic method detailed in this chapter. For many purposes, it is additionally of interest to bin the dianeutral transport according to a scalar density-like field defined so that its iso-surfaces are parallel to neutral tangent planes. However, as noted by [McDougall and Jackett \(1988\)](#), the locus of neutral tangent planes does not form a simply connected surface in the presence of a non-zero *helicity*

$$\mathcal{H} = \beta T \nabla p \cdot (\nabla S \wedge \nabla \Theta), \quad (36.25)$$

where

$$T = \beta \left(\frac{\partial(\alpha/\beta)}{\partial p} \right) \quad (36.26)$$

is the thermobaricity parameter. Non-simply connected surfaces are not very useful for binning seawater properties or transports, since there is no unique method to define this binning operation. Hence, our desire to bin dianeutral transport into density-like classes is not generally possible. Various simply connected surfaces have been proposed, with these surfaces aiming to balance, in a subjectively defined optimum sense, the incompatible desires of maintaining a simply connected topology whilst having iso-surfaces parallel the neutral tangent plane.

The standard simply connected density surface used to map seawater properties is the neutral density field, γ^n , of [Jackett and McDougall \(1997\)](#). Neutral density has become a common tool for analysis of model and observational data. However, a nontrivial limitation of the γ^n field is that it relies on a static observationally based climatology, such as that from [Levitus \(1982\)](#). This approach is not convenient for many modeling applications, such as climate change simulations, and a more general approach is under development based on work of [Klocker et al. \(2009\)](#). As an interim method, we have tested the polynomial equation for neutral density given in the appendix to [McDougall and Jackett \(2005\)](#). Although the polynomial is only an approximation to the neutral density coordinate of [Jackett and McDougall \(1997\)](#), it is more convenient to use in a simulation than the software package of [Jackett and McDougall \(1997\)](#). However, we are presently favoring the use of potential density, since the [McDougall and Jackett \(2005\)](#) can in fact be somewhat misleading (Trevor McDougall, personal communication 2011). A more general method, also based on a polynomial expression, is under development (Trevor McDougall, personal communication 2011).

36.3.2 The dia-surface velocity component

How do we measure the rate that seawater moves in an arbitrary direction, where the direction is generally a function of space and time? To help answer this question, we generalize a geometric presentation given in Section 6.7 of [Griffies \(2004\)](#), as well as Section 2.2 of [Griffies and Adcroft \(2008\)](#). The material here can also be found in [Viúdez \(2000\)](#).

The most fundamental geometric object in our considerations is a normal vector, $\hat{\mathbf{n}}$, where $\hat{\mathbf{n}}$ is generally a function of space and time. If we can measure how fluid motion is oriented according to $\hat{\mathbf{n}}$, then

we can answer the question posed above. A familiar expression for $\hat{\mathbf{n}}$ arises when we specify a simply connected smooth surface using a function $s = s(x, y, z, t)$, such as a generalized vertical coordinate (see Figure 36.2). In this case, we orient the surface with its outward normal vector,

$$\hat{\mathbf{n}}(s) = \frac{\nabla s}{|\nabla s|} \quad (36.27)$$

which points in the direction of increasing s . For many oceanographic situations, we are interested in the rate that seawater crosses surfaces of constant potential density in a stably stratified region of the ocean, in which case s is chosen as potential density. In more general cases, we may be interested in normal vectors defined only locally, such as the dianeutral unit vector $\hat{\mathbf{n}}(\gamma)$ defined by equation (36.21).

With a static normal vector, the rate that seawater moves in the direction of $\hat{\mathbf{n}}$ is proportional to $\mathbf{v} \cdot \hat{\mathbf{n}}$, where \mathbf{v} is the three dimensional velocity vector of a seawater parcel. With a more general moving normal vector, we measure the projection of the seawater velocity relative to the moving normal vector, in which case we write this projection as

$$\hat{\mathbf{n}} \cdot \Delta \mathbf{v} = \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) \quad (36.28)$$

where $\mathbf{v}^{(\text{ref})}$ is the velocity describing the motion of the moving normal vector. When the normal vector is described according to a function s as in equation (36.27), then the projection of the reference velocity $\mathbf{v}^{(\text{ref})}$ onto the normal vector is given by

$$\hat{\mathbf{n}}(s) \cdot \mathbf{v}^{(\text{ref})} = - \left(\frac{\partial s / \partial t}{|\nabla s|} \right) \quad (36.29)$$

Note that we only need to specify the normal component of the velocity $\mathbf{v}^{(\text{ref})}$ for our purposes. With the definition (36.29) we have

$$\begin{aligned} \hat{\mathbf{n}}(s) \cdot \Delta \mathbf{v} &= \frac{\mathbf{v} \cdot \nabla s + \partial_t s}{|\nabla s|} \\ &= \frac{1}{|\nabla s|} \frac{ds}{dt}, \end{aligned} \quad (36.30)$$

where ds/dt is the material time derivative of the surface. The result (36.30) is of fundamental importance for describing fluid motion in the ocean. It states that fluid moving with a velocity \mathbf{v} penetrates an arbitrary moving surface at a rate directly proportional to the material time derivative of the defining surface $s = s(x, y, z, t)$ itself, with proportionality determined by the inverse magnitude of the surface gradient.

The result (36.30) is enshrined by defining the *dia-surface velocity component* according to

$$\begin{aligned} w^{(s)} &\equiv \hat{\mathbf{n}}(s) \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) \\ &= \frac{1}{|\nabla s|} \frac{ds}{dt} \\ &= \frac{(\text{VOLUME/TIME}) \text{ FLUID PENETRATING SURFACE, IN DIRECTION OF INCREASING } s}{\text{LOCAL SURFACE AREA}}. \end{aligned} \quad (36.31)$$

In words, $w^{(s)}$ measures the volume per time of fluid penetrating a locally defined surface or tangent plane, as defined by the normal vector $\hat{\mathbf{n}}(s)$, divided by the local area on that surface. Positive $w^{(s)}$ indicates fluid moving across the surface in the direction of $\hat{\mathbf{n}}(s)$; e.g., into denser regions for the case where s is locally referenced potential density. If we introduce the following symbol for the surface area element

$$dA_{(s)} \equiv \text{LOCAL SURFACE AREA ELEMENT}, \quad (36.32)$$

then we have

$$w^{(s)} dA_{(s)} = (\text{VOLUME/TIME}) \text{ FLUID PENETRATING SURFACE, IN DIRECTION OF INCREASING } s. \quad (36.33)$$

Likewise, introducing the *in situ* density yields the dia-surface mass transport

$$\rho w^{(s)} dA_{(s)} = (\text{MASS/TIME}) \text{ FLUID PENETRATING SURFACE, IN DIRECTION OF INCREASING } s. \quad (36.34)$$

Equation (36.34) provides a definition of the mass flux of seawater penetrating a surface. Calculation of this mass flux requires knowledge of the local dia-surface velocity component, $w^{(s)}$, and the local area element, $dA_{(s)}$. We now consider these terms in some detail in order to provide practical methods for their computation in an ocean model.

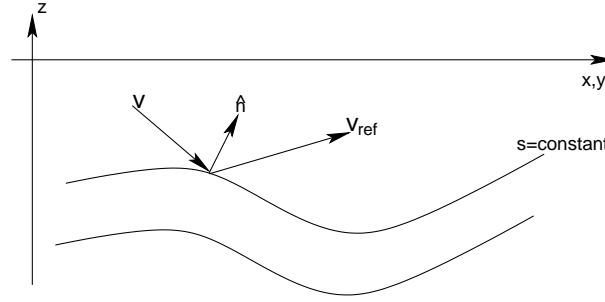


Figure 36.2: Surfaces of constant generalized vertical coordinate living interior to the ocean. A normal direction $\hat{\mathbf{n}}$ is indicated on one of the surfaces. Also shown is the orientation of the velocity of a fluid parcel \mathbf{v} and the velocity $\mathbf{v}^{(\text{ref})}$ of a reference point living on the surface. Note that the normal direction $\hat{\mathbf{n}}$ is drawn here assuming increasing values of s are parallel to the vertical direction $\hat{\mathbf{z}}$. Such is the case with many vertical coordinates used for modeling the ocean, such as $z^* = H(z - \eta)/(H + \eta)$. However, the opposite sense holds when s is taken as the locally referenced potential density γ , where stably stratified water has γ increasing downward.

For any particular orientation of the normal direction $\hat{\mathbf{n}}(s)$, it is possible to project the local area element $dA_{(s)}$ onto at least one of the three coordinate planes perpendicular to the three unit directions $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, and $\hat{\mathbf{z}}$. This projection provides the means for a practical calculation of the area element. For example, if s is a density-like field, then in the stratified ocean interior, $|\partial_z s|$ is nonzero. Indeed, it generally has the largest magnitude of the three spatial derivatives $(\partial_x s, \partial_y s, \partial_z s)$. In this case, $\hat{\mathbf{n}}(s) \approx -\hat{\mathbf{z}}$, so the area element $dA_{(s)}$ is nearly equal to

$$dA_{(z)} = dx dy, \quad (36.35)$$

which is the area element in the plane perpendicular to the vertical direction $\hat{\mathbf{z}}$. Our intuition based on this common case in the ocean interior can be made precise by an expression from differential geometry that relates two area elements. For this purpose, make use of the equation (6.58) in Griffies (2004), in which we have the exact relation

$$dA_{(s)} = |\partial_s Z| |\nabla s| dA_{(z)}. \quad (36.36)$$

In this equation, we introduced the depth of a s -surface, which has the following functional dependence

$$Z = Z(x, y, s, t). \quad (36.37)$$

We choose the capital Z to denote this depth to distinguish it from the depth z of an arbitrary position in the ocean. However, note that the inverse function is given by

$$\frac{\partial s}{\partial z} = \left(\frac{1}{\partial Z / \partial s} \right). \quad (36.38)$$

Again, for the highly stratified ocean interior with s chosen as a density-like field,

$$|\nabla s| \approx |\partial_z s| = |\partial Z / \partial s|^{-1}, \quad (36.39)$$

thus making $dA_{(s)} \approx dA_{(z)}$. The relation (36.36) holds in general, so long as the vertical derivative $\partial_z s$ remains nonzero. It thus provides for a general method to compute the area element $dA_{(s)}$ in regions of nonzero vertical stratification of s iso-surfaces.

In regions where $\partial_z s$ is tiny, such as in the surface boundary layer using s as a density-like variable, we cannot make use of expression (36.36). Instead, an alternative is needed based on assuming there is a

nontrivial stratification in at least one of the two horizontal directions. For example, consider a nontrivial zonal stratification, and let

$$X = X(y, z, s, t) \quad (36.40)$$

be the zonal position of the s isosurface, which we assume to be monotonic in a local patch. Then we may project the area element $dA_{(s)}$ onto a plane perpendicular to the $\hat{\mathbf{x}}$ direction, in which case

$$dA_{(s)} = |\partial_s X| |\nabla s| dA_{(x)} \quad (36.41)$$

where

$$dA_{(x)} = dy dz \quad (36.42)$$

is the area element in the plane perpendicular to the $\hat{\mathbf{x}}$ direction. Likewise, with a nontrivial meridional stratification, the area element can be computed using

$$dA_{(s)} = |\partial_s Y| |\nabla s| dA_{(y)} \quad (36.43)$$

where

$$dA_{(y)} = dz dx \quad (36.44)$$

is the area element in a plane perpendicular to $\hat{\mathbf{y}}$, and

$$Y = Y(x, z, s, t) \quad (36.45)$$

is the meridional position of the s isosurface. Either expression (36.41) or (36.43) are suitable when $\partial_z s$ is tiny, so long as $\partial_x s$ and $\partial_y s$ are nonzero. In general, a useful approach algorithmically is to choose the largest in magnitude from amongst the three derivatives ($\partial_x s, \partial_y s, \partial_z s$) to determine which identity (36.36), (36.41), or (36.43) to use for computing the area element $dA_{(s)}$. Implicit in this approach is that the ocean fluid has some nonzero stratification in at least one of the three coordinate directions. If this assumption is not satisfied, then the notion of dianeutral transport becomes meaningless, since we cannot determine a neutral direction, in which case the mathematical framework breaks down.

In most parts of the ocean with s a density-like vertical coordinate, the vertical derivative $\partial_z s$ is far larger than either of the horizontal derivatives, so that the constant s surface is more highly stratified in the vertical direction. However, for a hydrostatic ocean, even when $\partial_z s$ may be larger than the horizontal derivatives $\partial_x s$ and $\partial_y s$, we may still be in a situation where the slope of the constant s surfaces is nearly vertical in so far as physical processes are concerned. Indeed, for the neutral diffusion scheme detailed in Section 36.7.1, if the slope of the neutral surface is greater than a parameter $S_{\max} \sim 1/100$, the neutral diffusion scheme exponentially transitions to horizontal diffusion in recognition of the distinctly horizontal processes active in such weakly stratified regions (Treguier et al., 1997; Ferrari et al., 2008, 2010). More generally, the aspect ratio of the discrete ocean model grid generally has the vertical spacing, Δz , far smaller than horizontal spacings Δx and Δy . Consequently, the largest coordinate surface slope that is resolvable by the discrete grid is given by

$$S_{\text{resolvable}} = \frac{\Delta z}{\Delta h}, \quad (36.46)$$

where Δh is some measure of the horizontal grid spacing, such as the geometric mean

$$\Delta h = \frac{2 \Delta x \Delta y}{\Delta x + \Delta y}. \quad (36.47)$$

Consequently, for a discrete ocean model, we can consider the vertical stratification to be the largest stratification only so long as the coordinate surface slope is less than the grid aspect ratio

$$\frac{\partial_h s}{\partial_z s} < \frac{\Delta z}{\Delta h}, \quad (36.48)$$

or equivalently if

$$S_{\text{resolvable}} \partial_z s > \partial_h s. \quad (36.49)$$

The above discussion can be summarized by the following algorithmic approach to computing the dia-surface volume transport. In regions of nontrivial vertical stratification where $S_{\text{resolvable}} \partial_z s$ is larger than either of the two horizontal derivatives $\partial_x s$ and $\partial_y s$, we compute the volume transport according to

$$\begin{aligned} w^{(s)} dA_{(s)} &= \frac{1}{|\nabla s|} \frac{ds}{dt} dA_{(s)} \\ &= \left| \frac{\partial Z}{\partial s} \right| \frac{ds}{dt} dA_{(z)} \\ &= w^{(z)} dA_{(z)}, \end{aligned} \quad (36.50)$$

where we defined the dia-surface velocity component¹

$$w^{(z)} = \left| \frac{\partial Z}{\partial s} \right| \frac{ds}{dt}. \quad (36.51)$$

Likewise, in regions where $\partial_x s$ provides the largest derivative of the surface s , we compute the volume transport according to

$$\begin{aligned} w^{(s)} dA_{(s)} &= \frac{1}{|\nabla s|} \frac{ds}{dt} dA_{(s)} \\ &= \left| \frac{\partial X}{\partial s} \right| \frac{ds}{dt} dA_{(x)} \\ &= w^{(x)} dA_{(x)}, \end{aligned} \quad (36.52)$$

where

$$w^{(x)} = \left| \frac{\partial X}{\partial s} \right| \frac{ds}{dt}. \quad (36.53)$$

Finally, in regions where $\partial_y s$ provides the largest derivative, we compute the volume transport according to

$$\begin{aligned} w^{(s)} dA_{(s)} &= \frac{1}{|\nabla s|} \frac{ds}{dt} dA_{(s)} \\ &= \left| \frac{\partial Y}{\partial s} \right| \frac{ds}{dt} dA_{(y)} \\ &= w^{(y)} dA_{(y)}, \end{aligned} \quad (36.54)$$

where

$$w^{(y)} = \left| \frac{\partial Y}{\partial s} \right| \frac{ds}{dt}. \quad (36.55)$$

Table 36.1 summarizes this algorithm for the case where s is the locally referenced potential density.

36.3.3 Resolved and parameterized tracer advection

The third concept required to compute dianeutral transport concerns the appropriate velocity to use to measure the motion of fluid perpendicular to a neutral direction. In the absence of surface forcing, interior mixing, and with a simplified equilibrium thermodynamics, fluid transport occurs just through advection. In this idealized situation, advection leads to no irreversible transport. That is, water parcels are merely rearranged, and no fluid moves across a potential density surface, since potential density is materially conserved. When simulating such flow, we must acknowledge that some scales of motion will not be resolved. Since the seminal work of [Gent and McWilliams \(1990\)](#), ocean models, especially coarsely resolved

¹Note that in treatments of dianeutral velocity component, or dianeutral advection (e.g., [McDougall \(1987b\)](#) or Section 6.7 of [Griffies \(2004\)](#)), in which case s is the locally defined potential density, the absolute value operation is not applied to the vertical derivative on the right hand side of equation (36.51). In that case, $w^{(z)}$ has the opposite sign to $w^{(s)}$. We choose the convention common in the watermass transformation literature, whereby positive $w^{(z)}$ indicates downward motion.

global climate models, generally include advective tracer transport from both the resolved motions, captured by the model's resolved fluid velocity \mathbf{v} , as well as a parameterization of unresolved motion, often termed an *eddy-induced* or *quasi-Stokes* velocity \mathbf{v}^* . Consequently, it is the effective velocity, also termed the *residual mean velocity*,

$$\mathbf{v}^\dagger = \mathbf{v} + \mathbf{v}^* \quad (36.56)$$

that is relevant for considerations of dianeutral transport. Namely, the relevant fluid velocity must include both the resolved velocity of any particular model simulation, \mathbf{v} , as well as any parameterized subgrid scale velocity, \mathbf{v}^* such as arises from [Gent et al. \(1995\)](#) or [Fox-Kemper et al. \(2008b\)](#). Consequently, the material time derivative takes the form

$$\frac{d^\dagger}{dt} = \frac{\partial}{\partial t} + \mathbf{v}^\dagger \cdot \nabla. \quad (36.57)$$

We now expose some general properties of the quasi-Stokes velocity \mathbf{v}^* that will be useful in the following. We are in particular interested in a formulation of the dianeutral velocity component that is appropriate for both Boussinesq and non-Boussinesq fluids. As the [Gent et al. \(1995\)](#) paper focuses on Boussinesq fluids, we make a few comments here concerning its non-Boussinesq generalization. These points apply to any other quasi-Stokes velocity, such as that proposed by [Fox-Kemper et al. \(2008b\)](#) to parameterize submesoscale transport. For non-Boussinesq fluids, the parameterized advection velocity \mathbf{v}^* is assumed to satisfy the following constraint

$$\nabla \cdot (\rho \mathbf{v}^*) = 0 \quad \text{non-Boussinesq}, \quad (36.58)$$

which ensures that seawater mass locally remains unaffected by the parameterization. This condition ensures the existence of a vector streamfunction so that

$$\rho \mathbf{v}^* = \nabla \wedge (\rho \Psi) \quad \text{non-Boussinesq}. \quad (36.59)$$

For Boussinesq models, the constraint (36.58) reduces to the familiar non-divergence condition

$$\nabla \cdot \mathbf{v}^* = 0 \quad \text{Boussinesq}, \quad (36.60)$$

so that volume is locally unaffected, and the relation (36.59) reduces to

$$\mathbf{v}^* = \nabla \wedge \Psi \quad \text{Boussinesq}. \quad (36.61)$$

Notably, it is the vector streamfunction that is often employed in ocean models via the use of skew diffusive tracer fluxes rather than advection fluxes ([Griffies, 1998](#)). Regardless whether one uses a skew diffusive or an advection formulation of [Gent et al. \(1995\)](#) (or any other scheme parameterized as an eddy advection such as [Fox-Kemper et al. \(2008b\)](#)), for quantifying dianeutral transport, the quasi-Stokes transport should be considered within a modified material time derivative operator (36.57).

36.4 The dianeutral transport

We now bring together the ideas from Section 36.3 to render the following definition of dianeutral velocity component

$$w^{(\gamma)} \equiv \hat{\mathbf{n}}(\gamma) \cdot (\mathbf{v}^\dagger - \mathbf{v}^{(\text{ref})}) \quad (36.62)$$

$$= \frac{(\text{VOLUME/TIME}) \text{ FLUID THROUGH } \gamma\text{-SURFACE IN DIRECTION OF INCREASING } \gamma}{\text{LOCAL SURFACE AREA}}. \quad (36.63)$$

In this equation $\mathbf{v}^{(\text{ref})}$ is the velocity of a point taken on a locally defined potential density surface, whose normal projection is given by

$$\begin{aligned} \hat{\mathbf{n}}(\gamma) \cdot \mathbf{v}^{(\text{ref})} &= - \left(\frac{\rho_{,\Theta} \partial_t \Theta + \rho_{,S} \partial_t S}{|\rho_{,\Theta} \nabla \Theta + \rho_{,S} \nabla S|} \right) \\ &= - \left(\frac{-\alpha \partial_t \Theta + \beta \partial_t S}{|-\alpha \nabla \Theta + \beta \nabla S|} \right), \end{aligned} \quad (36.64)$$

with $\hat{\mathbf{n}}(\gamma)$ is the dianeutral unit vector given by equation (36.21) that points in the direction of increasing locally referenced potential density. Inserting these expressions into the definition (36.63) renders

$$\begin{aligned} w^{(\gamma)} &= \frac{1}{|\rho_{,\Theta} \nabla \Theta + \rho_{,S} \nabla S|} \left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right) \\ &= \frac{1}{|-\alpha \nabla \Theta + \beta \nabla S|} \left(-\alpha \frac{d^+ \Theta}{dt} + \beta \frac{d^+ S}{dt} \right), \end{aligned} \quad (36.65)$$

where again the material time derivative

$$\frac{d^+}{dt} = \frac{\partial}{\partial t} + \mathbf{v}^\dagger \cdot \nabla \quad (36.66)$$

is determined by the effective or residual mean velocity

$$\mathbf{v}^\dagger = \mathbf{v} + \mathbf{v}^* \quad (36.67)$$

according to equation (36.57), with \mathbf{v}^* determined by a SGS parameterization, such as Gent et al. (1995) and/or Fox-Kemper et al. (2008b). Using the results from Section 36.3.2, we arrive at the algorithm summarized in Table 36.1 to compute the dianeutral volume and mass transports, as a function of which spatial direction provides the largest stratification.

LARGEST DERIVATIVE	VOLUME TRANSPORT	W-EXPRESSION	AREA ELEMENT
$ -\alpha \partial_x \Theta + \beta \partial_x S $	$w^{(\gamma)} dA_{(\gamma)} = w^{(x)} dA_{(x)}$	$w^{(x)} = -\alpha \partial_x \Theta + \beta \partial_x S ^{-1} (-\alpha d^+ \Theta/dt + \beta d^+ S/dt)$	$dA_{(x)} = dy dz$
$ -\alpha \partial_y \Theta + \beta \partial_y S $	$w^{(\gamma)} dA_{(\gamma)} = w^{(y)} dA_{(y)}$	$w^{(y)} = -\alpha \partial_y \Theta + \beta \partial_y S ^{-1} (-\alpha d^+ \Theta/dt + \beta d^+ S/dt)$	$dA_{(y)} = dz dx$
$S_{\text{resolvable}} -\alpha \partial_z \Theta + \beta \partial_z S $	$w^{(\gamma)} dA_{(\gamma)} = w^{(z)} dA_{(z)}$	$w^{(z)} = -\alpha \partial_z \Theta + \beta \partial_z S ^{-1} (-\alpha d^+ \Theta/dt + \beta d^+ S/dt)$	$dA_{(z)} = dx dy$

Table 36.1: A summary of the algorithm proposed to compute the dianeutral volume transport $w^{(\gamma)} dA_{(\gamma)}$ as a function of the magnitude of the spatial derivative of the locally referenced potential density. We choose to use the expression listed according to the magnitude of the derivatives in order to ensure an accurate calculation in regions where stratification is absent in a particular direction. The slope $S_{\text{resolvable}}$ is set according to the model grid aspect ratio (equation (36.46)). The material time derivative is $d^+/dt = \partial_t + \mathbf{v}^\dagger \cdot \nabla$, as given by equation (36.57). Note that the volume transport $w^{(\gamma)} dA_{(\gamma)}$ is trivially converted to a mass transport $\rho w^{(\gamma)} dA_{(\gamma)}$ through multiplication by the *in situ* density ρ .

Calculation of the material time derivative

$$\frac{\partial \rho}{\partial \Theta} \frac{d^+ \Theta}{dt} + \frac{\partial \rho}{\partial S} \frac{d^+ S}{dt} = -\alpha \left(\rho \frac{d^+ \Theta}{dt} \right) + \beta \left(\rho \frac{d^+ S}{dt} \right) \quad (36.68)$$

is fundamental to the calculation of dianeutral transport. We consider two independent methods to evaluate the material time derivatives. The *kinematic method* evaluates the material time derivative by diagnosing the Eulerian time derivative plus the residual mean transport. This method provides a straightforward means to answer the question: *What is the dianeutral transport?* Its calculation in MOM is detailed in Section 36.6. The second method, termed the *process method*, allows us to answer the question: *What physical processes and boundary fluxes cause the dianeutral transport?* This method is detailed in Section 36.7, and it amounts to a diagnosis of each boundary and physical process contributing to the material evolution of locally referenced potential density.

The kinematic and process methods represent two equivalent accountings of the same material time derivative. Hence, they render the same value. In general, making such agreement manifest in a numerical simulation is nontrivial, as doing so involves many diagnostic steps and the storage of intermediate terms that are generally not considered when writing a prognostic model algorithm. MOM has the diagnostic code modifications facilitating an accounting of both the kinematic and process methods. Assuming all terms are properly saved in a particular simulation, the kinematic and process methods agree to within numerical roundoff. Given the sometimes difficult numerical issues associated with diagnosing some of the terms giving rise to dianeutral water transformation, it is prudent to exploit both methods in order to develop confidence and robustness in the results.

Once we have a diagnostic computation of $w^{(\gamma)} dA_{(\gamma)}$ according to the algorithm in Table 36.1, we may choose to remap it to its corresponding density surface, and then integrate over that density surface to obtain the watermass transformation

$$\mathcal{G}(\gamma) = \int_{A(\gamma)} \rho w^{(\gamma)} dA_{(\gamma)}. \quad (36.69)$$

We thus have all terms available for the density layer mass budget (36.1). Sections 36.6 and 36.7 detail two methods available for computing $w^{(\gamma)} dA_{(\gamma)}$, thus providing information regarding the processes giving rise to cross-density mass transport. Before doing so, we consider an alternative calculation of the watermass transformation.

36.5 Layer calculation of the watermass transformation $\mathcal{G}(\gamma)$

The purpose of this section is to detail an alternative method for computing the watermass transformation $\mathcal{G}(\gamma)$. This method is formulated in density coordinates, and so it is fundamentally a density-based or layer approach.

36.5.1 An expression for $\mathcal{G}(\gamma)$ via Leibniz's Rule

There is an expression for $\mathcal{G}(\gamma)$ that has proven fundamental to the practical calculation of the transformation rate using the layered approach pioneered by Walin (1982). The expression makes use of Leibniz's Rule from calculus, in which we write the watermass transformation rate in the form

$$\begin{aligned} \mathcal{G}(\gamma) &= \frac{\partial}{\partial \gamma} \left(\int_{\gamma_o}^{\gamma} \mathcal{G}(\sigma) d\sigma \right) \\ &= \frac{\partial}{\partial \gamma} \left(\int_{\gamma_o}^{\gamma} d\sigma \int_{A(\sigma)} \rho w^{(\sigma)} dA_{(\sigma)} \right) \end{aligned} \quad (36.70)$$

where γ_o is an arbitrary reference density, σ is a dummy variable of integration, and we substituted expression (36.69) for the watermass transformation $\mathcal{G}(\sigma)$. The integral represents the accumulation of dianeutral mass transport over the finite volume of the density layer $\gamma_o \leq \sigma \leq \gamma$. After computing the integral, we take the derivative with respect to the density γ . Now temporarily assume the vertical stratification is stable, so that (see equation (36.50))

$$\begin{aligned} w^{(\sigma)} dA_{(\sigma)} d\sigma &= w^{(z)} dA_{(z)} d\sigma \\ &= \left| \frac{\partial \sigma}{\partial z} \right| w^{(z)} dV \\ &= \left(\frac{\partial \sigma}{\partial \Theta} \frac{d^+ \Theta}{dt} + \frac{\partial \sigma}{\partial S} \frac{d^+ S}{dt} \right) dV \end{aligned} \quad (36.71)$$

where dV is the three-dimensional volume of the region within the density layer. This result then leads to the general expression, applicable for an arbitrary stratification,

$$\mathcal{G}(\gamma) = \frac{\partial}{\partial \gamma} \left(\int_{V(\gamma, \gamma_o)} dV \rho \frac{d^+ \sigma}{dt} \right), \quad (36.72)$$

where $V(\gamma, \gamma_o)$ is the volume of the density layer bounded by γ and γ_o , and we introduced the shorthand

$$\frac{d^+ \sigma}{dt} = \frac{\partial \sigma}{\partial \Theta} \frac{d^+ \Theta}{dt} + \frac{\partial \sigma}{\partial S} \frac{d^+ S}{dt} \quad (36.73)$$

for the material derivative of the locally referenced potential density. For a vertically stratified fluid, this result takes the form

$$\mathcal{G}(\gamma) = \frac{\partial}{\partial \gamma} \left(\int dx dy \int_{z(\gamma_o)}^{z(\gamma)} dz \rho \frac{d^+ \sigma}{dt} \right) \quad \text{vertically stable stratification,} \quad (36.74)$$

where $z(\gamma)$ is the depth of the γ density surface.

We next develop a finite difference approximation, in which the right hand side of equation (36.72) takes the form (see also equation (4) in [Maze et al. \(2009\)](#))

$$\begin{aligned} \mathcal{G}(\gamma) &= \frac{\partial}{\partial \gamma} \left(\int_{V(\gamma, \gamma_o)} dV \rho \frac{d^+ \sigma}{dt} \right) \\ &\approx \frac{1}{\delta \gamma} \left(\int_{V(\gamma + \delta \gamma/2, \gamma_o)} dV \rho \frac{d^+ \sigma}{dt} - \int_{V(\gamma - \delta \gamma/2, \gamma_o)} dV \rho \frac{d^+ \sigma}{dt} \right) \\ &= \frac{1}{\delta \gamma} \left(\int_{V(\gamma + \delta \gamma/2, \gamma - \delta \gamma/2)} dV \rho \frac{d^+ \sigma}{dt} \right). \end{aligned} \quad (36.75)$$

Notice how the reference density γ_o dropped out, so that the final expression is an integral over the volume bounded by the two density surfaces $\gamma - \delta \gamma/2 \leq \sigma \leq \gamma + \delta \gamma/2$. The integrand is the density weighted material time derivative of the locally referenced potential density. Again, it is useful to touch base with the vertically stratified case, in which the previous result takes the form

$$\begin{aligned} \mathcal{G}(\gamma) &= \frac{\partial}{\partial \gamma} \left(\int_{V(\gamma, \gamma_o)} dV \rho \frac{d^+ \sigma}{dt} \right) \\ &= \frac{\partial}{\partial \gamma} \left(\int dx dy \int_{z(\gamma_o)}^{z(\gamma)} dz \rho \frac{d^+ \sigma}{dt} \right) \\ &\approx \frac{1}{\delta \gamma} \left(\int dx dy \int_{z(\gamma_o)}^{z(\gamma + \delta \gamma/2)} dz \rho \frac{d^+ \sigma}{dt} - \int dx dy \int_{z(\gamma_o)}^{z(\gamma - \delta \gamma/2)} dz \rho \frac{d^+ \sigma}{dt} \right) \\ &= \frac{1}{\delta \gamma} \left(\int dx dy \int_{z(\gamma - \delta \gamma/2)}^{z(\gamma + \delta \gamma/2)} dz \rho \frac{d^+ \sigma}{dt} \right). \end{aligned} \quad (36.76)$$

This result, or rather the more general version given by equation (36.75), forms the basis for the watermass transformation diagnostics appearing in such studies as [Iudicone et al. \(2008\)](#) and [Maze et al. \(2009\)](#).

36.5.2 Neutral density versus locally referenced potential density

In an attempt to exploit the watermass transformation formalism using neutral density, [Iudicone et al. \(2008\)](#) introduce the scale factor

$$\mathcal{B} \equiv \frac{|\nabla \gamma^n|}{|\rho_{,\Theta} \nabla \Theta + \rho_{,S} \nabla S|}, \quad (36.77)$$

that approximately accounts for the differences between the evolution of neutral density γ^n ([McDougall and Jackett, 2005](#)) and locally referenced potential density. Regions where \mathcal{B} deviates from unity include,

in particular, the Southern Ocean. We have had mixed results using the \mathcal{B} factor for online diagnostics. The problem arises in regions where the ratio can become unreasonably large, which is available for various physical and numerical reasons. It is for this reason that we generally apply the watermass diagnostics with the layer method based on potential density rather than neutral density.

As shown by [ludicone et al. \(2008\)](#), the watermass transformation equation (36.72) using neutral density surfaces takes on the form

$$\mathcal{G}(\gamma^n) = \frac{\partial}{\partial \gamma^n} \left(\int_{V(\gamma^n, \gamma_o)} dV \rho \mathcal{B} \frac{d^+ \gamma}{dt} \right), \quad (36.78)$$

with the finite difference approximation (36.75) given by

$$\mathcal{G}(\gamma^n) = \frac{1}{\delta \gamma^n} \left(\int_{V(\gamma^n + \delta \gamma^n / 2, \gamma^n - \delta \gamma^n / 2)} dV \rho \mathcal{B} \frac{d^+ \gamma}{dt} \right). \quad (36.79)$$

The introduction of \mathcal{B} allows for an extension of the water mass transformation formalism to neutral density framework. Unfortunately, the calculation of \mathcal{B} is not well posed. Namely, in any particular simulation there may be points or regions where \mathcal{B} can obtain very large or very small values. However, fundamental to its use is the assumption that γ and γ^n surfaces deviate only modestly at any given point. However, as the ratio of two numbers, each of which are subject to numerical truncation errors, there is no guarantee that \mathcal{B} in a numerical simulation will remain well bounded within its physically reasonable range. Indeed, what constitutes a physically reasonable range is a difficult question to answer in general. Our experience has therefore been that the online calculation of \mathcal{B} is not robust. This behaviour suggests that further research is required to bring about a more appropriate generalization of the watermass transformation formalism.

Given the above noted difficulties with \mathcal{B} , we provide various options in MOM. First, we may set \mathcal{B} to unity everywhere, which returns the calculation to the traditional locally referenced potential density approach. Second, we provide options to bound \mathcal{B} , with values ranging from 1/2 to 2 considered physically reasonable. The alternative approach from [ludicone et al. \(2008\)](#), who computed \mathcal{B} based on a climatology, is not generally appropriate for the MOM diagnostic. Namely, our aim is to use the diagnostic online, with watermass transformation fields computed each time step during a simulation.

36.5.3 The discrete approximation

Equation (36.79) provides a method to compute the net watermass transformation associated with a neutral density surface γ^n . A finite difference calculation of the volume integral takes the form

$$\begin{aligned} \mathcal{G}(\gamma^n) &= \frac{1}{\delta \gamma^n} \left(\int_{V(\gamma^n + \delta \gamma^n / 2, \gamma^n - \delta \gamma^n / 2)} dV \rho \mathcal{B} \frac{d^+ \gamma}{dt} \right) \\ &\approx \frac{1}{\delta \gamma^n} \sum_{i,j,k} dV \mathcal{B} \rho \frac{d^+ \gamma}{dt} \Pi(\gamma^n \pm \delta \gamma^n / 2). \end{aligned} \quad (36.80)$$

The dimensionless function Π acts as a binning operator that vanishes when the density at a particular i, j, k grid box is outside of the chosen density layer, and unity when inside the layer. The integrand is computed at each grid point in the level model, and then a density binning is performed to approximate the volume integral. Both sides of equation (36.80) have the physical dimensions of mass per time. The right hand side is diagnosed in MOM, where the material time derivative is expanded into its various contributions. From this diagnostic, one can then diagnose terms in the mass equation (36.1).

36.6 Kinematic method to compute the material time derivative

The purpose of this section is to introduce the kinematic method used to compute the material time derivative of locally referenced potential density. More details are provided in Section 36.8.

36.6.1 Principles of the kinematic method

The kinematic method works directly with the Eulerian time tendency and advection tendency (arising from both resolved and parameterized advection) of potential temperature and salinity in order to diagnose their material time derivatives. To unpack the terms required for this approach, we write the material time derivative d^+/dt as

$$\frac{d^+}{dt} = \frac{d}{dt} + \mathbf{v}^* \cdot \nabla. \quad (36.81)$$

Again, \mathbf{v}^* is a parameterized eddy-induced transport velocity that satisfies the non-divergence constraint (36.58) for a non-Boussinesq fluid, or the constraint (36.60) for a Boussinesq fluid. Expanding the derivatives leads to

$$\begin{aligned} \rho_{,\Theta} \left(\rho \frac{d^+\Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+S}{dt} \right) &= \rho_{,\Theta} \left(\rho \frac{d\Theta}{dt} + \rho \mathbf{v}^* \cdot \nabla \Theta \right) + \rho_{,S} \left(\rho \frac{dS}{dt} + \rho \mathbf{v}^* \cdot \nabla S \right) \\ &= \rho_{,\Theta} \left(\rho \frac{d\Theta}{dt} + \nabla \cdot (\rho \mathbf{v}^* \Theta) \right) + \rho_{,S} \left(\rho \frac{dS}{dt} + \nabla \cdot (\rho \mathbf{v}^* S) \right) \end{aligned} \quad (36.82)$$

where we used the non-divergence condition (36.58) for the last step. We now expand the material time derivative acting on Θ and S . For this purpose, we expose a detail concerning the model's generalized level coordinates, in which a smooth function $s = s(x, y, z, t)$ defines surfaces of constant generalized vertical coordinate.² In this case, we choose to write the material time derivative from the resolved motions in the form

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla_s + w^{(s)} \partial_z \quad (36.83)$$

where the time and horizontal derivative operators are taken on surfaces of constant s , \mathbf{u} is the horizontal velocity component, and $w^{(s)}$ is the model's dia-surface velocity component defined according to equation (36.31). Furthermore, we note that the generalized vertical coordinate form of mass conservation is given by

$$\partial_t(z_{,s} \rho) + \nabla_s \cdot (z_{,s} \rho \mathbf{u}) + \partial_s(\rho w^{(s)}) = 0. \quad (36.84)$$

These results then lead to the following form for the *in situ* density weighted material time derivative of locally referenced potential density

$$\begin{aligned} \rho_{,\Theta} \left(\rho \frac{d^+\Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+S}{dt} \right) &= \rho_{,\Theta} \left(\rho \frac{d\Theta}{dt} + \nabla \cdot (\rho \mathbf{v}^* \Theta) \right) + \rho_{,S} \left(\rho \frac{dS}{dt} + \nabla \cdot (\rho \mathbf{v}^* S) \right) \\ &= \rho_{,\Theta} \left(\partial_t(z_{,s} \rho \Theta) + \nabla_s \cdot (z_{,s} \rho \Theta \mathbf{u}) + \partial_s(\rho \Theta w^{(s)}) + \nabla \cdot (\rho \mathbf{v}^* \Theta) \right) \\ &\quad + \rho_{,S} \left(\partial_t(z_{,s} \rho S) + \nabla_s \cdot (z_{,s} \rho S \mathbf{u}) + \partial_s(\rho S w^{(s)}) + \nabla \cdot (\rho \mathbf{v}^* S) \right). \end{aligned} \quad (36.85)$$

The second equality writes the material derivative as the sum of two flux-form derivatives, each weighted by their respective density partial derivatives.

36.6.2 Finite volume considerations

In developing diagnostic methods to measure dianeutral transport, it is useful to closely follow the numerical methods used to time step the prognostic model variables. For this purpose, we briefly recall salient elements of the tracer budget as discretized in MOM. More details are provided in Section 36.8.

²Be careful not to confuse the general vertical coordinate s with the salinity S .

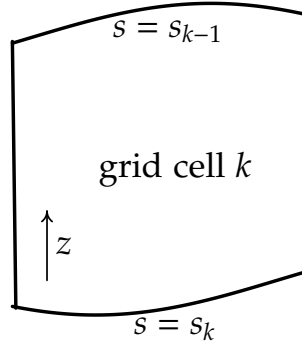


Figure 36.3: Schematic of an ocean grid cell labeled by the vertical integer k . Its sides are vertical and oriented according to $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$, and its horizontal position is fixed in time. The top and bottom surfaces are determined by constant generalized vertical coordinates s_{k-1} and s_k , respectively. Furthermore, the top and bottom are assumed to always have an outward normal with a nonzero component in the vertical direction $\hat{\mathbf{z}}$. That is, the top and bottom are never vertical. We take the convention that the discrete vertical label k increases as moving downward in the column, and grid cell k is bounded at its upper face by $s = s_{k-1}$ and lower face by $s = s_k$.

Grid cells in MOM are bounded on the sides by surfaces of constant horizontal coordinates (x, y) , and the vertical by surfaces of constant generalized vertical coordinate s (see Figure 36.3). The continuous expression for the material time derivative of tracer

$$\rho \frac{d^+ C}{dt} = \rho \frac{dC}{dt} + \nabla \cdot (\rho \mathbf{v}^* C) \quad (36.86)$$

takes the following semi-continuous form when integrated vertically over the extent of a grid cell

$$\int_{z_k}^{z_{k-1}} \left(\rho \frac{d^+ C}{dt} \right) dz = \partial_t (\rho C dz) + \nabla_s \cdot (\rho C \mathbf{u}^+ dz) + \Delta_k (\rho C w^{(s)+}). \quad (36.87)$$

In equation (36.87), C is the tracer concentration (tracer mass per seawater mass) within the grid cell; ρdz is the seawater mass per horizontal area; $dz \rho C \mathbf{u}^+$ is the horizontal flux of tracer mass associated with residual mean advection with

$$\mathbf{u}^+ = \mathbf{u} + \mathbf{u}^* \quad (36.88)$$

the horizontal component of the residual mean velocity. The flux crossing the vertical interfaces of the grid cell is given by $\rho C w^{(s)+}$, and the difference of these fluxes across the top and bottom of the cell is denoted by the operator

$$\Delta_k A = A_{k-1} - A_k, \quad (36.89)$$

with k a discrete label for the vertical level. For Boussinesq vertical coordinates (e.g., geopotential, z^*), the density factors are cancelled, whereas they remain for the non-Boussinesq pressure based vertical coordinates.

It is the expression (36.87) that forms the basis for numerically integrating the tracer budgets in MOM. Therefore, we use this expression as the basis for the diagnostic calculation of the material time derivative of temperature and salinity in a model grid cell

$$\begin{aligned} \int_{z_k}^{z_{k-1}} dz \left[\rho_{,\Theta} \left(\rho \frac{d^+ \Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+ S}{dt} \right) \right] &= \rho_{,\Theta} \left(\partial_t (\rho \Theta dz) + \nabla_s \cdot (\rho \Theta \mathbf{u}^+ dz) + \Delta_k (\rho \Theta w^{(s)+}) \right) \\ &+ \rho_{,S} \left(\partial_t (\rho S dz) + \nabla_s \cdot (\rho S \mathbf{u}^+ dz) + \Delta_k (\rho S w^{(s)+}) \right). \end{aligned} \quad (36.90)$$

The result (36.90) represents a semi-discrete analog to the continuum kinematic expression (36.85). Its evaluation requires the time tendency for $(\Theta \rho dz)$ and $(S \rho dz)$, and the corresponding tendencies arising from the residual mean advection. Note that the material time derivative of locally referenced potential density does not allow for a fully finite volume expression, since $\rho_{,\Theta}$ and $\rho_{,S}$ are generally functions of space and time. We choose to evaluate these density derivatives at the grid cell center.

36.6.3 Distinguishing material evolution from local time evolution

When contemplating various test problems to ensure robustness of the diagnostics, one may encounter a conundrum that is resolved when remaining mindful of the distinction between material time evolution and local time evolution. For example, a grid cell may change its heat and/or temperature through the convergence of an advective heat flux into the cell, which in turn will change the grid cell density. But if the grid cell change arises solely from advection by the residual mean velocity, then there is no corresponding material time evolution of temperature or density in the grid cell. In this situation, there is no penetration of the moving neutral tangent plane, so that there is no dianeutral transport. This situation is fundamental to the formulation of dia-surface transport in Sections 36.3.2 and 36.4. In short, it is only when the grid cell evolution of heat and salt has an associated irreversible component will there be a nonzero material evolution, which in turn will lead to nonzero dianeutral transport.

36.7 Process method to compute the material time derivative

The second method for diagnosing the material time derivative informs us how boundary fluxes of buoyancy and physical processes individually impact material evolution, and thus dianeutral transport. We term this the *process method*, as it identifies those processes, including boundary fluxes, that give rise to irreversible mixing of water masses. In this approach, we note that the material time derivatives of conservative temperature and salinity are driven by physical processes that are parameterized either by a local flux convergence, or a source. The tracer equation then takes the continuum form

$$\rho \frac{d^+ \Theta}{dt} = -\nabla \cdot \mathbf{J}^\Theta + \rho \mathcal{S}^\Theta \quad (36.91)$$

$$\rho \frac{d^+ S}{dt} = -\nabla \cdot \mathbf{J}^S + \rho \mathcal{S}^S, \quad (36.92)$$

where again \mathbf{J} consists of SGS fluxes arising from lateral and dia-surface mixing processes. The local source/sink terms \mathcal{S}^Θ and \mathcal{S}^S impact conservative temperature and salinity in a non-flux convergence manner, with the nonlocal transport term from KPP (Large et al., 1994) an important example. These forms for the tracer equations lead to

$$\left[\rho_{,\Theta} \left(\rho \frac{d^+ \Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+ S}{dt} \right) \right] = \rho_{,\Theta} (-\nabla \cdot \mathbf{J}^\Theta + \rho \mathcal{S}^\Theta) + \rho_{,S} (-\nabla \cdot \mathbf{J}^S + \rho \mathcal{S}^S). \quad (36.93)$$

We evaluate this expression using the same convention taken for the kinematic method, in which $\rho_{,\Theta}$ and $\rho_{,S}$ are evaluated at a grid cell center and then multiply the finite volume integrated expressions for the flux convergence and sources, so that

$$dz \left[\rho_{,\Theta} \left(\rho \frac{d^+ \Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+ S}{dt} \right) \right] = \rho_{,\Theta} \left(\nabla_s \cdot (\mathbf{J}^\Theta dz) + \Delta_k (J_\Theta^{(s)}) \right) + \rho_{,S} \left(\nabla_s \cdot (\mathbf{J}^S dz) + \Delta_k (J_S^{(s)}) \right) + \rho dz (\rho_{,\Theta} \mathcal{S}^\Theta + \rho_{,S} \mathcal{S}^S). \quad (36.94)$$

The remaining subsections identify particular processes contributing to the right hand side.

36.7.1 Neutral diffusion, cabbeling, and thermobaricity

The diagnosis of the dianeutral transport utilizes the numerical methods of MOM, which are based on finite volume techniques applied to the scalar tracer equations. However, these techniques are not precisely

compatible with certain results arising from a continuum representation; in particular, with the manipulations required to isolate the impacts from thermobaricity and cabbeling. That is, thermobaricity and cabbeling result from manipulating the conservative temperature and salinity equations to expose the specific nature of the nonlinear equation of state and its impacts on locally referenced potential density. However, those manipulations are not respected precisely by the discrete equations. We therefore diagnose thermobaricity and cabbeling as a discrete representation of the analytical results presented in this section.

Consider the material time evolution of conservative temperature and salinity associated with neutral diffusion, and write this evolution in the form

$$\begin{aligned} \left[\rho_{,\Theta} \left(\rho \frac{d^+ \Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+ S}{dt} \right) \right]_{\text{ndiffuse}} &= -\rho_{,\Theta} \nabla \cdot \mathbf{J}^\Theta - \rho_{,S} \nabla \cdot \mathbf{J}^S \\ &= -\nabla \cdot (\rho_{,\Theta} \mathbf{J}^\Theta + \rho_{,S} \mathbf{J}^S) + \nabla \rho_{,\Theta} \cdot \mathbf{J}^\Theta + \nabla \rho_{,S} \cdot \mathbf{J}^S \\ &= \nabla \rho_{,\Theta} \cdot \mathbf{J}^\Theta + \nabla \rho_{,S} \cdot \mathbf{J}^S, \end{aligned} \quad (36.95)$$

where we set

$$\rho_{,\Theta} \mathbf{J}^\Theta + \rho_{,S} \mathbf{J}^S = 0 \quad \text{neutral diffusive fluxes.} \quad (36.96)$$

for neutral diffusion fluxes, which is a property revealed by the specific form of neutral diffusion fluxes detailed below (see, for example [Griffies et al., 1998](#)).

In the presence of neutral diffusion, the horizontal and vertical tracer flux components take the form

$$\mathbf{J}^h = -A_n \rho \nabla_n C \quad (36.97)$$

$$J^z = -A_n \rho \mathbf{S} \cdot \nabla_n C, \quad (36.98)$$

with $A_n > 0$ the neutral diffusivity, \mathbf{S} the neutral slope vector relative to the horizontal, and

$$\nabla_n = \nabla_z + \mathbf{S} \partial_z \quad (36.99)$$

the horizontal gradient operator oriented along neutral directions. Note that we are assuming for the meantime that the vertical coordinate is geopotential. However, as we will see, the final result of our manipulations will be coordinate invariant, thus facilitating evaluation with any vertical coordinate.

With the flux components (36.97) and (36.98), one may indeed show that there is no flux of locally referenced potential density, as per equation (36.96). However, in the presence of realistic seawater equilibrium thermodynamics, the non-flux terms in equation (36.95) give rise to cabbeling, thermobaricity, and halobaricity ([McDougall, 1987b](#)). A series of straightforward manipulations (e.g., Section 14.1.7 of [Griffies, 2004](#)) render

$$\mathbf{J}^\Theta \cdot \nabla \rho_{,\Theta} + \mathbf{J}^S \cdot \nabla \rho_{,S} = A_n \rho^2 \left(\mathcal{C} |\nabla_n \Theta|^2 + \mathcal{T} \nabla_n p \cdot \nabla_n \Theta \right), \quad (36.100)$$

where

$$\begin{aligned} \mathcal{T} &= \beta \partial_p \left(\frac{\alpha}{\beta} \right) \\ &= \frac{\partial \alpha}{\partial p} - \frac{\alpha}{\beta} \frac{\partial \beta}{\partial p} \\ &= -\rho^{-1} \rho_{,S} \partial_p \left(\frac{\rho_{,\Theta}}{\rho_{,S}} \right) \\ &= -\rho^{-1} \left[\rho_{,\Theta p} - \rho_{,pS} \left(\frac{\rho_{,\Theta}}{\rho_{,S}} \right) \right] \end{aligned} \quad (36.101)$$

is the thermobaricity parameter (units of inverse temperature times inverse pressure), and

$$\begin{aligned} \mathcal{C} &= \frac{\partial \alpha}{\partial \Theta} + 2 \frac{\alpha}{\beta} \frac{\partial \alpha}{\partial S} - \left(\frac{\alpha}{\beta} \right)^2 \frac{\partial \beta}{\partial S} \\ &= -\rho^{-1} \left[\rho_{,\Theta \Theta} - 2 \rho_{,\Theta S} \left(\frac{\rho_{,\Theta}}{\rho_{,S}} \right) + \rho_{,SS} \left(\frac{\rho_{,\Theta}}{\rho_{,S}} \right)^2 \right] \end{aligned} \quad (36.102)$$

is the *cabbeling* parameter (units of squared inverse temperature). Empirically, the cabbeling parameter is strictly positive for seawater, thus leading to an increase in density through cabbeling. The thermobaricity parameter is predominantly positive. However, the product $\nabla_n p \cdot \nabla_n \Theta$ need not be sign-definite, thus allowing for thermobaricity to either increase or decrease density. Use of the expression (36.100) in the material time derivative in equation (36.95) leads to the material time derivative arising from cabbeling and thermobaricity

$$\left[\rho_{,\Theta} \left(\rho \frac{d^+ \Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+ S}{dt} \right) \right]_{\text{ndiffuse}} = A_n \rho^2 \left(C |\nabla_n \Theta|^2 + T \nabla_n p \cdot \nabla_n \Theta \right) \quad \text{cabbeling and thermobaricity.} \quad (36.103)$$

The right hand side is discretized for the cabbeling and thermobaricity diagnostic in MOM.

36.7.2 Dianeutral diffusion

Dianeutral diffusion of tracer is generally parameterized using a downgradient vertical diffusive flux (see Section 7.4.3 of Griffies (2004) for discussion)

$$\mathbf{J}^\Theta = J^\Theta \hat{\mathbf{z}} \quad (36.104)$$

$$\mathbf{J}^S = J^S \hat{\mathbf{z}} \quad (36.105)$$

where

$$J^\Theta = -D^{(\Theta)} \rho \partial_z \Theta \quad (36.106)$$

$$J^S = -D^{(S)} \rho \partial_z S, \quad (36.107)$$

with eddy diffusivities $D^{(\Theta)} > 0$ and $D^{(S)} > 0$ for potential temperature and salinity, respectively. The corresponding material evolution is given by

$$\left[\rho_{,\Theta} \left(\rho \frac{d^+ \Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+ S}{dt} \right) \right]_{\text{vdiffuse}} = \rho_{,\Theta} \frac{\partial}{\partial z} \left(\rho D^{(\Theta)} \frac{\partial \Theta}{\partial z} \right) + \rho_{,S} \frac{\partial}{\partial z} \left(\rho D^{(S)} \frac{\partial S}{\partial z} \right) \quad \text{vertical diffusion.} \quad (36.108)$$

We identify the following regimes for which vertical diffusion acts to produce a nonzero dianeutral transport.

- In the ocean interior, $D \approx 10^{-6} \text{ m}^2 \text{ s}^{-1}$ at the equator beneath the region of strong vertical shear associated with the equatorial undercurrent (Gregg et al., 2003), and $D \approx 10^{-5} \text{ m}^2 \text{ s}^{-1}$ in the middle latitudes (Ledwell et al., 1993, 2011), with far larger values near rough topography and other boundary regions (Polzin et al., 1997; Naveira-Garabato et al., 2004).
- The diffusivity can be set to a very large value in gravitationally unstable regions.
- The diffusivity for conservative temperature and salinity differ in regions where double diffusive processes occur (Schmitt, 1994; Large et al., 1994).
- The upper and lower ocean boundary conditions for the vertical tracer diffusive flux are Neumann conditions, whereby the temperature and salt fluxes are specified through boundary interactions with other components of the climate system. Hence, the vertical diffusion operator serves as the conduit through which the boundary fluxes are fed into the tracer equations. There will be a correspondingly very large dianeutral velocity component in the surface grid cell, and potentially nontrivial dianeutral velocity component at the ocean bottom if geothermal heating is applied.

36.7.3 Sources

The sources appearing in equation (36.95) are evaluated in the model to determine the following dianeutral velocity component

$$\left[\rho_{,\Theta} \left(\rho \frac{d^+ \Theta}{dt} \right) + \rho_{,S} \left(\rho \frac{d^+ S}{dt} \right) \right]_{\text{sources}} = \rho \left(\rho_{,\Theta} S^\Theta + \rho_{,S} S^S \right). \quad (36.109)$$

We consider the following sources that arise in ocean climate models.

- Penetrative shortwave radiation provides a heating source S^Θ that extends over the upper 50m-100m of the ocean, depending on optical properties. [Iudicone et al. \(2008\)](#) emphasized the importance of this term for dianeutral transport within the [Walín \(1982\)](#) framework. However, in MOM, the shortwave heating is implemented as a flux convergence, as detailed in Section 17.1. So shortwave penetration is in fact not formulated as a source term.
- The mixed layer parameterization of [Large et al. \(1994\)](#) includes a nonlocal transport term, whose form appears as a source to the potential temperature and salinity equations. The importance of this term for dianeutral transport remains unexplored in the literature.
- There are potentially other source terms that may arise in any particular simulation, such as those associated with nonlocal mixing across land locked marginal seas ([Griffies et al., 2005](#)).

36.8 Finite volume estimate of the advective-form material time derivative

Diagnosis of the dianeutral transport must confront the need to estimate a material flux across a moving surface using finite volume budget equations of a quasi-Eulerian ocean model. Namely, we must estimate a material time derivative, written in advective-form, based on finite volume flux-form computations of the temperature and salinity changes. We expose here the choices made for the diagnostics available in MOM, with the following representing an outline of the approach.

- Formulate the finite-volume flux-form budgets for temperature and salinity;
- Multiply the budgets by ρ_Θ and ρ_S to obtain a finite-volume budget for locally referenced potential density;
- Expose the advective-form material time derivative of locally referenced potential density that is implied by the flux-form finite volume discretization;
- Isolate the advective-form material time derivative and identify various contributions to both the kinematic method and process method.

36.8.1 A transport theorem for grid cells

The starting point of the finite volume formulation is development of an expression for the material time derivative of tracer, ψ , evaluated over a finite size grid cell in an ocean model, with the continuum form of the material evolution written in the flux-form

$$\rho \frac{d^+ \psi}{dt} = \partial_t (\rho \psi) + \nabla \cdot (\rho \mathbf{v}^+ \psi). \quad (36.110)$$

Grid cells of concern with generalized level models, such as MOM, have static vertical side boundaries. However, the top and bottom boundaries undulate in time, with these surfaces specified by a value of the general vertical coordinate s . The manipulations in this section require methods of vector calculus, with elements reviewed in Section 6.7 of [Griffies \(2004\)](#) and Section 2.2 of [Griffies and Adcroft \(2008\)](#). The resulting equation (36.118) is a transport theorem for a grid cell, with this theorem similar to the Reynolds Transport Theorem appropriate for a volume moving with the fluid motion (e.g., [Aris, 1962](#); [Batchelor, 1967](#)). Those uninterested in the mathematical details can safely skip to the final result (36.118), as that result is quite intuitive.

The most complex part of the derivation concerns the partial time derivative on the right hand side of

equation (36.110). We integrate this term over a grid cell by introducing Cartesian coordinates (x, y, z)

$$\begin{aligned} \int dV (\rho \psi)_{,t} &= \int dx dy dz (\rho \psi)_{,t} \\ &= \int dx dy \int_{z_1}^{z_2} dz (\rho \psi)_{,t} \\ &= \int dx dy \left[-(\rho \psi)|_{z_2} \partial_t z_2 + (\rho \psi)|_{z_1} \partial_t z_1 \right] + \partial_t \left(\int dV (\rho \psi) \right). \end{aligned} \quad (36.111)$$

The second equality follows by noting that the horizontal extent of a grid cell remains static, thus allowing for the horizontal integral to be brought outside of the time derivative. In contrast, the vertical extents of the cell, $z_1(x, y, t) \leq z \leq z_2(x, y, t)$, are generally functions of space and time, which necessitates the use of Leibniz's Rule.

We next seek to relate the time derivatives of the depth of the generalized levels $\partial_t z$ to the velocity of a point on that surface. For this purpose, first relate $\partial_t z$ to time tendencies $\partial_t s$ of the surface itself, making use of the identity (see, for example, Section 6.5.4 of Griffies, 2004)

$$\left(\frac{\partial z}{\partial t} \right)_s = - \frac{\partial z}{\partial s} \frac{\partial s}{\partial t}. \quad (36.112)$$

Next, make use of the kinematics associated with dia-surface transport, as formulated in Section 6.7 of Griffies (2004) or Section 2.2 of Griffies and Adcroft (2008), to write

$$\partial_t z = \partial_s z |\nabla s| \hat{\mathbf{n}} \cdot \mathbf{v}^{(\text{ref})}, \quad (36.113)$$

where $\mathbf{v}^{(\text{ref})}$ is the three dimensional velocity vector for a point attached to the generalized level surface. As a final step, relate the area element on the generalized surface to the horizontal projection $dA = dx dy$ of that surface

$$dA_{(\hat{\mathbf{n}})} = |\partial_s z \nabla s| dA, \quad (36.114)$$

which then renders

$$\partial_t z dA = \hat{\mathbf{n}} \cdot \mathbf{v}^{(\text{ref})} dA_{(\hat{\mathbf{n}})}. \quad (36.115)$$

This equation relates the time tendency of the depth of the generalized surface to the normal component of the velocity at a point on the surface. The two are related through the ratio of the area elements. This result is now used for the top and bottom boundary terms in relation (36.111), yielding

$$\int dV (\rho \psi)_{,t} = \partial_t \left(\int \rho dV \psi \right) - \int dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{v}^{(\text{ref})} (\rho \psi). \quad (36.116)$$

Hence, the domain integrated Eulerian time tendency of the density weighted field equals the time tendency of the density weighted field integrated over the domain, minus an integral over the domain boundary associated with motion of the domain. If the domain boundary has zero velocity, as for a geopotential coordinate model with fixed vertical grid spacing, then the second term in equation (36.116) vanishes.

The next step needed for volume integrating the density weighted material time derivative in equation (36.110) involves the divergence of the density weighted field

$$\int dV \nabla \cdot (\rho \mathbf{v}^\dagger \psi) = \int dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{v}^\dagger (\rho \psi), \quad (36.117)$$

which follows from Gauss' Law. Combining this relation with equation (36.116) leads to the desired result

$$\int \rho dV \frac{d\psi}{dt} = \partial_t \left(\int \rho dV \psi \right) + \int dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot (\mathbf{v}^\dagger - \mathbf{v}^{(\text{ref})}) (\rho \psi). \quad (36.118)$$

Hence, the mass weighted grid cell integral of the material time derivative of a field is given by the time derivative of the mass weighted field integrated over the domain, plus a boundary term that accounts for the transport across the domain boundaries, with allowance made for moving domain boundaries.

In the case of a grid cell of concern here, the moving domain boundaries are only those at the top and bottom of the cell, with motion associated with time dependence of the generalized vertical coordinate s . It is for these surfaces that the reference velocity $\mathbf{v}^{(\text{ref})}$ is nonzero. Special interest arises for the ocean surface boundary, where mass and buoyancy fluxes enter or leave the ocean. The bottom boundary allows for geothermal heat fluxes to warm the deep ocean layers. Notably, the ocean bottom is kinematically simpler than the ocean top surface, since the interface between the liquid ocean bottom and solid earth is assumed to be static for our purposes. Hence, $\mathbf{v}^{(\text{ref})} = 0$ for the ocean bottom. Finally, the vertical side boundaries of all grid cells are static, which means $\mathbf{v}^{(\text{ref})} = 0$ for these surfaces as well.

36.8.2 Tracer and mass budgets for an interior grid cell

We now apply the transport theorem (36.118) to derive budgets of seawater mass and tracer mass in a grid cell, with applications then made to the semi-discrete budgets of a generalized level coordinate ocean model. We start from the general expression for the material time derivative of a tracer

$$\rho \frac{d^+ \psi}{dt} = -\nabla \cdot \mathbf{J} + \rho \mathcal{S}^\psi \quad (36.119)$$

and use the transport theorem (36.118) to render a budget for the total tracer mass within a grid cell

$$\partial_t \left(\int \psi \rho dV \right) + \int dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v}^+ - \mathbf{v}^{\text{ref}}) \rho \psi] = - \int dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot \mathbf{J} + \int \mathcal{S}^\psi \rho dV. \quad (36.120)$$

The left hand side of this tracer budget is the finite volume expression of the kinematic material time derivative of the tracer, whereas the right hand side is the process version. It is useful to keep the kinematic and process versions of the material time derivative distinct for purposes of attributing terms in the budget for locally referenced potential density. When the tracer concentration is uniform, the subgrid scale flux \mathbf{J} vanishes. Additionally, the eddy-induced velocity generally satisfies $\nabla \cdot (\rho \mathbf{v}^*) = 0$. The resulting finite domain mass budget takes the form

$$\partial_t \left(\int \rho dV \right) + \int dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v} - \mathbf{v}^{\text{ref}}) \rho] = \int \mathcal{S}^\rho \rho dV, \quad (36.121)$$

where \mathcal{S}^ρ is a mass source tendency (units of inverse time). For brevity in the following, we assume $\mathcal{S}^\rho = 0$.

36.8.2.1 Semi-discrete flux-form expression

Taking the limit as the time independent horizontal area $dx dy$ goes to zero leads, after some steps, to the semi-discrete flux-form tracer budget

$$\partial_t(dz \rho \psi) + \nabla_s \cdot (dz \rho \mathbf{u}^+ \psi) + \Delta_k(\rho w^{(s)+} \psi) = -\nabla_s \cdot (dz \mathbf{J}) - \Delta_k(J^{(s)}) + \rho dz \mathcal{S}^\psi, \quad (36.122)$$

where again

$$\Delta_k A = A_{k-1} - A_k \quad (36.123)$$

is a discrete operator acting in the vertical. In equation (36.122), $w^{(s)+}$ measures the volume flux crossing the surfaces of constant generalized vertical coordinates due to the residual-mean transport velocity, and $J^{(s)}$ is the corresponding subgrid scale tracer flux. Both $w^{(s)+}$ and $J^{(s)}$ are straightforward generalizations of their geopotential coordinate model forms. Again, the left hand side of the tracer budget (36.122) is a semi-discrete flux-form kinematic expression for the material time derivative, whereas the right hand side is flux-form process version of the material time derivative. By setting the tracer concentration in equation (36.122) to a uniform constant we recover the mass budget for a grid cell

$$\partial_t(dz \rho) + \nabla_s \cdot (dz \rho \mathbf{u}^+) + \Delta_k(\rho w^{(s)+}) = 0. \quad (36.124)$$

Since the quasi-Stokes transport satisfies $\nabla \cdot (\rho \mathbf{v}^*) = 0$, we could just as well write

$$\partial_t(dz \rho) + \nabla_s \cdot (dz \rho \mathbf{u}) + \Delta_k(\rho w^{(s)}) = 0. \quad (36.125)$$

36.8.2.2 Advective-form expression

Equations (36.122) and (36.125)-(36.124) are the semi-discrete grid cell budgets for tracer mass and sea-water mass that correspond directly to the flux-form discrete equations implemented in MOM. These equations are the basis for the diagnostic budgets developed for locally referenced potential density. However, to diagnose the dianeutral transport, we need the advective-form of the material time derivative. We thus wish to derive the semi-discrete expression for the advective-form of the material time derivative implied by the flux-form. For this purpose, expand the time and horizontal space derivatives on the left hand side of the flux-form tracer equation (36.122) to yield

$$\begin{aligned}\partial_t(dz\rho\psi) + \nabla_s \cdot (dz\rho\mathbf{u}^\dagger\psi) + \Delta_k(\rho w^{(s)\dagger}\psi) &= \rho dz \left(\partial_t\psi + \mathbf{u}^\dagger \cdot \nabla_s\psi \right) + \psi \left(\partial_t(dz\rho) + \nabla_s \cdot (dz\rho\mathbf{u}^\dagger) \right) + \Delta_k(\rho w^{(s)\dagger}\psi) \\ &= \rho dz \left(\partial_t\psi + \mathbf{u}^\dagger \cdot \nabla_s\psi \right) - \psi \Delta_k(\rho w^{(s)\dagger}) + \Delta_k(\rho w^{(s)\dagger}\psi) \\ &= \rho dz \left(\partial_t\psi + \mathbf{u}^\dagger \cdot \nabla_s\psi + \frac{\Delta_k(\rho w^{(s)\dagger}\psi) - \psi \Delta_k(\rho w^{(s)\dagger})}{dz} \right),\end{aligned}\quad (36.126)$$

where the mass balance (36.124) was used to reach the second equality. In the continuum limit, the right hand side indeed reduces to a density and thickness weighted estimate for the material time derivative. Hence, dividing the flux-form expression by ρdz yields a consistent estimate for the advective-form material time derivative in an interior grid cell.

36.8.3 Material time derivative of locally referenced potential density for an interior cell

We use the previous results to derive an expression for the material time derivative of locally referenced potential density as integrated over an interior model grid cell. The kinematic version of the material time derivative yields

$$\begin{aligned}\int dV \rho \left(\rho_{,\Theta} \frac{d^\dagger\Theta}{dt} + \rho_{,S} \frac{d^\dagger S}{dt} \right) &= dA \rho_{,\Theta} \left(\partial_t(\rho dz\Theta) + \nabla_s \cdot (\rho dz\mathbf{u}^\dagger\Theta) + (\rho w^{(s)\dagger}\Theta)_{k-1} - (\rho w^{(s)\dagger}\Theta)_k \right) \\ &\quad + dA \rho_{,S} \left(\partial_t(\rho dzS) + \nabla_s \cdot (\rho dz\mathbf{u}^\dagger S) + (\rho w^{(s)\dagger}S)_{k-1} - (\rho w^{(s)\dagger}S)_k \right).\end{aligned}\quad (36.127)$$

It is common to represent the tracer effects from a quasi-Stokes transport velocity \mathbf{v}^* from [Gent and McWilliams \(1990\)](#) as a skew tracer diffusion rather than as an advection process ([Griffies, 1998](#)). Doing so does *not* alter the association of the skew tracer transport with the kinematic form of the material time derivative. That is, just because skew diffusion might be computed along with neutral diffusion, the skew diffusion should *not* be associated with the process version of the material time derivative.

The finite-volume process version of the material time derivative is given by

$$\begin{aligned}\int dV \rho \left(\rho_{,\Theta} \frac{d^\dagger\Theta}{dt} + \rho_{,S} \frac{d^\dagger S}{dt} \right) &= dA \rho_{,\Theta} \left(-\nabla_s \cdot (dz\mathbf{J}^\Theta) - (J^{(s)\Theta})_{s=s_{k-1}} + (J^{(s)\Theta})_{s=s_k} + \rho dz\mathcal{S}^\Theta \right) \\ &\quad + dA \rho_{,S} \left(-\nabla_s \cdot (dz\mathbf{J}^S) - (J^{(s)S})_{s=s_{k-1}} + (J^{(s)S})_{s=s_k} + \rho dz\mathcal{S}^S \right).\end{aligned}\quad (36.128)$$

Again, the fluxes \mathbf{J}^Θ and \mathbf{J}^S arise from non-advective subgrid scale processes active in the ocean interior, as well as penetrative boundary fluxes from solar radiation (which contributes just to the temperature flux).

36.8.4 Tracer and mass budgets for a bottom grid cell

We now apply the tracer budget (36.120) for a grid cell adjacent to the ocean bottom, where we assume that just the bottom face of the cell abuts the solid earth boundary. The advective term vanishes

$$\int dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v}^\dagger - \mathbf{v}^{\text{ref}})\rho\psi] = 0 \quad (36.129)$$

since there is no normal flow at the static ocean bottom with $\mathbf{v}^{\text{ref}} = 0$

$$\hat{\mathbf{n}} \cdot \mathbf{v}^\dagger = 0 \implies w^{(s)\dagger} = 0 \text{ at } z = -H(x, y). \quad (36.130)$$

For the subgrid scale flux \mathbf{J} , we note that the outward normal at the bottom is given by

$$\hat{\mathbf{n}}_H = -\left(\frac{\nabla(z+H)}{|\nabla(z+H)|} \right) \quad \text{at } z = -H(x, y), \quad (36.131)$$

and the area element along the bottom is (see Section 20.13.2 of [Griffies \(2004\)](#))

$$dA_H = |\nabla(z+H)| dA, \quad (36.132)$$

with $dA = dx dy$ the horizontal area element. Hence, tracer transport across the solid earth boundary is

$$-\int dA_H \hat{\mathbf{n}}_H \cdot \mathbf{J} = \int dA (\nabla H + \hat{\mathbf{z}}) \cdot \mathbf{J}. \quad (36.133)$$

We allow for the possibility of a nonzero geothermal tracer transport, which renders

$$Q^{\text{bott}} = (\nabla H + \hat{\mathbf{z}}) \cdot \mathbf{J} = -J^{(s)\Theta}, \quad (36.134)$$

with $Q^{\text{bott}} > 0$ indicating an input of heat to the ocean through the bottom. The corresponding thickness weighted budget for a grid cell next to the ocean bottom is given by

$$\partial_t(dz \rho \psi) + \nabla_s \cdot (dz \rho \mathbf{u}^\dagger \psi) + (\rho w^{(s)\dagger} \psi)_{s=s_{kb-1}} = -\nabla_s \cdot (dz \mathbf{J}) - (J^{(s)})_{s=s_{kb-1}} + Q^{\text{bott}} \quad (36.135)$$

where kb is the number of vertical grid cells in a fluid column. The corresponding budget for mass in the bottom cell is given by

$$\partial_t(dz \rho) + \nabla_s \cdot (dz \rho \mathbf{u}) + (\rho w^{(s)})_{s=s_{kb-1}} = 0, \quad (36.136)$$

assuming zero mass flux of seawater through the ocean bottom.

As for the discussion in Section 36.8.2, we may rewrite the kinematic (left hand side) of the tracer budget (36.135) in the advective-form

$$\partial_t(dz \rho \psi) + \nabla_s \cdot (dz \rho \mathbf{u}^\dagger \psi) + \Delta_k(\rho w^{(s)\dagger} \psi) = \rho dz \left(\partial_t \psi + \mathbf{u}^\dagger \cdot \nabla_s \psi + \frac{\Delta_k(\rho w^{(s)\dagger} \psi) - \psi \Delta_k(\rho w^{(s)\dagger})}{dz} \right), \quad (36.137)$$

which follows from equation (36.126) using the bottom no-flow kinematic boundary condition (36.130). So just as for an interior cell, we estimate the advective-form of the material time derivative via division of the flux-form by ρdz .

36.8.5 Material time derivative of locally referenced potential density for a bottom cell

For a bottom ocean grid cell, the kinematic version of the material time derivative of locally referenced potential density is given by

$$\begin{aligned} \int dV \rho \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right) &= dA \rho_{,\Theta} \left(\partial_t(\rho dz \Theta) + \nabla_s \cdot (\rho dz \mathbf{u}^\dagger \Theta) + (\rho w^{(s)\dagger} \Theta)_{k-1} \right) \\ &\quad + dA \rho_{,S} \left(\partial_t(\rho dz S) + \nabla_s \cdot (\rho dz \mathbf{u}^\dagger S) + (\rho w^{(s)\dagger} S)_{k-1} \right). \end{aligned} \quad (36.138)$$

The process version of the material time derivative yields

$$\begin{aligned} \int dV \rho \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right) &= dA \rho_{,\Theta} \left(-\nabla_s \cdot (dz \mathbf{J}^\Theta) - (J^{(s)\Theta})_{s=s_{k-1}} - Q^{\text{bott}} + \rho dz S^\Theta \right) \\ &\quad + dA \rho_{,S} \left(-\nabla_s \cdot (dz \mathbf{J}^S) - (J^{(s)S})_{s=s_{k-1}} + \rho dz S^S \right). \end{aligned} \quad (36.139)$$

This expression has assumed a zero flux of salt through the ocean bottom. As noted by equation (36.137), an estimate for the advective-form of the material time derivative in a bottom grid cell is obtained through dividing the flux-form by ρdz .

36.8.6 Tracer and mass budgets for a surface grid cell

Transport across the ocean surface occurs when water parcels enter or leave the ocean domain. As water parcels generally carry tracer content (e.g., heat and salt), their cross boundary transport changes tracer content of the ocean domain. The boundary transport of matter appears as an added term in the kinematic boundary condition for mass conservation. Additionally, cross boundary matter transfer involves an irreversible mixing process to bring a parcel across the skin layer of the boundary and to incorporate the matter into the liquid ocean. Details of the mixing that occurs when matter crosses the ocean surface are not easily deduced, and such mixing is fundamental to boundary layer turbulence schemes and formulation of flux relations (e.g., bulk formulae for heat transport; gas exchange formulae for biogeochemical transport).

For the interior and bottom grid cells, there is no distinction between the flux-form and advective-form of the material time derivative. However, for surface grid cells, there is a difference, with this difference arising from the open nature of the surface boundary, in which mass is transported across the boundary. Formulating the advective-form of the material time derivative in a surface cell thus proves to be more subtle relative to the interior and bottom cells. We expose the necessary details in this section.

36.8.6.1 Kinematic formulation

For a grid cell adjacent to the ocean surface, assume that just the upper face of this cell abuts the boundary between the ocean and the atmosphere or ice. The ocean surface is a time dependent boundary $z = \eta(x, y, t)$, with an outward normal

$$\hat{\mathbf{n}}_\eta = \frac{\nabla(z - \eta)}{|\nabla(z - \eta)|} \quad \text{at } z = \eta(x, y, t), \quad (36.140)$$

pointing from the ocean surface into the overlying media. We restrict attention to those free surface undulations where $\hat{\mathbf{n}}_\eta$ has a positive projection in the vertical. That is, we do not consider overturning or breaking surface waves, which is an assumption appropriate for hydrostatic ocean models. The area element dA_η measures an infinitesimal area on the ocean surface $z = \eta$, and it is related to the horizontal area element dA via

$$dA_\eta = |\nabla(z - \eta)| dA. \quad (36.141)$$

As the ocean free surface can move, the advective transport across this surface must be measured with respect to the moving surface. We formulate the advective transport across this boundary just as for the dia-surface transport crossing a moving generalized vertical coordinate surface. For this purpose, consider the velocity of a reference point on the surface

$$\mathbf{v}^{\text{ref}} = \mathbf{u}^{\text{ref}} + \hat{\mathbf{z}} w^{\text{ref}}. \quad (36.142)$$

Since $z = \eta$ represents the vertical position of the reference point, the vertical component of the velocity for this point is given by

$$w^{\text{ref}} = (\partial_t + \mathbf{u}^{\text{ref}} \cdot \nabla) \eta \quad (36.143)$$

which then leads to

$$\mathbf{v}^{\text{ref}} \cdot \nabla(z - \eta) = \eta_{,t}. \quad (36.144)$$

Hence, the advective transport leaving the ocean surface is

$$\begin{aligned} \int_{z=\eta} dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot (\mathbf{v}^\dagger - \mathbf{v}^{\text{ref}}) \rho \psi &= \int_{z=\eta} dA (-\eta_{,t} + w - \mathbf{u} \cdot \nabla \eta) \rho \psi \\ &= - \int_{z=\eta} dA Q_m \psi. \end{aligned} \quad (36.145)$$

For the first equality, we noted that the eddy induced velocity satisfies the no normal flow condition at all ocean boundaries

$$\hat{\mathbf{n}} \cdot \mathbf{v}^* = 0 \quad \text{at boundaries.} \quad (36.146)$$

For the second equality, we made use of the surface kinematic boundary condition, which can be written in its Lagrangian form

$$\rho \left(\frac{d(z-\eta)}{dt} \right) = -Q_m \quad \text{at } z = \eta(x, y, t) \quad (36.147)$$

or Eulerian form

$$\partial_t \eta + \mathbf{u} \cdot \nabla \eta = \rho w + Q_m \quad \text{at } z = \eta(x, y, t). \quad (36.148)$$

The mass flux Q_m measures the mass per time per horizontal area of mass crossing the ocean surface, with a sign chosen so that $Q_m > 0$ represents an input of mass to the ocean domain. We can summarize the result (36.145) with the local relation

$$\begin{aligned} Q_m &= \frac{(\text{MASS/TIME}) \text{ THROUGH FREE SURFACE}}{\text{HORIZONTAL AREA UNDER FREE SURFACE}} \\ &= - \left(\frac{\rho dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot (\mathbf{v} - \mathbf{v}^{\text{ref}})}{dA} \right) \quad \text{at } z = \eta. \end{aligned} \quad (36.149)$$

36.8.6.2 Process formulation and the boundary layer

In summary, the tracer flux leaving the ocean free surface is given by

$$\int_{z=\eta} dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v}^\dagger - \mathbf{v}^{\text{ref}}) \rho \psi + \mathbf{J}] = \int_{z=\eta} dx dy (-Q_m \psi + J^{(s)}). \quad (36.150)$$

To evaluate right hand side requires, it seems, the tracer concentration and tracer flux $J^{(s)}$ precisely at the ocean surface $z = \eta$. How literal should we interpret this kinematic result? Seawater properties precisely at the ocean surface are not what an ocean model carries as its prognostic variable in its top grid cell. Instead, the model carries a bulk property averaged over the upper finite sized grid cell, which averages over many regions that occur at the microscale next to the ocean surface (e.g., Robinson, 2005). Additionally, within the surface skin region of the ocean, irreversible mixing supports the passage of a fluid parcel across the boundary, either into the ocean interior (e.g., river runoff and precipitation) or away from the ocean domain (e.g., evaporation).

To proceed, we assume there to be a boundary layer model that renders the total tracer flux passing through the ocean surface. Developing such a model is a nontrivial problem in air-sea and ice-sea interaction theory and phenomenology. For present purposes, we are not concerned with details of such models. Instead, just introduce this boundary flux in the form

$$(-Q_m \psi + J^{(s)})_{z=\eta} = -Q_m \psi_m - Q_{\text{pbl}}, \quad (36.151)$$

where ψ_m is the tracer concentration within the incoming mass flux Q_m . The first term on the right hand side of equation (36.151) represents a first order upwind represented advective transport of tracer through the surface with the water (i.e., ice melt, rivers, precipitation, evaporation). The term Q_{pbl} arises from parameterized turbulence and/or radiative fluxes within the surface planetary boundary layer, such as sensible, latent, shortwave, and longwave heating appropriate for the temperature equation. A positive value for Q_{pbl} signals tracer entering the ocean through its surface.

Using the result (36.151) in expression (36.150) leads to

$$\int_{z=\eta} dA_{(\hat{\mathbf{n}})} \hat{\mathbf{n}} \cdot [(\mathbf{v}^\dagger - \mathbf{v}^{\text{ref}}) \rho \psi + \mathbf{J}] = - \int_{z=\eta} dx dy (Q_m \psi_m + Q_{\text{pbl}}), \quad (36.152)$$

which renders the flux-form finite volume tracer budget for a surface ocean grid cell

$$\partial_t (dz \rho \psi) + \nabla_s \cdot (dz \rho \mathbf{u}^\dagger \psi) - (\rho w^{(s)\dagger} \psi)_{s=s_{k=1}} - Q_m \psi_m = -\nabla_s \cdot (dz \mathbf{J}) + (J^{(s)})_{s=s_{k=1}} + Q_{\text{pbl}}. \quad (36.153)$$

The corresponding mass budget is given by

$$\partial_t (dz \rho) + \nabla_s \cdot (dz \rho \mathbf{u}^\dagger) - (\rho w^{(s)\dagger})_{s=s_{k=1}} - Q_m = 0, \quad (36.154)$$

or the equivalent expression

$$\partial_t (dz \rho) + \nabla_s \cdot (dz \rho \mathbf{u}) - (\rho w^{(s)})_{s=s_{k=1}} - Q_m = 0, \quad (36.155)$$

which follows since $\nabla \cdot (\rho \mathbf{v}^*) = 0$.

36.8.6.3 Advective form

Following from our discussion of the interior and bottom grid cells (see equations (36.126) and (36.137)), we rewrite the finite volume flux-form of the tracer equation (36.153) in the advective-form via the following steps

$$\begin{aligned} \partial_t (dz \rho \psi) + \nabla_s \cdot (dz \rho \mathbf{u}^\dagger \psi) - (\rho w^{(s)\dagger} \psi)_{s=s_{k=1}} - Q_m \psi_m &= (\rho dz) (\partial_t + \mathbf{u}^\dagger \cdot \nabla_s) \psi - (\rho w^{(s)\dagger} \psi)_{s=s_{k=1}} - Q_m \psi_m \\ &\quad + \psi \left(\partial_t (dz \rho) + \nabla_s \cdot (dz \rho \mathbf{u}^\dagger) \right) \\ &= (\rho dz) \left(\partial_t \psi + \mathbf{u}^\dagger \cdot \nabla_s \psi + \frac{\psi (w^{(s)\dagger})_{s=s_{k=1}} - (\psi w^{(s)\dagger})_{s=s_{k=1}}}{dz} \right) \\ &\quad + Q_m (\psi - \psi_m), \end{aligned} \quad (36.156)$$

where the last step used the finite volume flux-form of the mass budget (36.154). The large bracketed term in the final step of equation (36.156) reduces to the advective-form of the material time derivative in the continuum limit. Rearrangement then leads to the identity

$$\begin{aligned} (\rho dz) \left(\partial_t \psi + \mathbf{u}^\dagger \cdot \nabla_s \psi + \frac{\psi (w^{(s)\dagger})_{s=s_{k=1}} - (\psi w^{(s)\dagger})_{s=s_{k=1}}}{dz} \right) &= \partial_t (dz \rho \psi) + \nabla_s \cdot (dz \rho \mathbf{u}^\dagger \psi) - (\rho w^{(s)\dagger} \psi)_{s=s_{k=1}} - Q_m \psi \\ &= -\nabla_s \cdot (dz \mathbf{J}) + (J^{(s)})_{s=s_{k=1}} + Q_m (\psi_m - \psi) + Q_{\text{pbl}}. \end{aligned} \quad (36.157)$$

where we used equation (36.153) to introduce the process form of the tracer budget.

The result (36.157) suggests that we define the following surface boundary flux forcing that impacts the advective-form of the material evolution of tracer

$$\Gamma_{\text{advective}}^\psi = Q_m (\psi_m - \psi) + Q_{\text{pbl}}. \quad (36.158)$$

For many numerical implementations, the value for ψ is taken as the concentration $\psi_{k=1}$ in the surface grid cell. In this case, the boundary term $Q_m (\psi_m - \psi)$ vanishes if the tracer concentration ψ_m equals $\psi_{k=1}$. That is, tracer concentration in a surface grid cell is unchanged by the passage of tracer across the surface boundary, if the tracer concentration in the boundary flux is the same as in the ocean surface grid cell. Taking $\psi_m = \psi_{k=1}$ is commonly assumed for temperature. However, for salinity it is more appropriate to assume $S_m = 0$, which leads to a change in surface ocean salinity as fresh water crosses the surface ocean boundary; e.g., precipitating water reduces ocean salinity whereas evaporating water increases salinity. Other material tracers (e.g., biogeochemical tracers) generally have concentrations in river water that are distinct from open ocean seawater.

36.8.6.4 Comments

MOM is formulated using finite volume flux-form budgets for tracers, as in the flux-form tracer equation (36.153). One key advantage of flux-form is that it facilitates tracer conservation to within numerical roundoff, whereas advective form tracer budgets cannot ensure such accuracy. Hence, the advective form of the material time derivative is typically *not* used to formulate the equations of an ocean model.

However, the advective form, as derived for the surface cell in equation (36.157), is the appropriate form to estimate the material time evolution of tracer, which in turn is what we need for the dianeutral

mass transport diagnostic. In particular, in an ocean with constant salinity, there is no watermass transformation induced by surface boundary fluxes if a boundary flux of freshwater is added with the same temperature as the sea surface temperature. However, the ocean heat content does change. An ocean model should be based on an evolution equation for heat content, as it is heat that is conserved. Yet watermass diagnostic as determined by dianeutral mass transport should be based on advective-form material changes in temperature (and salinity), which in this trivial example does not change (at least through the boundary fluxes). It is therefore the analog of $\Gamma_{\text{advective}}^{\psi}$ (equation (36.157)) that we require to deduce how surface buoyancy fluxes impact watermass transformation. We detail these considerations in Section 36.8.7.

36.8.7 Material time derivative of locally referenced potential density for a surface cell

The kinematic expression of the material time derivative of locally referenced potential density, for a surface ocean grid cell ($k = 1$), is written in the flux-form as

$$\begin{aligned} \int dV \rho \left(\rho_{,\Theta} \frac{d^{\dagger}\Theta}{dt} + \rho_{,S} \frac{d^{\dagger}S}{dt} \right) &= dA \rho_{,\Theta} \left(\partial_t (\rho dz \Theta) + \nabla_s \cdot (\rho dz \mathbf{u}^{\dagger} \Theta) - (\rho w^{(s)\dagger} \Theta)_{s=s_k} - Q_m \Theta_m \right) \\ &\quad + dA \rho_{,S} \left(\partial_t (\rho dz S) + \nabla_s \cdot (\rho dz \mathbf{u}^{\dagger} S) - (\rho w^{(s)\dagger} S)_{s=s_k} - Q_m S_m \right). \end{aligned} \quad (36.159)$$

Following the steps considered in Section 36.8.6.3 exposes the advective-form

$$\begin{aligned} \int dV \rho \left(\rho_{,\Theta} \frac{d^{\dagger}\Theta}{dt} + \rho_{,S} \frac{d^{\dagger}S}{dt} \right) &= dA \rho_{,\Theta} \left((\rho dz) \left(\partial_t \Theta + \mathbf{u}^{\dagger} \cdot \nabla_s \Theta + \frac{\Theta (w^{(s)\dagger})_{s=s_k} - (\Theta w^{(s)\dagger})_{s=s_k}}{dz} \right) + Q_m (\Theta - \Theta_m) \right) \\ &\quad + dA \rho_{,S} \left((\rho dz) \left(\partial_t S + \mathbf{u}^{\dagger} \cdot \nabla_s S + \frac{S (w^{(s)\dagger})_{s=s_k} - (S w^{(s)\dagger})_{s=s_k}}{dz} \right) + Q_m (S - S_m) \right). \end{aligned} \quad (36.160)$$

Likewise, the process version of the material time derivative for a surface grid cell ($k = 1$) is given by

$$\begin{aligned} \int dV \rho \left(\rho_{,\Theta} \frac{d^{\dagger}\Theta}{dt} + \rho_{,S} \frac{d^{\dagger}S}{dt} \right) &= dA \rho_{,\Theta} \left(-\nabla_s \cdot (dz \mathbf{J}^{\Theta}) + (J^{(s)\Theta})_{s=s_{k=1}} + Q_{\text{pbl}}^{\Theta} + \rho dz \mathcal{S}^{\Theta} \right) \\ &\quad + dA \rho_{,S} \left(-\nabla_s \cdot (dz \mathbf{J}^S) + (J^{(s)S})_{s=s_{k=1}} + Q_{\text{pbl}}^S + \rho dz \mathcal{S}^S \right). \end{aligned} \quad (36.161)$$

As in Section 36.8.6.3, we isolate the advective-form of the kinematic expression (36.160) to render the identity

$$\begin{aligned} &\rho_{,\Theta} \left((\rho dz) \left(\partial_t \Theta + \mathbf{u}^{\dagger} \cdot \nabla_s \Theta + \frac{\Theta (w^{(s)\dagger})_{s=s_{k=1}} - (\Theta w^{(s)\dagger})_{s=s_{k=1}}}{dz} \right) \right) \\ &+ \rho_{,S} \left((\rho dz) \left(\partial_t S + \mathbf{u}^{\dagger} \cdot \nabla_s S + \frac{S (w^{(s)\dagger})_{s=s_{k=1}} - (S w^{(s)\dagger})_{s=s_{k=1}}}{dz} \right) \right) \\ &= \rho_{,\Theta} \left(\partial_t (\rho dz \Theta) + \nabla_s \cdot (\rho dz \mathbf{u}^{\dagger} \Theta) - (\rho w^{(s)\dagger} \Theta)_{s=s_{k=1}} - Q_m \Theta \right) \\ &+ \rho_{,S} \left(\partial_t (\rho dz S) + \nabla_s \cdot (\rho dz \mathbf{u}^{\dagger} S) - (\rho w^{(s)\dagger} S)_{s=s_{k=1}} - Q_m S \right) \\ &= \rho_{,\Theta} \left(-\nabla_s \cdot (dz \mathbf{J}^{\Theta}) + (J^{(s)\Theta})_{s=s_{k=1}} + Q_{\text{pbl}}^{\Theta} + Q_m (\Theta_m - \Theta) + \rho dz \mathcal{S}^{\Theta} \right) \\ &+ \rho_{,S} \left(-\nabla_s \cdot (dz \mathbf{J}^S) + (J^{(s)S})_{s=s_{k=1}} + Q_{\text{pbl}}^S + Q_m (S_m - S) + \rho dz \mathcal{S}^S \right). \end{aligned} \quad (36.162)$$

Either the kinematic expression (first equality) or process expression (second equality) can be used to estimate the advective-form material time derivative.

36.8.7.1 Surface buoyancy forcing

Based on the result (36.162), we introduce the following surface mass flux associated with surface buoyancy fluxes

$$\Gamma^\rho = \left(\frac{\rho dA}{\Delta\gamma^n} \right) \left[-\alpha \left(Q_{\text{pbl}}^\Theta + (\Theta_m - \Theta) Q_m \right) + \beta \left(Q_{\text{pbl}}^S + (S_m - S) Q_m \right) \right] \quad \text{kg s}^{-1}, \quad (36.163)$$

which is directly analogous to the forcing $\Gamma_{\text{advective}}^\psi$ affecting tracer concentration according to equation (36.158). The division by $\Delta\gamma^n$ is relevant when binning the surface buoyancy flux into density classes (Section 36.15). If $\Gamma^\rho > 0$, then density increases, and the opposite occurs for $\Gamma^\rho < 0$. A special case is worth mentioning, in which we are away from sea ice, so that the salt flux vanishes $Q_{\text{pbl}}^S = 0$; the surface temperature equals to the temperature in the boundary flux, $\Theta_{k=1} = \Theta_m$; and the boundary concentration of salt is zero, $S_m = 0$. In this case, the density forcing takes the form

$$\Gamma^\rho = - \left(\frac{\rho dA}{\Delta\gamma^n} \right) \left(\alpha Q_{\text{pbl}}^\Theta + \beta S Q_m \right) \quad \Theta_{k=1} = \Theta_m, S_m = 0, Q_{\text{pbl}}^S = 0. \quad (36.164)$$

This form is more familiar (e.g., Large and Nurser, 2001) than the more general expression (36.163). As noted in Section 36.6.3, the transport of water across the ocean surface is, by itself, not sufficient for water-mass transformation to occur. There must also be mixing of the boundary flux with the ocean. Therefore, by identifying the density forcing (36.163) as a mechanism of watermass transformation, we are assuming that such mixing will occur. This assumption is generally quite good, since the upper ocean is very turbulent and boundary fluxes are readily incorporated to the ocean mixed layer.

36.8.7.2 Further comments on surface buoyancy fluxes

Consider surface water and buoyancy fluxes passing into a uniform ocean, or equivalently an ocean comprised of a single grid cell. In this case, we drop all lateral and vertical advection, as well as all subgrid scale and source terms, so that equation (36.162) for the material time derivative reduces to

$$\begin{aligned} \rho dz \left(\rho_{,\Theta} \partial_t \Theta + \rho_{,S} \partial_t S \right) &= \rho_{,\Theta} \left(\partial_t (\rho dz \Theta) - Q_m \Theta \right) + \rho_{,S} \left(\partial_t (\rho dz S) - Q_m S \right) \\ &= \rho_{,\Theta} \left(Q_m (\Theta_m - \Theta) + Q_{\text{pbl}}^\Theta \right) + \rho_{,S} \left(Q_m (S_m - S) + Q_{\text{pbl}}^S \right), \end{aligned} \quad (36.165)$$

where we cancelled the common factor of dA from each term. If there are no other physical processes impacting temperature or salinity, then

$$\partial_t (\rho \Theta dz) = Q_m \Theta_m + Q_{\text{pbl}}^\Theta \quad (36.166)$$

$$\partial_t (\rho S dz) = Q_m S_m + Q_{\text{pbl}}^S \quad (36.167)$$

$$\partial_t (\rho dz) = Q_m, \quad (36.168)$$

which makes the two expressions in equation (36.165) consistent.

36.9 Comments on the MOM diagnostic calculation

The following represent some issues regarding the diagnostic calculation of a budget for locally referenced potential density, and the associated dianeutral transport diagnostic.

36.9.1 Sampling

All of the diagnostics saved in relation to the budget of locally referenced potential density and the dianeutral transport are sampled at each model time step. Additionally, the rebinning into density layers is performed at each model time step. Sampling at each model time step ensures there are no aliasing issues, or problems related to poor sampling of seasonal and/or diurnal cycling in the upper ocean. However, there can be some noise, especially for the rebinned fields. Thus, it is best to allow for a thorough sampling of density space as arrived at through multiple seasonal cycles.

36.9.2 Accounting for time-explicit and time-implicit processes

MOM is based on a finite volume time stepping of the mass of tracer per unit area within a grid cell. There are two basic sub-steps involved in the update of a single time step. The first sub-step accumulates time-explicit contributions, with the associated time tendencies computed based on the present state of the concentration $C(\tau)$. Upon accumulating all time-explicit tendencies, the tracer concentration is updated to $C^\#(\tau + 1)$. The second sub-step updates based on the time-implicit processes, starting incrementally from the concentration $C^\#(\tau + 1)$. Time-implicit processes are generally associated with fast vertical processes, such as surface boundary fluxes and gravitationally unstable columns. Upon updating the implicit pieces yields the new tracer concentration $C(\tau + 1)$.

36.9.2.1 Explicit plus implicit time stepping

A schematic update for the explicit time stepping for tracer concentration takes the form

$$\rho dz(\tau + 1) C^\#(\tau + 1) = \rho dz(\tau) C(\tau) + \Delta\tau \left(\frac{\delta(\rho dz C)}{\delta\tau} \right)^{\text{expl}}, \quad (36.169)$$

where $\Delta\tau$ is the time step, ρdz is the mass per area of a grid cell. The term $\delta(\rho dz C)/\delta\tau$ is the time tendency associated with time-explicit processes such as advection and lateral physical parameterizations. Diagnosing the time tendencies associated with the time-explicit portion of the time step is straitforward, as these time tendencies are computed directly, and so can be saved as desired. The tracer concentration $C^\#(\tau + 1)$ results from this portion of the time stepping.

The second portion of a MOM time step of tracer involves the time-implicit sub-step, which takes the following schematic form

$$C(\tau + 1) = C^\#(\tau + 1) + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)^{\text{impl}}. \quad (36.170)$$

Combining this step with the time-explicit step (36.169) leads to the expression

$$\frac{\rho dz(\tau + 1) C(\tau + 1) - \rho dz(\tau) C(\tau)}{\Delta\tau} = \left(\frac{\delta(\rho dz C)}{\delta\tau} \right)^{\text{expl}} + \rho dz(\tau + 1) \left(\frac{\delta C}{\delta\tau} \right)^{\text{impl}}. \quad (36.171)$$

The time-implicit step (36.170) includes a single inversion of a tridiagonal solver, with the inversion wrapping together the surface boundary fluxes, the bottom boundary fluxes, and the interior vertical diffusivity. We can unambiguously identify a tendency associated with surface boundary fluxes and bottom boundary fluxes, merely by inverting the tridiagonal matrix with the diffusivities set to zero but with the boundary fluxes retained. However, tendencies from interior mixing that arise from a number of physical processes (e.g., parameterized tide mixing; shear mixing; gravitationally unstable columns) must be diagnosed in another manner. The reason is that the inversion process for the net of interior diffusion does not equal to separate inversions of the processes. Alternatively, the single inversion does not equal to an incremental update of the tracer concentration that would result from a series of inversions (see Section 36.9.2.2). However, since we know the updated tracer $C(\tau + 1)$ at the end of the single inversion, we can diagnose the individual contributions as if they were computed explicitly in time, but using the updated concentration $C(\tau + 1)$. We detail the MOM diagnostic method in Section 36.9.2.3.

36.9.2.2 What if we diagnosed vertical processes in series?

Consider the case of vertical processes in which temperature or salinity are updated implicitly in time with a surface boundary flux, a vertical diffusivity, an implicit piece of the K33 neutral diffusion tensor, and the bottom boundary flux (i.e., geothermal heating for the temperature equation). Again, MOM handles these processes using a single call to a combined time-implicit step. However, consider instead an update of vertical processes occurring in series, with each step invoking a tridiagonal inversion and thus updating tracer concentration in a piecewise or incremental manner. In that case, we would write the various time-implicit pieces on the right hand side of equation (36.170) in the form

$$C(\tau + 1) = C^\#(\tau + 1) + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{sbc}}^{\text{impl}} + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{diff.cbt}}^{\text{impl}} + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{K33}}^{\text{impl}} + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{bbc}}^{\text{impl}}. \quad (36.172)$$

Importantly, when computed in series, each of the interior mixing process acts on a different intermediate tracer state that depend on which processes occurred earlier in the series. Hence, we cannot unambiguously attribute a tendency to a separate mixing process, since the tendencies depend on the arbitrary order that the mixing processes are computed. It is for this reason that an incremental update of interior diffusive processes is not sensible, since there is no fundamental reason to apply one piece of the vertical diffusivity prior to another piece. Contributions from the surface and bottom boundary fluxes are not subject to this ambiguity, since their tendencies are determined solely from the flux itself, which is computed based on the concentration $C(\tau)$ and atmospheric conditions.

Let us nonetheless pursue the next step that would be required to assign a tendency to a process by assuming the above ordering, so that

$$C_{\text{sbc}}(\tau + 1) = C^\#(\tau + 1) + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{sbc}}^{\text{impl}} \quad (36.173)$$

$$C_{\text{diff_cbt}}(\tau + 1) = C_{\text{sbc}}(\tau + 1) + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{diff_cbt}}^{\text{impl}} \quad (36.174)$$

$$C_{\text{K33}}(\tau + 1) = C_{\text{diff_cbt}}(\tau + 1) + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{K33}}^{\text{impl}} \quad (36.175)$$

$$C_{\text{bbc}}(\tau + 1) = C_{\text{K33}}(\tau + 1) + \Delta\tau \left(\frac{\delta C}{\delta\tau} \right)_{\text{bbc}}^{\text{impl}} \quad (36.176)$$

$$C(\tau + 1) = C_{\text{bbc}}(\tau + 1). \quad (36.177)$$

Intermediate values of the tracer concentration are required to diagnose time tendencies via the following steps

$$\left(\frac{\delta C}{\delta\tau} \right)_{\text{sbc}}^{\text{impl}} = \frac{C_{\text{sbc}}(\tau + 1) - C^\#(\tau + 1)}{\Delta\tau} \quad (36.178)$$

$$\left(\frac{\delta C}{\delta\tau} \right)_{\text{diff_cbt}}^{\text{impl}} = \frac{C_{\text{diff_cbt}}(\tau + 1) - C_{\text{sbc}}(\tau + 1)}{\Delta\tau} \quad (36.179)$$

$$\left(\frac{\delta C}{\delta\tau} \right)_{\text{K33}}^{\text{impl}} = \frac{C_{\text{K33}}(\tau + 1) - C_{\text{diff_cbt}}(\tau + 1)}{\Delta\tau} \quad (36.180)$$

$$\left(\frac{\delta C}{\delta\tau} \right)_{\text{bbc}}^{\text{impl}} = \frac{C_{\text{bbc}}(\tau + 1) - C_{\text{K33}}(\tau + 1)}{\Delta\tau}. \quad (36.181)$$

36.9.2.3 *A priori and a posteriori* diagnostics

What we wish is to unravel the tendency $(\delta C / \delta\tau)^{\text{impl}}$ in equation (36.170) to identify how individual mixing processes impact this term. We take the following approach for individual mixing processes, whereby the updated tracer concentration $C(\tau + 1)$ is used to diagnose the tendencies. That is, we diagnose

$$\text{tendency} = \Delta_k (\rho_o D \partial_z C(\tau + 1)), \quad (36.182)$$

where D is the diffusivity associated with the particular process of interest. This operator is just like a time-explicit calculation of vertical diffusion. To within time truncation errors, the sum of the individually diagnosed tendencies will equal to that arising from the time-implicit tendency.

Based on the above considerations, the following diagnostic suite has been implemented in MOM.

- Bottom and surface flux tendency contributions are split from mixing contributions. This split is unambiguous, since the tendencies from the boundary fluxes are due solely to the boundary fluxes. We diagnose these contributions by calling the tridiagonal inversion with just the boundary fluxes. This call to the tridiagonal solver is trivial, since with zero interior mixing, there is no inversion required.
- The contribution from the `diff_cbt` vertical diffusivity is computed based on the diagnostic equation (36.182).

- The contribution from the time-implicit piece of K33 is computed based on the diagnostic equation (36.182).
- The contribution from the combined `diff_cbt` vertical diffusivity and time-implicit piece of K33 is computed based on an intermediate update to the tracer fields based solely on the surface and bottom boundary fluxes, and the diagnostic calls the tridiagonal inversion.

The resulting diagnostic fields should satisfy the following equalities

- total tendency from vertical processes = surface boundary + bottom boundary + combined interior mixing (combined `diff_cbt` + time-implicit K33). For example, we have

$$\begin{aligned} \text{neut_rho_vdiffuse} = & \text{neut_rho_sbc} + \text{neut_rho_bbc} \\ & + \text{neut_rho_diff_cbt} + \text{neut_rho_k33}, \end{aligned} \quad (36.183)$$

with similar equalities holding for `wdian_rho` and `tform_rho` diagnostics mapped onto levels or binned to density layers (see Sections 36.10 and 36.11).

- tendency from combined interior mixing (`diff_cbt` + K33-implicit) equals to the separate calculation of tendency from `diff_cbt` + tendency from K33-implicit. For example, we have

$$\text{neut_rho_vmix} = +\text{neut_rho_diff_cbt} + \text{neut_rho_k33}, \quad (36.184)$$

with similar equalities holding for `wdian_rho` and `tform_rho` diagnostics mapped onto levels or binned to density layers (see Sections 36.10 and 36.11).

36.9.3 Splitting physics into flux convergence plus thermodynamic source

In Section 36.7.1, we split the contributions from physical processes into the convergence of a subgrid scale flux plus a thermodynamic source. This split then facilitated our deducing properties satisfied by various of the subgrid scale processes, in particular where neutral diffusion reduced to cabbeling and thermobaricity. However, in the discussion of dianeutral diffusion in Section 36.7.2, we chose not to consider this split as it leads there to no new physical insights.

As MOM time steps the temperature and salinity equations, rather than the locally referenced potential density equation, there is no guarantee that the discrete equations can be exactly split according to the decomposition given in Section 36.7.1. Correspondingly, when diagnosing the contributions to the material evolution, we do not artificially introduce the split to the diagnostic routines. Instead, we diagnose the terms from the temperature and salinity equations just as computed by MOM. We therefore expect that the split given in Section 36.7.1 will only hold approximately in the diagnostic calculation.

36.9.4 Cabbeling, thermobaricity, and neutral diffusion

We highlight here an issue related to the contribution from neutral diffusion to the dianeutral velocity component. First, as just mentioned in Section 36.9.3, split of the flux convergence discussed in Section 36.7.1 may not be precisely reflected in the numerical model. Hence, the contribution from neutral diffusion, given by the cabbeling and thermobaricity expressions in equation (36.103), may not be fully reflected in model diagnostics computed according to equation (36.95). To assess the agreement, we separately evaluated these two expressions, and have found that, in regions where the neutral slopes are less than the pre-defined maximum slope, the quantitative and qualitative features are reflected in both approaches. This result serves as a useful cross-check on the diagnostic code.

In regions where the neutral slopes are steep, the two approaches differ for the following reasons. The expressions for cabbeling and thermobaricity given by equation (36.103) assume that the lateral diffusion of temperature and salinity occurs along neutral directions in regions of vertically stable stratification, and that the neutral diffusion fluxes for temperature and salinity satisfy the balance (36.96). However, when the neutral slopes steepen greater than a numerically specified value S_{\max} , the numerical implementation of neutral diffusion exponentially transitions to horizontal diffusion aligned parallel to iso-surfaces of the

vertical coordinate. Furthermore, the horizontal diffusion fluxes no longer satisfy the balance (36.96). The transition to horizontal diffusion is physically motivated by the work of Treguier et al. (1997) and Ferrari et al. (2008), who note the presence of increased mixing of watermasses in boundary layer regions typically associated with steep neutral directions. That is, when neutral directions approach the vertical, dianeutral diffusion becomes lateral diffusion. There can be a large contribution to dianeutral transport associated with horizontal diffusion in regions of steep neutral directions. Yet such mixing processes are *not* associated with the traditional notion of cabbeling and thermobaricity, and so they are removed when diagnosing equation (36.103).

36.9.5 Concerning spurious dianeutral transport

Thus far in our formulation, there has been an assumption that the numerically diagnosed expressions for the dianeutral transport correspond, at least qualitatively, to the continuum forms, with quantitative differences arising from truncation errors. We identify here some caveats to this assumption.

For the case of a linear equation of state with zero vertical diffusion, zero sources, and zero boundary fluxes, we should diagnose a vanishing material evolution of both temperature and salinity, and an associated vanishing $w^{(\gamma)}$, since there will be zero dianeutral transport in the continuum fluid. Indeed, such will be the case by definition of the kinematic method (equation (36.90)), since the method uses the diagnosed time tendency and advection tendencies, which will be equal and opposite when there are no boundary fluxes nor physical mixing processes. The question arises whether this result is a correct accounting of the dianeutral transport occurring in the numerical simulation.

In a discrete model, there are generally nonzero levels of spurious mixing that arise from the numerical discretization of advection, either resolved advection or parameterized subgrid scale advection (Griffies et al., 2000b; Ilicak et al., 2012). If we could extract these spurious effects from the numerical advection operator, the associated mixing (or unmixing) could be isolated and mapped. However, an explicit diagnosis of truncation errors that create spurious dianeutral mixing is generally not possible, given the many sophisticated expressions for the numerical treatment of either resolved or parameterized advection.

Hence, even with zero physical subgrid scale fluxes, the spurious numerical fluxes do not generally vanish. The expressions for the diagnosed dianeutral transport considered in the MOM diagnostic detailed in this chapter fail to account for these spurious terms, since we do not know how to generally incorporate them into the diagnostic framework. However, one may wish the diagnostic to behave just in this manner, so that it focuses on the physical aspects of the simulation. Nonetheless, one must be careful not to over interpret the results from this diagnostic, since there may be more dianeutral transport actually occurring in the simulation than might be diagnosed via $w^{(\gamma)}$. So in practice, one can be confident in the diagnosed form of $w^{(\gamma)}$ only in those cases that the spurious levels of mixing are smaller than the physical levels.

36.10 Kinematic method diagnosed in MOM

We now summarize the MOM diagnostics computed online in support of the kinematic method used to estimate the dianeutral transport as well as the associated budgets for locally referenced potential density. For this approach, we diagnose all terms contributing to the advective-form of the material time evolution of temperature and salinity according to the discussion in Section 36.8. For completeness, we repeat here the salient equations from that section.

- SURFACE GRID CELL: The kinematic portion of equation (36.162) yields

$$\left(\rho_{,\Theta} \frac{d^{\dagger}\Theta}{dt} + \rho_{,S} \frac{d^{\dagger}S}{dt} \right)_{\text{kinematic}} \approx \frac{\rho_{,\Theta} \left(\partial_t (\rho dz \Theta) + \nabla_s \cdot (\rho dz \mathbf{u}^{\dagger} \Theta) - (\rho w^{(s)\dagger} \Theta)_{s=s_{k=1}} - Q_m \Theta \right)}{\rho dz} + \frac{\rho_{,S} \left(\partial_t (\rho dz S) + \nabla_s \cdot (\rho dz \mathbf{u}^{\dagger} S) - (\rho w^{(s)\dagger} S)_{s=s_{k=1}} - Q_m S \right)}{\rho dz}. \quad (36.185)$$

- INTERIOR GRID CELL: equation (36.127) yields

$$\left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{kinematic}} \approx \frac{\rho_{,\Theta} \left(\partial_t (\rho dz \Theta) + \nabla_s \cdot (\rho dz \mathbf{u}^+ \Theta) + \Delta_k (\rho w^{(s)+} \Theta) \right)}{\rho dz} + \frac{\rho_{,S} \left(\partial_t (\rho dz S) + \nabla_s \cdot (\rho dz \mathbf{u}^+ S) + \Delta_k (\rho w^{(s)+} S) \right)}{\rho dz}. \quad (36.186)$$

- BOTTOM GRID CELL: equation (36.138) yields

$$\left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{kinematic}} \approx \frac{\rho_{,\Theta} \left(\partial_t (\rho dz \Theta) + \nabla_s \cdot (\rho dz \mathbf{u}^+ \Theta) + (\rho w^{(s)+} \Theta)_{k-1} \right)}{\rho dz} + \frac{\rho_{,S} \left(\partial_t (\rho dz S) + \nabla_s \cdot (\rho dz \mathbf{u}^+ S) + (\rho w^{(s)+} S)_{k-1} \right)}{\rho dz}. \quad (36.187)$$

Notice the division by the mass per horizontal area, (ρdz) , which renders an expression for the density evolution in units of $(\text{kg m}^{-3}) \text{ s}^{-1}$.

We identify a suite of diagnostics in the following, where “name” refers to the name of a particular physical process.

- `neut_rho_name` = contribution to the material time derivative of locally referenced potential density, given in units of $(\text{kg m}^{-3}) \text{ s}^{-1}$.
- `neut_rho_name_on_nrho` = result of binning `neut_rho_name` into neutral density classes.
- `wdian_rho_name` = contribution to the dianeutral mass transport computed according to the algorithm summarized in Table 36.1. It is given in units of kg s^{-1} . It is computed according to

$$\text{wdian_rho_name} = \text{neut_rho_name} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.188)$$

where h is one of the coordinates x, y, z according to the algorithm in Table 36.1.

- `wdian_rho_name_on_nrho` = is the result of binning `wdian_rho_name` into neutral density classes.
- `tform_rho_name` = the contribution to the watermass transformation according to the discrete approximation (36.80), which is based on the layer approach of Walin (1982). It is computed according to

$$\text{tform_rho_name} = \text{neut_rho_name} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right), \quad (36.189)$$

where ρdV is the mass within a grid cell, and $\Delta \gamma^n$ is the specified density bins detailed in Section 36.15. Note that because of the division by $\Delta \gamma^n$, this diagnostic makes more sense when binned into neutral density classes, as per the diagnostic `tform_rho_name_on_nrho`. Nonetheless, we provide for the diagnostic `tform_rho_name` as a direct analog to `wdian_rho_name`.

- `tform_rho_name_on_nrho` = is the result of binning `tform_rho_name` into neutral density classes.

Each contributing term is diagnosed as an Eulerian time tendency, using the same numerical operations as used for the prognostic equations of conservative temperature and salinity. The motivation for doing so is to facilitate a straightforward budget diagnostic for the locally referenced potential density of a grid cell. In particular, the advection contribution is written as a convergence rather than divergence. This

point is critical to keep in mind when forming the material time derivative for computing the dianeutral mass transport.

It is also of interest to separate out the temperature and salinity contributions for many of the diagnosed terms, as a means to uncover whether it is salinity or temperature that dominates in any particular region. We focus only on those pieces that are written in advective form, as they are used to compute the dianeutral transport. For these terms we introduce the following diagnostic fields

- neut_temp_name
- neut_salt_name
- wdian_temp_name
- wdian_salt_name
- tform_temp_name
- tform_salt_name
- neut_temp_name_on_nrho
- neut_salt_name_on_nrho
- wdian_temp_name_on_nrho
- wdian_salt_name_on_nrho
- tform_temp_name_on_nrho
- tform_salt_name_on_nrho

36.10.1 Eulerian time tendency

$$\text{neut_rho_tendency} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right) \quad (36.190)$$

$$\text{neut_temp_tendency} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right) \quad (36.191)$$

$$\text{neut_salt_tendency} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right) \quad (36.192)$$

$$\text{wdian_rho_tendency} = \text{neut_rho_tendency} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right). \quad (36.193)$$

$$\text{wdian_temp_tendency} = \text{neut_temp_tendency} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right). \quad (36.194)$$

$$\text{wdian_salt_tendency} = \text{neut_salt_tendency} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right). \quad (36.195)$$

$$\text{tform_rho_tendency} = \text{neut_rho_tendency} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.196)$$

$$\text{tform_temp_tendency} = \text{neut_temp_tendency} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.197)$$

$$\text{tform_salt_tendency} = \text{neut_salt_tendency} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.198)$$

This term represents the time tendency at a grid cell due to all processes impacting temperature and salinity.

36.10.2 Advection by resolved flow

$$\text{neut_rho_advect} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,\Theta} [\nabla_s \cdot (\Theta \mathbf{u} \rho dz) + \Delta_k (\Theta w^{(s)} \rho)] + \rho_{,S} [\nabla_s \cdot (S \mathbf{u} \rho dz) + \Delta_k (S w^{(s)} \rho)] \right) \quad (36.199)$$

$$\text{neut_temp_advect} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,\Theta} [\nabla_s \cdot (\Theta \mathbf{u} \rho dz) + \Delta_k (\Theta w^{(s)} \rho)] \right) \quad (36.200)$$

$$\text{neut_salt_advect} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,S} [\nabla_s \cdot (S \mathbf{u} \rho dz) + \Delta_k (S w^{(s)} \rho)] \right) \quad (36.201)$$

$$\text{wdian_rho_advect} = \text{neut_rho_advect} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.202)$$

$$\text{wdian_temp_advect} = \text{neut_temp_advect} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.203)$$

$$\text{wdian_salt_advect} = \text{neut_salt_advect} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.204)$$

$$\text{tform_rho_advect} = \text{neut_rho_advect} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.205)$$

$$\text{tform_temp_advect} = \text{neut_temp_advect} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.206)$$

$$\text{tform_salt_advect} = \text{neut_salt_advect} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.207)$$

36.10.3 Gent-McWilliams transport

If enabling the mesoscale parameterization of [Gent and McWilliams \(1990\)](#) and [Gent et al. \(1995\)](#), we may evaluate the following contribution

$$\text{neut_rho_gm} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,\Theta} [\nabla_s \cdot (\Theta \mathbf{u}^{\text{gm}} \rho dz) + \Delta_k (\Theta w^{\text{gm}} \rho)] + \rho_{,S} [\nabla_s \cdot (S \mathbf{u}^{\text{gm}} \rho dz) + \Delta_k (S w^{\text{gm}} \rho)] \right) \quad (36.208)$$

$$\text{neut_temp_gm} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,\Theta} [\nabla_s \cdot (\Theta \mathbf{u}^{\text{gm}} \rho dz) + \Delta_k (\Theta w^{\text{gm}} \rho)] \right) \quad (36.209)$$

$$\text{neut_salt_gm} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,S} [\nabla_s \cdot (S \mathbf{u}^{\text{gm}} \rho dz) + \Delta_k (S w^{\text{gm}} \rho)] \right) \quad (36.210)$$

$$\text{wdian_rho_gm} = \text{neut_rho_gm} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.211)$$

$$\text{wdian_temp_gm} = \text{neut_temp_gm} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.212)$$

$$\text{wdian_salt_gm} = \text{neut_salt_gm} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.213)$$

$$\text{tform_rho_gm} = \text{neut_rho_gm} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.214)$$

$$\text{tform_temp_gm} = \text{neut_temp_gm} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.215)$$

$$\text{tform_salt_gm} = \text{neut_salt_gm} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.216)$$

Note that we have written the Gent-McWilliams contribution as an eddy advection term. However, this term may instead be evaluated according to the skew flux approach of [Griffies \(1998\)](#). Additionally, the modified form of GM transport proposed by [Ferrari et al. \(2010\)](#) is also diagnosed in this field.

36.10.4 Submesoscale transport

If enabling the submesoscale parameterization of [Fox-Kemper et al. \(2008b\)](#), we may evaluate the following contribution

$$\text{neut_rho_submeso} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,\Theta} [\nabla_s \cdot (\Theta \mathbf{u}^{\text{sub}} \rho dz) + \Delta_k (\Theta w^{\text{sub}} \rho)] + \rho_{,S} [\nabla_s \cdot (S \mathbf{u}^{\text{sub}} \rho dz) + \Delta_k (S w^{\text{sub}} \rho)] \right) \quad (36.217)$$

$$\text{neut_temp_submeso} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,\Theta} [\nabla_s \cdot (\Theta \mathbf{u}^{\text{sub}} \rho dz) + \Delta_k (\Theta w^{\text{sub}} \rho)] \right) \quad (36.218)$$

$$\text{neut_salt_submeso} = -\left(\frac{1}{\rho dz}\right) \left(\rho_{,S} [\nabla_s \cdot (S \mathbf{u}^{\text{sub}} \rho dz) + \Delta_k (S w^{\text{sub}} \rho)] \right) \quad (36.219)$$

$$\text{wdian_rho_submeso} = \text{neut_rho_submeso} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.220)$$

$$\text{wdian_temp_submeso} = \text{neut_temp_submeso} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.221)$$

$$\text{wdian_salt_submeso} = \text{neut_salt_submeso} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.222)$$

$$\text{tform_rho_submeso} = \text{neut_rho_submeso} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.223)$$

$$\text{tform_temp_submeso} = \text{neut_temp_submeso} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.224)$$

$$\text{tform_salt_submeso} = \text{neut_salt_submeso} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.225)$$

Note that we have written the submesoscale contribution as an eddy advection term. However, this term may instead be evaluated according to the skew flux approach of [Griffies \(1998\)](#).

36.10.5 Precipitation minus evaporation: flux-form

The form that precipitation minus evaporation appears in the finite volume flux-form budget for tracer (Section [36.8.6](#)) is diagnosed as

$$\text{neut_rho_pme} = \frac{\text{pme}}{\rho dz} \left(\rho_{,\Theta} \Theta^{\text{pme}} + \rho_{,S} S^{\text{pme}} \right) \quad (36.226)$$

$$\text{wdian_rho_pme} = \text{neut_rho_pme} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.227)$$

$$\text{tform_rho_pme} = \text{neut_rho_pme} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.228)$$

In this expression, pme is the mass flux of precipitation minus evaporation that “advects” temperature and salinity across the ocean surface, with Θ^{pme} and S^{pme} the temperature and salinity of the precipitation minus evaporation.

36.10.6 Precipitation minus evaporation: advective-form

According to equation (36.185), the manner that precipitation minus evaporation impacts the surface grid cell advective-form material time derivative is diagnosed according to

$$\text{neut_rho_pbl_pme_kn} = -\frac{\text{pme}}{\rho dz} \left(\rho, \Theta \Theta_{k=1} + \rho, S S_{k=1} \right) \quad (36.229)$$

$$\text{neut_temp_pbl_pme_kn} = -\frac{\text{pme}}{\rho dz} \left(\rho, \Theta \Theta_{k=1} \right) \quad (36.230)$$

$$\text{neut_salt_pbl_pme_kn} = -\frac{\text{pme}}{\rho dz} \left(\rho, S S_{k=1} \right) \quad (36.231)$$

$$\text{wdian_rho_pbl_pme_kn} = \text{neut_rho_pbl_pme_kn} \left(\frac{\rho dA_{(h)}}{|\rho, \Theta \Theta_{,h} + \rho, S S_{,h}|} \right) \quad (36.232)$$

$$\text{wdian_temp_pbl_pme_kn} = \text{neut_temp_pbl_pme_kn} \left(\frac{\rho dA_{(h)}}{|\rho, \Theta \Theta_{,h} + \rho, S S_{,h}|} \right) \quad (36.233)$$

$$\text{wdian_salt_pbl_pme_kn} = \text{neut_salt_pbl_pme_kn} \left(\frac{\rho dA_{(h)}}{|\rho, \Theta \Theta_{,h} + \rho, S S_{,h}|} \right) \quad (36.234)$$

$$\text{tform_rho_pbl_pme_kn} = \text{neut_rho_pbl_pme_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.235)$$

$$\text{tform_temp_pbl_pme_kn} = \text{neut_temp_pbl_pme_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.236)$$

$$\text{tform_salt_pbl_pme_kn} = \text{neut_salt_pbl_pme_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.237)$$

36.10.7 Liquid plus solid river runoff: flux-form

The form that liquid plus solid river mass flux enters the ocean appears in the finite volume flux-form budget for tracer is diagnosed as

$$\text{neut_rho_rivermix} = \frac{\text{river}}{\rho dz} \left(\rho, \Theta \Theta^{\text{river}} + \rho, S S^{\text{river}} \right) \quad (36.238)$$

$$\text{wdian_rho_rivermix} = \text{neut_rho_rivermix} \left(\frac{\rho dA_{(h)}}{|\rho, \Theta \Theta_{,h} + \rho, S S_{,h}|} \right) \quad (36.239)$$

$$\text{tform_rho_rivermix} = \text{neut_rho_rivermix} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.240)$$

In this expression, *river* is the mass flux that “advects” temperature and salinity across the ocean surface due to liquid plus solid river runoff, with Θ^{river} and S^{river} the conservative temperature and salinity of the river water, respectively. This diagnostic is used when the liquid plus solid (i.e., calving land ice) river runoff is combined into one field. Note that we generally insert the river runoff over a few of the upper ocean model grid cells, so that river runoff will appear over more than just the top model grid cell.

36.10.8 Liquid plus solid river runoff: advective-form

According to equation (36.185), the manner that liquid plus solid river mass flux impacts the advective-form material time derivative is diagnosed as

$$\text{neut_rho_pbl_riv_kn} = -\frac{\text{river}}{\rho dz} \left(\rho_{,\Theta} \Theta_{k=1} + \rho_{,S} S_{k=1} \right) \quad (36.241)$$

$$\text{neut_temp_pbl_riv_kn} = -\frac{\text{river}}{\rho dz} \left(\rho_{,\Theta} \Theta_{k=1} \right) \quad (36.242)$$

$$\text{neut_salt_pbl_riv_kn} = -\frac{\text{river}}{\rho dz} \left(\rho_{,S} S_{k=1} \right) \quad (36.243)$$

$$\text{wdian_rho_pbl_riv_kn} = \text{neut_rho_pbl_riv_kn} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.244)$$

$$\text{wdian_temp_pbl_riv_kn} = \text{neut_temp_pbl_riv_kn} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.245)$$

$$\text{wdian_salt_pbl_riv_kn} = \text{neut_salt_pbl_riv_kn} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.246)$$

$$\text{tform_rho_pbl_riv_kn} = \text{neut_rho_pbl_riv_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.247)$$

$$\text{tform_temp_pbl_riv_kn} = \text{neut_temp_pbl_riv_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.248)$$

$$\text{tform_salt_pbl_riv_kn} = \text{neut_salt_pbl_riv_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.249)$$

36.10.9 Liquid river runoff: flux-form

The form that liquid river mass flux enters the ocean appears in the finite volume flux-form budget for tracer is diagnosed as

$$\text{neut_rho_runoffmix} = \frac{\text{runoff}}{\rho dz} \left(\rho_{,\Theta} \Theta^{\text{runoff}} + \rho_{,S} S^{\text{runoff}} \right) \quad (36.250)$$

$$\text{wdian_rho_runoffmix} = \text{neut_rho_runoffmix} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.251)$$

$$\text{tform_rho_runoffmix} = \text{neut_rho_runoffmix} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.252)$$

In this expression, runoff is the mass flux that “advects” temperature and salinity across the ocean surface due to liquid river runoff, with Θ^{runoff} and S^{runoff} the temperature and salinity of the liquid river runoff. Note that we generally insert the river runoff over a few of the upper ocean model grid cells, so that river runoff will appear over more than just the top model grid cell.

36.10.10 Liquid river runoff: advective-form

According to equation (36.185), the manner that liquid river mass flux impacts the advective-form material time derivative is diagnosed as

$$\text{neut_rho_pbl_rn_kn} = -\frac{\text{runoff}}{\rho dz} \left(\rho_{,\Theta} \Theta_{k=1} + \rho_{,S} S_{k=1} \right) \quad (36.253)$$

$$\text{neut_temp_pbl_rn_kn} = -\frac{\text{runoff}}{\rho dz} \left(\rho_{,\Theta} \Theta_{k=1} \right) \quad (36.254)$$

$$\text{neut_salt_pbl_rn_kn} = -\frac{\text{runoff}}{\rho dz} \left(\rho_{,S} S_{k=1} \right) \quad (36.255)$$

$$\text{wdian_rho_pbl_rn_kn} = \text{neut_rho_pbl_rn_kn} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.256)$$

$$\text{wdian_temp_pbl_rn_kn} = \text{neut_temp_pbl_rn_kn} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.257)$$

$$\text{wdian_salt_pbl_rn_kn} = \text{neut_salt_pbl_rn_kn} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.258)$$

$$\text{tform_rho_pbl_rn_kn} = \text{neut_rho_pbl_rn_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.259)$$

$$\text{tform_temp_pbl_rn_kn} = \text{neut_temp_pbl_rn_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.260)$$

$$\text{tform_salt_pbl_rn_kn} = \text{neut_salt_pbl_rn_kn} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.261)$$

36.10.11 Solid calving land ice: flux-form

The form that solid mass flux (i.e., calving) enters the ocean appears in the finite volume flux-form budget for tracer is diagnosed as

$$\text{neut_rho_calvingmix} = \frac{\text{calving}}{\rho dz} \left(\rho_{,\Theta} \Theta^{\text{calving}} + \rho_{,S} S^{\text{calving}} \right) \quad (36.262)$$

$$\text{wdian_rho_calvingmix} = \text{neut_rho_calvingmix} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.263)$$

$$\text{tform_rho_calvingmix} = \text{neut_rho_calvingmix} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.264)$$

In this expression, calving is the mass flux that “advects” temperature and salinity across the ocean surface due to solid calving land ice, with Θ^{calving} and S^{calving} the temperature and salinity of the land ice.

36.10.12 Solid calving runoff: advective-form

According to equation (36.185), the form that solid calving mass flux impacts the advective-form material time derivative is diagnosed as

$$\text{neut_rho_pbl_cl_kn} = -\frac{\text{calving}}{\rho \, dz} \left(\rho_{,\Theta} \Theta_{k=1} + \rho_{,S} S_{k=1} \right) \quad (36.265)$$

$$\text{neut_temp_pbl_cl_kn} = -\frac{\text{calving}}{\rho \, dz} \left(\rho_{,\Theta} \Theta_{k=1} \right) \quad (36.266)$$

$$\text{neut_salt_pbl_cl_kn} = -\frac{\text{calving}}{\rho \, dz} \left(\rho_{,S} S_{k=1} \right) \quad (36.267)$$

$$\text{wdian_rho_pbl_cl_kn} = \text{neut_rho_pbl_cl_kn} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.268)$$

$$\text{wdian_temp_pbl_cl_kn} = \text{neut_temp_pbl_cl_kn} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.269)$$

$$\text{wdian_salt_pbl_cl_kn} = \text{neut_salt_pbl_cl_kn} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.270)$$

$$\text{tform_rho_pbl_cl_kn} = \text{neut_rho_pbl_cl_kn} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right). \quad (36.271)$$

$$\text{tform_temp_pbl_cl_kn} = \text{neut_temp_pbl_cl_kn} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right). \quad (36.272)$$

$$\text{tform_salt_pbl_cl_kn} = \text{neut_salt_pbl_cl_kn} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right). \quad (36.273)$$

36.10.13 Summary of the kinematic method

We take the convention that each term is diagnosed as an Eulerian time tendency, using the same numerical operations as used for the prognostic equations of conservative temperature and salinity. This point explains the minus signs in front of the advection and surface water flux terms appearing below in equation (36.274), with these signs required to form the material time derivative for computing the dianeutral mass transport.

36.10.13.1 Material time derivative

The finite volume estimate of the advective-form of the kinematic material time derivative is given by

$$\begin{aligned} \left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{kinematic level}} &\approx \text{neut_rho_tendency} - \text{neut_rho_advect} \\ &\quad - \text{neut_rho_gm} - \text{neut_rho_submeso} \\ &\quad + \text{neut_rho_pbl_pme_kn} \\ &\quad + \text{neut_rho_pbl_rn_kn} + \text{neut_rho_pbl_cl_kn} \end{aligned} \quad (36.274)$$

where we assumed liquid plus solid runoff are split into their own separate arrays, as per a realistic climate model with a liquid runoff and solid calving scheme. If these terms are combined, as commonly done in ocean-ice models, we should instead diagnose

$$\begin{aligned} \left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{kinematic level}} &\approx \text{neut_rho_tendency} - \text{neut_rho_advect} \\ &\quad - \text{neut_rho_gm} - \text{neut_rho_submeso} \\ &\quad + \text{neut_rho_pbl_pme_kn} \\ &\quad + \text{neut_rho_pbl_riv_kn}. \end{aligned} \quad (36.275)$$

The same relations also hold for the density binned form

$$\begin{aligned} \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right)_{\text{kinematic layer}} &\approx \text{neut_rho_tendency_on_nrho} - \text{neut_rho_advect_on_nrho} \\ &\quad - \text{neut_rho_gm_on_nrho} - \text{neut_rho_submeso_on_nrho} \\ &\quad + \text{neut_rho_pbl_pme_kn_on_nrho} \\ &\quad + \text{neut_rho_pbl_rn_kn_on_nrho} + \text{neut_rho_pbl_cl_kn_on_nrho}. \end{aligned} \quad (36.276)$$

36.10.13.2 Dianeutral transport from wdian diagnostics

The dianeutral mass transport, in units of kg s^{-1} , can be estimated using `wdian` both on levels and binned to neutral density layers. Following from the `neut_rho` diagnostics above, we have

$$\begin{aligned} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right)_{\text{kinematic level}} &\approx \text{wdian_rho_tendency} - \text{wdian_rho_advect} \\ &\quad - \text{wdian_rho_gm} - \text{wdian_rho_submeso} \\ &\quad + \text{wdian_rho_pbl_pme_kn} \\ &\quad + \text{wdian_rho_pbl_rn_kn} + \text{wdian_rho_pbl_cl_kn}. \end{aligned} \quad (36.277)$$

The same relations also hold for the density binned dianeutral mass transport

$$\begin{aligned} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right)_{\text{kinematic layer}} &\approx \text{wdian_rho_tendency_on_nrho} - \text{wdian_rho_advect_on_nrho} \\ &\quad - \text{wdian_rho_gm_on_nrho} - \text{wdian_rho_submeso_on_nrho} \\ &\quad + \text{wdian_rho_pbl_pme_kn_on_nrho} \\ &\quad + \text{wdian_rho_pbl_rn_kn_on_nrho} + \text{wdian_rho_pbl_cl_kn_on_nrho}. \end{aligned} \quad (36.278)$$

36.10.13.3 Dianeutral transport from tform diagnostics

The dianeutral mass transport, in units of kg s^{-1} , can also be estimated using `tform` diagnostics. Here, the results are only sensible when binned to neutral density layers. Following from the `wdian_rho` diagnostics above, we have

$$\begin{aligned} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right)_{\text{kinematic layer}} &\approx \text{tform_rho_tendency_on_nrho} - \text{tform_rho_advect_on_nrho} \\ &\quad - \text{tform_rho_gm_on_nrho} - \text{tform_rho_submeso_on_nrho} \\ &\quad + \text{tform_rho_pbl_pme_kn_on_nrho} \\ &\quad + \text{tform_rho_pbl_rn_kn_on_nrho} + \text{tform_rho_pbl_cl_kn_on_nrho}. \end{aligned} \quad (36.279)$$

36.11 Process method diagnosed in MOM

We now summarize the MOM diagnostics computed online in support of the process method used to estimate the dianeutral transport as well as the associated budgets for locally referenced potential density. For this approach, we diagnose all terms contributing to the advective-form of the material time evolution of temperature and salinity according to the discussion in Section 36.8. For completeness, we repeat here the salient equations from that section.

- SURFACE GRID CELL: The process portion of equation (36.162) yields

$$\left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{kinematic}} \approx \frac{\rho_{,\Theta} \left(-\nabla_s \cdot (dz \mathbf{J}^\Theta) + (J^{(s)\Theta})_{s=s_{k-1}} + Q_{\text{pbl}}^\Theta + Q_m (\Theta_m - \Theta) + \rho dz \mathcal{S}^\Theta \right)}{\rho dz} + \frac{\rho_{,S} \left(-\nabla_s \cdot (dz \mathbf{J}^S) + (J^{(s)S})_{s=s_{k-1}} + Q_{\text{pbl}}^S + Q_m (S_m - S) + \rho dz \mathcal{S}^S \right)}{\rho dz}. \quad (36.280)$$

- INTERIOR GRID CELL: equation (36.128) yields

$$\left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{kinematic}} \approx \frac{\rho_{,\Theta} \left(-\nabla_s \cdot (dz \mathbf{J}^\Theta) - (J^{(s)\Theta})_{s=s_{k-1}} + (J^{(s)\Theta})_{s=s_k} + \rho dz \mathcal{S}^\Theta \right)}{\rho dz} + \frac{\rho_{,S} \left(-\nabla_s \cdot (dz \mathbf{J}^S) - (J^{(s)S})_{s=s_{k-1}} + (J^{(s)S})_{s=s_k} + \rho dz \mathcal{S}^S \right)}{\rho dz}. \quad (36.281)$$

- BOTTOM GRID CELL: equation (36.139) yields

$$\left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{kinematic}} = \frac{\rho_{,\Theta} \left(-\nabla_s \cdot (dz \mathbf{J}^\Theta) - (J^{(s)\Theta})_{s=s_{k-1}} - Q^{\text{bott}} + \rho dz \mathcal{S}^\Theta \right)}{\rho dz} + \frac{\rho_{,S} \left(-\nabla_s \cdot (dz \mathbf{J}^S) - (J^{(s)S})_{s=s_{k-1}} + \rho dz \mathcal{S}^S \right)}{\rho dz}. \quad (36.282)$$

As for the kinematic method in Section 36.10, we identify a suite of diagnostics in the following, where “name” refers to the name of a particular physical process:

- `neut_rho_name` = contribution to the material time derivative of locally referenced potential density in units of $(\text{kg m}^{-3}) \text{ s}^{-1}$.
- `neut_rho_name_on_nrho` = result of binning `neut_rho_name` into neutral density classes.
- `wdian_rho_name` = contribution to the dianeutral mass transport computed according to the algorithm summarized in Table 36.1. It is given in units of kg s^{-1} . It is computed according to

$$\text{wdian_rho_name} = \text{neut_rho_name} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.283)$$

where h is one of the coordinates x, y, z according to the algorithm in Table 36.1.

- `wdian_rho_name_on_nrho` = is the result of binning `wdian_rho_name` into neutral density classes.
- `tform_rho_name` = the contribution to the watermass transformation according to the discrete approximation (36.80), which is based on the layer approach of Walin (1982). It is computed according to

$$\text{tform_rho_name} = \text{neut_rho_name} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right), \quad (36.284)$$

where ρdV is the mass within a grid cell, and $\Delta \gamma^n$ is the specified density bins detailed in Section 36.15. Note that because of the division by $\Delta \gamma^n$, this diagnostic makes more sense when binned into neutral density classes, as per the diagnostic `tform_rho_name_on_nrho`. Nonetheless, we provide for the diagnostic `tform_rho_name` as a direct analog to `wdian_rho_name`.

- `tform_rho_name_on_nrho` = is the result of binning `tform_rho_name` into neutral density classes.

As for the kinematic approach, our convention is to diagnose each term as a time tendency. Additionally, we decompose the terms into temperature and salinity contributions, thus allowing for further refined diagnostics to quantify the separate impacts from salinity and temperature.

36.11.1 Boundary fluxes of heat and salt through the vertical mixing operator

This suite of diagnostics computes the contribution to the material time derivative of locally referenced potential density arising from boundary buoyancy fluxes associated with heat and salt fluxes. All of these diagnostics are computed in the module

`ocean_param/mixing/ocean_vert_mix.F90`

as per the model implementation of boundary fluxes as part of the time-implicit vertical mixing operator. We diagnose these contributions according to the incremental methods detailed in Section 36.9.2.

There is some redundancy here with the diagnostic for vertical mixing discussed in Section 36.11.3.1. As mentioned in Section 36.11.3.11, this redundancy allows for a double-check on the integrity of the diagnostics.

36.11.1.1 Surface boundary heat fluxes

The following diagnostic accounts for the contribution from surface boundary fluxes of heat. In a coupled model, or a model using the CORE protocol of Griffies et al. (2009), this heat flux is associated with short-wave, longwave, latent (both latent heat of vaporization and latent heat of fusion), and sensible heating. In a model that has a surface restoring boundary flux, then the restoring flux is included in this diagnostic. Note that there is a conversion between a heat flux and conservative temperature flux according to

$$Q^\Theta = \frac{Q^{\text{heat}}}{C_p} \quad (36.285)$$

where

$$C_p = 3992.103 \text{ J kg}^{-1} \text{ C}^{-1} \quad (36.286)$$

is the seawater heat capacity and Q^{heat} is the surface heat flux (in units of W m^{-2}).

$$\text{neut_rho_sbc_temp} = \frac{1}{\rho dz} \left(\rho_{,\Theta} (Q_{\text{sw}}^\Theta + Q_{\text{lw}}^\Theta + Q_{\text{latent}}^\Theta + Q_{\text{sensible}}^\Theta) \right) \quad (36.287)$$

$$\text{wdian_rho_sbc_temp} = \text{neut_rho_sbc_temp} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.288)$$

$$\text{tform_rho_sbc_temp} = \text{neut_rho_sbc_temp} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.289)$$

36.11.1.2 Surface boundary salt fluxes

The following diagnostic field accounts for the contribution from surface boundary fluxes of salt associated with melting and forming of sea ice. Additionally, in an ocean-ice model, there may be a salt flux associated with surface salinity restoring towards a climatology. Such surface restoring fluxes will also be accounted for in this diagnostic field.

$$\text{neut_rho_sbc_salt} = \frac{1}{\rho dz} \left(\rho_{,S} Q^S \right) \quad (36.290)$$

$$\text{wdian_rho_sbc_salt} = \text{neut_rho_sbc_salt} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.291)$$

$$\text{tform_rho_sbc_salt} = \text{neut_rho_sbc_salt} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.292)$$

36.11.1.3 Net surface boundary heat and salt fluxes

The following diagnostic field accounts for the contribution from the combined surface boundary fluxes of heat and salt.

$$\text{neut_rho_sbc} = \frac{1}{\rho dz} \left(\rho_{,\Theta} (Q_{\text{sw}}^{\Theta} + Q_{\text{lw}}^{\Theta} + Q_{\text{latent}}^{\Theta} + Q_{\text{sensible}}^{\Theta}) + \rho_{,S} Q^S \right) \quad (36.293)$$

$$\text{wdian_rho_sbc} = \text{neut_rho_sbc} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.294)$$

$$\text{tform_rho_sbc} = \text{neut_rho_sbc} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.295)$$

36.11.1.4 Bottom boundary heat flux

The following diagnostic field accounts for the contribution from bottom boundary fluxes of heat associated with geothermal heating.

$$\text{neut_rho_bbc_temp} = \frac{1}{\rho dz} \left(\rho_{,\Theta} Q_{\text{geothermal}}^{\Theta} \right) \quad (36.296)$$

$$\text{wdian_rho_bbc_temp} = \text{neut_rho_bbc_temp} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.297)$$

$$\text{tform_rho_bbc_temp} = \text{neut_rho_bbc_temp} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.298)$$

36.11.1.5 Penetrative shortwave radiation

$$\text{neut_rho_sw} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{sw}} \quad (36.299)$$

$$\text{wdian_rho_sw} = \text{neut_rho_sw} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.300)$$

$$\text{tform_rho_sw} = \text{neut_rho_sw} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.301)$$

This contribution arises from penetrative shortwave radiation. This term only impacts temperature.

Note that the contribution at model grid level $k = 1$ appears to be a large cooling. This is because the flux entering through the top of the top level is set to zero in the shortwave penetration module, since the model has already incorporated this flux through the surface boundary condition module `ocean_sbc.F90`. This approach avoids double counting the impact of shortwave radiation. Full details of this issue are given in Chapter 8 of [Griffies et al. \(2004\)](#). To get the full impact from shortwave radiation on watermass transformation requires adding the two diagnostic terms. In particular, for `tform`, we need

$$\text{full impacts from shortwave} = \text{tform_rho_sw_on_nrho} + \text{tform_rho_pbl_sw_on_nrho}, \quad (36.302)$$

where the diagnostic term `tform_rho_pbl_sw_on_nrho` is presented in Section [36.14.2](#).

36.11.2 Boundary fluxes of buoyancy arising from mass transport

This suite of diagnostics computes the contribution to the material time derivative of locally referenced potential density arising from boundary buoyancy fluxes associated with mass fluxes. These diagnostics are computed in the modules

```
ocean_core/ocean_sbc.F90
ocean_tracers/ocean_tracer.F90
ocean_param/sources/ocean_rivermix.F90
```

We base these diagnostics on expressions given in equation (36.162) (see also equation (36.280)).

36.11.2.1 Precipitation minus evaporation

According to equation (36.280), the form that precipitation minus evaporation impacts the surface grid cell process version of the material time derivative is diagnosed according to

$$\text{neut_rho_pbl_pme_pr} = \frac{\text{pme}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta_m - \Theta_{k=1}) + \rho_{,S} (S_m - S_{k=1}) \right) \quad (36.303)$$

$$\text{neut_temp_pbl_pme_pr} = \frac{\text{pme}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta_m - \Theta_{k=1}) \right) \quad (36.304)$$

$$\text{neut_salt_pbl_pme_pr} = \frac{\text{pme}}{\rho \, dz} \left(\rho_{,S} (S_m - S_{k=1}) \right) \quad (36.305)$$

$$\text{wdian_rho_pbl_pme_pr} = \text{neut_rho_pbl_pme_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.306)$$

$$\text{wdian_temp_pbl_pme_pr} = \text{neut_temp_pbl_pme_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.307)$$

$$\text{wdian_salt_pbl_pme_pr} = \text{neut_salt_pbl_pme_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.308)$$

$$\text{tform_rho_pbl_pme_pr} = \text{neut_rho_pbl_pme_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right) \quad (36.309)$$

$$\text{tform_temp_pbl_pme_pr} = \text{neut_temp_pbl_pme_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right) \quad (36.310)$$

$$\text{tform_salt_pbl_pme_pr} = \text{neut_salt_pbl_pme_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right). \quad (36.311)$$

36.11.2.2 Liquid plus solid river runoff

According to equation (36.280), the form that liquid plus solid water mass impacts the surface grid cell process version of the material time derivative is diagnosed according to

$$\text{neut_rho_pbl_rv_pr} = \frac{\text{river}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta_m - \Theta_{k=1}) + \rho_{,S} (S_m - S_{k=1}) \right) \quad (36.312)$$

$$\text{neut_temp_pbl_rv_pr} = \frac{\text{river}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta_m - \Theta_{k=1}) \right) \quad (36.313)$$

$$\text{neut_salt_pbl_rv_pr} = \frac{\text{river}}{\rho \, dz} \left(\rho_{,S} (S_m - S_{k=1}) \right) \quad (36.314)$$

$$\text{wdian_rho_pbl_rv_pr} = \text{neut_rho_pbl_rv_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.315)$$

$$\text{wdian_temp_pbl_rv_pr} = \text{neut_temp_pbl_rv_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.316)$$

$$\text{wdian_salt_pbl_rv_pr} = \text{neut_salt_pbl_rv_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.317)$$

$$\text{tform_rho_pbl_rv_pr} = \text{neut_rho_pbl_rv_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right) \quad (36.318)$$

$$\text{tform_temp_pbl_rv_pr} = \text{neut_temp_pbl_rv_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right) \quad (36.319)$$

$$\text{tform_salt_pbl_rv_pr} = \text{neut_salt_pbl_rv_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right). \quad (36.320)$$

In this expression, river is the mass flux that “advects” temperature and salinity across the ocean surface due to liquid plus solid river runoff, with Θ^{river} and S^{river} the conservative temperature and salinity of the river water, respectively. This diagnostic is used when the liquid plus solid (i.e., calving land ice) river runoff is combined into one field. Note that we generally insert the river runoff over a few of the upper ocean model grid cells, so that river runoff will appear over more than just the top model grid cell. However, this diagnostic assumes all the water enters the top cell, thus providing an approximate expression for the impacts from river water.

36.11.2.3 Liquid river runoff

According to equation (36.280), the form that liquid river runoff impacts the process version of the material time derivative is diagnosed according to

$$\text{neut_rho_pbl_rn_pr} = \frac{\text{runoff}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta^{\text{runoff}} - \Theta_{k=1}) + \rho_{,S} (S^{\text{runoff}} - S_{k=1}) \right) \quad (36.321)$$

$$\text{neut_temp_pbl_rn_pr} = \frac{\text{runoff}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta^{\text{runoff}} - \Theta_{k=1}) \right) \quad (36.322)$$

$$\text{neut_salt_pbl_rn_pr} = \frac{\text{runoff}}{\rho \, dz} \left(\rho_{,S} (S^{\text{runoff}} - S_{k=1}) \right) \quad (36.323)$$

$$\text{wdian_rho_pbl_rn_pr} = \text{neut_rho_pbl_rn_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.324)$$

$$\text{wdian_temp_pbl_rn_pr} = \text{neut_temp_pbl_rn_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.325)$$

$$\text{wdian_salt_pbl_rn_pr} = \text{neut_salt_pbl_rn_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.326)$$

$$\text{tform_rho_pbl_rn_pr} = \text{neut_rho_pbl_rn_pr} \left(\frac{B \rho \, dV}{\Delta \gamma^n} \right) \quad (36.327)$$

$$\text{tform_temp_pbl_rn_pr} = \text{neut_temp_pbl_rn_pr} \left(\frac{B \rho \, dV}{\Delta \gamma^n} \right) \quad (36.328)$$

$$\text{tform_salt_pbl_rn_pr} = \text{neut_salt_pbl_rn_pr} \left(\frac{B \rho \, dV}{\Delta \gamma^n} \right). \quad (36.329)$$

In this expression, `runoff` is the mass flux that “advects” temperature and salinity across the ocean surface due to liquid river runoff, with Θ^{runoff} and S^{runoff} the temperature and salinity of the liquid river runoff. Note that we generally insert the river runoff over a few of the upper ocean model grid cells, so that river runoff will appear over more than just the top model grid cell. However, this diagnostic assumes all the water enters the top cell, thus providing an approximate expression for the impacts from river water.

36.11.2.4 Solid calving runoff

According to equation (36.280), the form that solid river runoff, or calving, impacts the process version of the material time derivative is diagnosed according to

$$\text{neut_rho_pbl_cl_pr} = \frac{\text{calving}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta^{\text{calve}} - \Theta_{k=1}) + \rho_{,S} (S^{\text{calve}} - S_{k=1}) \right) \quad (36.330)$$

$$\text{neut_temp_pbl_cl_pr} = \frac{\text{calving}}{\rho \, dz} \left(\rho_{,\Theta} (\Theta^{\text{calve}} - \Theta_{k=1}) \right) \quad (36.331)$$

$$\text{neut_salt_pbl_cl_pr} = \frac{\text{calving}}{\rho \, dz} \left(\rho_{,S} (S^{\text{calve}} - S_{k=1}) \right) \quad (36.332)$$

$$\text{wdian_rho_pbl_cl_pr} = \text{neut_rho_pbl_cl_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.333)$$

$$\text{wdian_temp_pbl_cl_pr} = \text{neut_temp_pbl_cl_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.334)$$

$$\text{wdian_salt_pbl_cl_pr} = \text{neut_salt_pbl_cl_pr} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.335)$$

$$\text{tform_rho_pbl_cl_pr} = \text{neut_rho_pbl_cl_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right) \quad (36.336)$$

$$\text{tform_temp_pbl_cl_pr} = \text{neut_temp_pbl_cl_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right) \quad (36.337)$$

$$\text{tform_salt_pbl_cl_pr} = \text{neut_salt_pbl_cl_pr} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right). \quad (36.338)$$

In this expression, *calving* is the mass flux that “advects” temperature and salinity across the ocean surface due to liquid river runoff, with Θ^{calve} and S^{calve} the temperature and salinity of the liquid river runoff. Note that we generally insert the river runoff over a few of the upper ocean model grid cells, so that river runoff will appear over more than just the top model grid cell. However, this diagnostic assumes all the water enters the top cell, thus providing an approximate expression for the impacts from river water.

36.11.3 Vertical mixing processes

This suite of diagnostics computes the contribution to $d^+ \gamma / dt$ due to vertical mixing processes, including diffusion, boundary fluxes, and parameterized vertical convection. All of these diagnostics are computed in the module

ocean_param/ocean_vert_mix.F90

36.11.3.1 Vertical diffusion and boundary fluxes

$$\text{neut_rho_vdiffuse} = \frac{1}{\rho \, dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho \, dz) + \rho_{,S} \partial_t (S \rho \, dz) \right)_{\text{vdiffuse}} \quad (36.339)$$

$$\text{wdian_rho_vdiffuse} = \text{neut_rho_vdiffuse} \left(\frac{\rho \, dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.340)$$

$$\text{tform_rho_vdiffuse} = \text{neut_rho_vdiffuse} \left(\frac{\mathcal{B} \rho \, dV}{\Delta \gamma^n} \right). \quad (36.341)$$

This contribution arises from the vertical diffusion operator acting on temperature and salinity. It includes both the mixing from vertical diffusivity, as well as that from boundary fluxes. Note that the vertical diffusivity may be large in the boundary regions in order to stabilize gravitationally unstable columns.

36.11.3.2 Vertical diffusion from net vertical diffusivity

We estimate the contribution from dianeutral mixing due to the net vertical diffusivity in a manner discussed in Section 36.9.2.3. The following diagnostics are related to this contribution.

$$\text{neut_rho_diff_cbt} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt}} \quad (36.342)$$

$$\text{neut_temp_diff_cbt} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{diff_cbt}} \quad (36.343)$$

$$\text{neut_salt_diff_cbt} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt}} \quad (36.344)$$

$$\text{wdian_rho_diff_cbt} = \text{neut_rho_diff_cbt} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.345)$$

$$\text{wdian_temp_diff_cbt} = \text{neut_temp_diff_cbt} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.346)$$

$$\text{wdian_salt_diff_cbt} = \text{neut_salt_diff_cbt} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.347)$$

$$\text{tform_rho_diff_cbt} = \text{neut_rho_diff_cbt} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.348)$$

$$\text{tform_temp_diff_cbt} = \text{neut_temp_diff_cbt} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.349)$$

$$\text{tform_salt_diff_cbt} = \text{neut_salt_diff_cbt} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.350)$$

This contribution arises from dianeutral mixing acting on temperature and salinity, and it excludes the contribution from boundary fluxes. Note that vertical diffusivity may be large in the boundary regions in order to stabilize gravitationally unstable columns.

36.11.3.3 Vertical diffusion from static background vertical diffusivity

We estimate the contribution from dianeutral mixing due to background static vertical diffusivity in a manner discussed in Section 36.9.2.3. The following diagnostics are related to this contribution.

$$\text{neut_rho_diff_back} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt_back}} \quad (36.351)$$

$$\text{neut_temp_diff_back} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{diff_cbt_back}} \quad (36.352)$$

$$\text{neut_salt_diff_back} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt_back}} \quad (36.353)$$

$$\text{wdian_rho_diff_back} = \text{neut_rho_diff_back} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.354)$$

$$\text{wdian_temp_diff_back} = \text{neut_temp_diff_back} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.355)$$

$$\text{wdian_salt_diff_back} = \text{neut_salt_diff_back} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.356)$$

$$\text{tform_rho_diff_back} = \text{neut_rho_diff_back} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.357)$$

$$\text{tform_temp_diff_back} = \text{neut_temp_diff_back} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.358)$$

$$\text{tform_salt_diff_back} = \text{neut_salt_diff_back} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.359)$$

This contribution arises from dianeutral mixing acting on temperature and salinity.

36.11.3.4 Vertical diffusion from internal tide mixing vertical diffusivity

We estimate the contribution from dianeutral mixing due to the internal tide mixing parameterization of [Simmons et al. \(2004\)](#) (see Chapter 20), with the diagnostic method following that discussed in Section

36.9.2.3. The following diagnostics are contained in this contribution.

$$\text{neut_rho_diff_wave} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt_wave}} \quad (36.360)$$

$$\text{neut_temp_diff_wave} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{diff_cbt_wave}} \quad (36.361)$$

$$\text{neut_salt_diff_wave} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt_wave}} \quad (36.362)$$

$$\text{wdian_rho_diff_wave} = \text{neut_rho_diff_wave} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.363)$$

$$\text{wdian_temp_diff_wave} = \text{neut_temp_diff_wave} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.364)$$

$$\text{wdian_salt_diff_wave} = \text{neut_salt_diff_wave} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.365)$$

$$\text{tform_rho_diff_wave} = \text{neut_rho_diff_wave} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.366)$$

$$\text{tform_temp_diff_wave} = \text{neut_temp_diff_wave} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.367)$$

$$\text{tform_salt_diff_wave} = \text{neut_salt_diff_wave} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.368)$$

This contribution arises from dianeutral mixing acting on temperature and salinity.

36.11.3.5 Vertical diffusion from coastal tide mixing vertical diffusivity

We estimate the contribution from dianeutral mixing due to the coastal tide mixing parameterization of [Lee et al. \(2006\)](#) (see Chapter 20), with the diagnostic method following that discussed in Section 36.9.2.3. The following diagnostics are contained in this contribution.

$$\text{neut_rho_diff_drag} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt_drag}} \quad (36.369)$$

$$\text{neut_temp_diff_drag} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{diff_cbt_drag}} \quad (36.370)$$

$$\text{neut_salt_diff_drag} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbt_drag}} \quad (36.371)$$

$$\text{wdian_rho_diff_drag} = \text{neut_rho_diff_drag} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.372)$$

$$\text{wdian_temp_diff_drag} = \text{neut_temp_diff_drag} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.373)$$

$$\text{wdian_salt_diff_drag} = \text{neut_salt_diff_drag} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.374)$$

$$\text{tform_rho_diff_drag} = \text{neut_rho_diff_drag} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.375)$$

$$\text{tform_temp_diff_drag} = \text{neut_temp_diff_drag} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.376)$$

$$\text{tform_salt_diff_drag} = \text{neut_salt_diff_drag} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.377)$$

This contribution arises from dianeutral mixing acting on temperature and salinity.

36.11.3.6 Vertical diffusion from leewave induced vertical diffusivity

We estimate the contribution from dianeutral mixing due to a preliminary implementation of the leewave induced mixing scheme of [Nikurashin and Ferrari \(2010\)](#), with the diagnostic method following that discussed in Section 36.9.2.3. The following diagnostics are contained in this contribution.

$$\text{neut_rho_diff_lee} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbl_leewave}} \quad (36.378)$$

$$\text{neut_temp_diff_lee} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{diff_cbl_leewave}} \quad (36.379)$$

$$\text{neut_salt_diff_lee} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{diff_cbl_leewave}} \quad (36.380)$$

$$\text{wdian_rho_diff_lee} = \text{neut_rho_diff_lee} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.381)$$

$$\text{wdian_temp_diff_lee} = \text{neut_temp_diff_lee} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.382)$$

$$\text{wdian_salt_diff_lee} = \text{neut_salt_diff_lee} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.383)$$

$$\text{tform_rho_diff_lee} = \text{neut_rho_diff_lee} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.384)$$

$$\text{tform_temp_diff_lee} = \text{neut_temp_diff_lee} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.385)$$

$$\text{tform_salt_diff_lee} = \text{neut_salt_diff_lee} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.386)$$

This contribution arises from dianeutral mixing acting on temperature and salinity.

36.11.3.7 Vertical diffusion from K33-implicit

We estimate the contribution from the time-implicit K33 term in a manner discussed in Section 36.9.2.3, with the following diagnostics containing this contribution.

$$\text{neut_rho_k33} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{k33}} \quad (36.387)$$

$$\text{neut_temp_k33} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{k33}} \quad (36.388)$$

$$\text{neut_salt_k33} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{k33}} \quad (36.389)$$

$$\text{wdian_rho_k33} = \text{neut_rho_k33} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.390)$$

$$\text{wdian_temp_k33} = \text{neut_temp_k33} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.391)$$

$$\text{wdian_salt_k33} = \text{neut_salt_k33} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.392)$$

$$\text{tform_rho_k33} = \text{neut_rho_k33} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.393)$$

$$\text{tform_temp_k33} = \text{neut_temp_k33} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.394)$$

$$\text{tform_salt_k33} = \text{neut_salt_k33} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.395)$$

This field can be somewhat noisy in appearance. However, when combined the time-explicit portion of the neutral diffusion operator (see Section 36.11.4.1), their sum is far smoother. See Section 36.11.4 for more discussion about neutral diffusion. Nonetheless, we provide the `vmix` diagnostic as a means to check that the individual diagnostics of K33 and `diff_cbt` are correct.

36.11.3.8 Vertical diffusion from dianeutral mixing plus K33-implicit

We estimate the combined contribution from dianeutral mixing and K33-implicit in a manner discussed in Section 36.9.2.3, with the following diagnostics containing this contribution.

$$\text{neut_rho_vmix} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{vmix}} \quad (36.396)$$

$$\text{wdian_rho_vmix} = \text{neut_rho_vmix} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.397)$$

$$\text{tform_rho_vmix} = \text{neut_rho_vmix} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.398)$$

Physically, these two diffusion processes are not related. Instead, the K33-implicit term is associated with neutral diffusion. It should therefore be diagnosed alone and then added to the time-explicit portion of the neutral diffusion operator as discussed in Section 36.11.4.

36.11.3.9 Vertical mixing from convective adjustment schemes

$$\text{neut_rho_convect} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{convect adjust}} \quad (36.399)$$

$$\text{neut_temp_convect} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{convect adjust}} \quad (36.400)$$

$$\text{neut_salt_convect} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{convect adjust}} \quad (36.401)$$

$$\text{wdian_rho_diff_convect} = \text{neut_rho_diff_convect} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.402)$$

$$\text{wdian_temp_diff_convect} = \text{neut_temp_diff_convect} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.403)$$

$$\text{wdian_salt_diff_convect} = \text{neut_salt_diff_convect} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.404)$$

$$\text{tform_rho_convect} = \text{neut_rho_convect} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.405)$$

$$\text{tform_temp_convect} = \text{neut_temp_convect} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.406)$$

$$\text{tform_salt_convect} = \text{neut_salt_convect} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.407)$$

This contribution arises from the vertical mixing associated with convective adjustment schemes, such as that of [Rahmstorf \(1993\)](#).

36.11.3.10 Nonlocal KPP transport

$$\text{neut_rho_kpp_nloc} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{kpp nonlocal}} \quad (36.408)$$

$$\text{neut_temp_kpp_nloc} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{kpp nonlocal}} \quad (36.409)$$

$$\text{neut_salt_kpp_nloc} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{kpp nonlocal}} \quad (36.410)$$

$$\text{wdian_rho_kpp_nloc} = \text{neut_rho_kpp_nloc} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.411)$$

$$\text{wdian_temp_kpp_nloc} = \text{neut_temp_kpp_nloc} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.412)$$

$$\text{wdian_salt_kpp_nloc} = \text{neut_salt_kpp_nloc} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.413)$$

$$\text{tform_rho_kpp_nloc} = \text{neut_rho_kpp_nloc} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.414)$$

$$\text{tform_temp_kpp_nloc} = \text{neut_temp_kpp_nloc} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.415)$$

$$\text{tform_salt_kpp_nloc} = \text{neut_salt_kpp_nloc} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.416)$$

This contribution arises from the nonlocal transport portion from KPP ([Large et al., 1994](#)).

36.11.3.11 Diagnostic checks for vertical processes

In all ocean regions, we should have the following results respected to within numerical truncation errors

$$\text{neut_rho_sbc} = \text{neut_rho_sbc_temp} + \text{neut_rho_sbc_salt} \quad (36.417)$$

$$\text{neut_rho_vdiffuse} = \text{neut_rho_vmix} + \text{neut_rho_sbc} + \text{neut_rho_bbc_temp} \quad (36.418)$$

$$\text{neut_rho_vmix} = \text{neut_rho_diff_cbt} + \text{neut_rho_k33}, \quad (36.419)$$

with the same equalities also holding when binned to density layers. Additionally, the same relations hold for the `wdian` and `tform` diagnostics, again both on model levels and binned to density layers.

36.11.4 Neutral diffusion

This suite of diagnostics arises from the time-explicit portion of the neutral diffusion operator and various versions of this operator. To get the full effects from neutral diffusion, we should in addition add the time-implicit portion detailed in Section 36.11.3.7.

36.11.4.1 Neutral diffusion operator: time-explicit piece

$$\text{neut_rho_ndiff} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{ndiffuse}} \quad (36.420)$$

$$\text{neut_temp_ndiff} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{ndiffuse}} \quad (36.421)$$

$$\text{neut_salt_ndiff} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{ndiffuse}} \quad (36.422)$$

$$\text{wdian_rho_ndiff} = \text{neut_rho_ndiff} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.423)$$

$$\text{wdian_temp_ndiff} = \text{neut_temp_ndiff} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.424)$$

$$\text{wdian_salt_ndiff} = \text{neut_salt_ndiff} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.425)$$

$$\text{tform_rho_ndiff} = \text{neut_rho_ndiff} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.426)$$

$$\text{tform_temp_ndiff} = \text{neut_temp_ndiff} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.427)$$

$$\text{tform_salt_ndiff} = \text{neut_salt_ndiff} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.428)$$

This contribution arises from the time-explicit portion of neutral diffusion. In the ocean interior, neutral diffusion contributes to the evolution of locally referenced potential density via cabbeling and thermobaricity (Section 36.7.1). In regions of steep neutral directions, MOM generally converts neutral diffusion to horizontal diffusion as per the recommendations of [Treguier et al. \(1997\)](#); [Ferrari et al. \(2008, 2010\)](#). Horizontal diffusion next to boundaries generally contributes to significant dianeutral transport, with this transport leading to either an increase or decrease in density, depending on local gradients. The full effects from the neutral diffusion operator are obtained by adding the `ndiff` diagnostic to the `k33` diagnostic detailed in Section 36.11.3.7.

36.11.4.2 Cabbelling in the ocean interior

$$\text{neut_rho_cabbelling} = \rho A_n \mathcal{C} |\nabla_n \Theta|^2 \quad (36.429)$$

$$\text{wdian_cabbelling} = \frac{\rho dz dA}{dz(\rho_{,\Theta} \Theta_{,z} + \rho_{,S} S_{,z})} \left(\rho A_n \mathcal{C} |\nabla_n \Theta|^2 \right) \quad (36.430)$$

$$\text{tform_rho_cabbell_on_nrho} = \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma} \right) (A_n \rho \mathcal{C} |\nabla_\gamma \Theta|^2). \quad (36.431)$$

The diagnostics `neut_rho_cabbelling_on_nrho` and `wdian_cabbelling_on_nrho` are obtained by binning into neutral density classes. In general, this diagnostic computes the contribution from cabbelling in the ocean interior according to the analytical manipulations provided in Section 36.7.1, in particular equation (36.100). Regions of steep neutral slope and boundaries are omitted, since it is here that neutral diffusion transfers into horizontal diffusion.

36.11.4.3 Thermobaricity in the ocean interior

$$\text{neut_rho_thermob} = \rho A_n \mathcal{T} \nabla_n p \cdot \nabla_n \Theta \quad (36.432)$$

$$\text{wdian_thermob} = \frac{\rho dz dA}{dz(\rho_{,\Theta} \Theta_{,z} + \rho_{,S} S_{,z})} \left(\rho A_n \mathcal{T} \nabla_n p \cdot \nabla_n \Theta \right) \quad (36.433)$$

$$\text{tform_rho_thermb_on_nrho} = \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma} \right) (A_n \rho \mathcal{T} \nabla_\gamma p \cdot \nabla_\gamma \Theta). \quad (36.434)$$

The diagnostics `neut_rho_thermob_on_nrho` and `wdian_thermob_on_nrho` are obtained by binning into neutral density classes. In general, this diagnostic computes the contribution from thermobaricity in the ocean interior according to the manipulations provided in Section 36.7.1, in particular equation (36.100). Regions of steep neutral slope and boundaries are omitted, since it is here that neutral diffusion transfers into horizontal diffusion.

36.11.4.4 Diagnostic checks for neutral diffusion

In regions away from steep neutral slopes, we should have the approximate relations

$$\text{neut_rho_cabbelling} + \text{neut_rho_thermob} \approx \text{neut_rho_ndiff} \quad (36.435)$$

$$\text{wdian_cabbelling} + \text{wdian_thermob} \approx \text{wdian_rho_ndiff}. \quad (36.436)$$

These relations fail in regions of steep neutral slopes since the neutral diffusion process switches to horizontal diffusion in these regions. In steep neutral slope regions, the diagnostic terms `neut_rho_cabbelling`, `neut_rho_thermob`, `wdian_cabbelling`, and `wdian_thermob` are set to zero.

Splitting the neutral diffusion operator into a time-explicit piece and time-implicit piece makes diagnosing the net effects of this operator cumbersome. It is with this caveat in mind that the total effects of neutral diffusion, plus horizontal mixing in the steep sloped regions, is approximated via

$$\text{neutral diffusion plus steep slope horiz diffusion} \approx \text{neutral_rho_ndiff} + \text{neut_rho_k33}. \quad (36.437)$$

Besides being approximate due to the time splitting of the neutral diffusion operator, this is an approximate relation due to splitting the K33 piece from other pieces of the time-implicit inversion (Section 36.9.2). Nonetheless, experience has shown that both time truncation errors are not critical to the use of this diagnostic.

36.11.5 Submesoscale horizontal diffusion

If enabling the submesoscale parameterization of [Fox-Kemper et al. \(2008b\)](#), and enabling the horizontal diffusive component to this parameterization as detailed in Section 24.6, we may evaluate the following

contribution

$$\text{neut_rho_subdiff} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{submeso diffuse}} \quad (36.438)$$

$$\text{neut_temp_subdiff} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{submeso diffuse}} \quad (36.439)$$

$$\text{neut_salt_subdiff} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{submeso diffuse}} \quad (36.440)$$

$$\text{wdian_rho_subdiff} = \text{neut_rho_subdiff} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.441)$$

$$\text{wdian_temp_subdiff} = \text{neut_temp_subdiff} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.442)$$

$$\text{wdian_salt_subdiff} = \text{neut_salt_subdiff} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.443)$$

$$\text{tform_rho_subdiff} = \text{neut_rho_subdiff} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.444)$$

$$\text{tform_temp_subdiff} = \text{neut_temp_subdiff} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.445)$$

$$\text{tform_salt_subdiff} = \text{neut_salt_subdiff} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.446)$$

The same equalities also hold when binning the diagnostics to density layers. Additionally, the same relations hold for the wdian and tform diagnostics, again both on model levels and binned to density layers.

36.11.6 Quasi-physical parameterizations of overflow and marginal sea exchange

This suite of diagnostics arises from the the various means of representing/parameterizing overflow processes and to connect the open ocean with spuriously land-locked marginal seas.

36.11.6.1 Over-exchange scheme

$$\text{neut_rho_overex} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{overexch}} \quad (36.447)$$

$$\text{neut_temp_overex} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{overexch}} \quad (36.448)$$

$$\text{neut_salt_overex} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{overexch}} \quad (36.449)$$

$$\text{wdian_rho_overex} = \text{neut_rho_overex} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.450)$$

$$\text{wdian_temp_overex} = \text{neut_temp_overex} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.451)$$

$$\text{wdian_salt_overex} = \text{neut_salt_overex} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.452)$$

$$\text{tform_rho_overex} = \text{neut_rho_overex} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.453)$$

$$\text{tform_temp_overex} = \text{neut_temp_overex} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.454)$$

$$\text{tform_salt_overex} = \text{neut_salt_overex} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.455)$$

This contribution arises from one of the schemes available in MOM to parameterize processes associated with deep overflows.

36.11.6.2 Overflow scheme

$$\text{neut_rho_overfl} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{overflow}} \quad (36.456)$$

$$\text{neut_temp_overfl} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{overflow}} \quad (36.457)$$

$$\text{neut_salt_overfl} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{overflow}} \quad (36.458)$$

$$\text{wdian_rho_overfl} = \text{neut_rho_overfl} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.459)$$

$$\text{wdian_temp_overfl} = \text{neut_temp_overfl} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.460)$$

$$\text{wdian_salt_overfl} = \text{neut_salt_overfl} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.461)$$

$$\text{tform_rho_overfl} = \text{neut_rho_overfl} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.462)$$

$$\text{tform_temp_overfl} = \text{neut_temp_overfl} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.463)$$

$$\text{tform_salt_overfl} = \text{neut_salt_overfl} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.464)$$

This contribution arises from one of the schemes available in MOM to parameterize processes associated with deep overflows.

36.11.6.3 Overflow scheme from NCAR

$$\text{neut_rho_overflow} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{overflow_ofp}} \quad (36.465)$$

$$\text{neut_temp_overflow} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{overflow_ofp}} \quad (36.466)$$

$$\text{neut_salt_overflow} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{overflow_ofp}} \quad (36.467)$$

$$\text{wdian_rho_overflow} = \text{neut_rho_overflow} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.468)$$

$$\text{wdian_temp_overflow} = \text{neut_temp_overflow} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.469)$$

$$\text{wdian_salt_overflow} = \text{neut_salt_overflow} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.470)$$

$$\text{tform_rho_overflow} = \text{neut_rho_overflow} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.471)$$

$$\text{tform_temp_overflow} = \text{neut_temp_overflow} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.472)$$

$$\text{tform_salt_overflow} = \text{neut_salt_overflow} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.473)$$

This contribution arises from one of the schemes available in MOM to parameterize processes associated with deep overflows. It is based on the overflow scheme of [Danabasoglu et al. \(2010\)](#).

36.11.6.4 Mixdownslope scheme

$$\text{neut_rho_mixdown} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{mixdown}} \quad (36.474)$$

$$\text{neut_temp_mixdown} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{mixdown}} \quad (36.475)$$

$$\text{neut_salt_mixdown} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{mixdown}} \quad (36.476)$$

$$\text{wdian_rho_mixdown} = \text{neut_rho_mixdown} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.477)$$

$$\text{wdian_temp_mixdown} = \text{neut_temp_mixdown} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.478)$$

$$\text{wdian_salt_mixdown} = \text{neut_salt_mixdown} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.479)$$

$$\text{tform_rho_mixdown} = \text{neut_rho_mixdown} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.480)$$

$$\text{tform_temp_mixdown} = \text{neut_temp_mixdown} \left(\frac{B \rho dV}{\Delta \gamma^n} \right) \quad (36.481)$$

$$\text{tform_salt_mixdown} = \text{neut_salt_mixdown} \left(\frac{B \rho dV}{\Delta \gamma^n} \right). \quad (36.482)$$

This contribution arises from one of the schemes available in MOM to parameterize processes associated with deep overflows.

36.11.6.5 Sigma diffusion scheme

$$\text{neut_rho_sigma} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{sigma}} \quad (36.483)$$

$$\text{neut_temp_sigma} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{sigma}} \quad (36.484)$$

$$\text{neut_salt_sigma} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{sigma}} \quad (36.485)$$

$$\text{wdian_rho_sigma} = \text{neut_rho_sigma} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.486)$$

$$\text{wdian_temp_sigma} = \text{neut_temp_sigma} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.487)$$

$$\text{wdian_salt_sigma} = \text{neut_salt_sigma} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.488)$$

$$\text{tform_rho_sigma} = \text{neut_rho_sigma} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.489)$$

$$\text{tform_temp_sigma} = \text{neut_temp_sigma} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.490)$$

$$\text{tform_salt_sigma} = \text{neut_salt_sigma} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.491)$$

This contribution arises from one of the schemes available in MOM to parameterize processes associated with deep overflows. It is based on the diffusive portion of the [Beckmann and Döscher \(1997\)](#) scheme.

36.11.6.6 Cross land mixing scheme

$$\text{neut_rho_xm} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{xm}} \quad (36.492)$$

$$\text{neut_temp_xm} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{xm}} \quad (36.493)$$

$$\text{neut_salt_xm} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{xm}} \quad (36.494)$$

$$\text{wdian_rho_xm} = \text{neut_rho_xm} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.495)$$

$$\text{wdian_temp_xm} = \text{neut_temp_xm} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.496)$$

$$\text{wdian_salt_xm} = \text{neut_salt_xm} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.497)$$

$$\text{tform_rho_xm} = \text{neut_rho_xm} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.498)$$

$$\text{tform_temp_xm} = \text{neut_temp_xm} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.499)$$

$$\text{tform_salt_xm} = \text{neut_salt_xm} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.500)$$

This contribution arises from the parameterization of mixing across land-locked marginal seas ([Griffies et al., 2005](#)), where the land-locking arises from the coarse resolution of the model grid.

36.11.6.7 Cross land insertion scheme

$$\text{neut_rho_xinsert} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{xinsert}} \quad (36.501)$$

$$\text{neut_temp_xinsert} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{xinsert}} \quad (36.502)$$

$$\text{neut_salt_xinsert} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{xinsert}} \quad (36.503)$$

$$\text{wdian_rho_xinsert} = \text{neut_rho_xinsert} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.504)$$

$$\text{wdian_temp_xinsert} = \text{neut_temp_xinsert} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.505)$$

$$\text{wdian_salt_xinsert} = \text{neut_salt_xinsert} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.506)$$

$$\text{tform_rho_xinsert} = \text{neut_rho_xinsert} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.507)$$

$$\text{tform_temp_xinsert} = \text{neut_temp_xinsert} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.508)$$

$$\text{tform_salt_xinsert} = \text{neut_salt_xinsert} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.509)$$

This contribution arises from the parameterization of mixing across land-locked marginal seas ([Griffies et al., 2005](#)), where the land-locking arises from the coarse resolution of the model grid.

36.11.7 Miscellaneous schemes

The following diagnostics are associated with miscellaneous processes.

36.11.7.1 Frazil heating of ocean liquid

As liquid seawater in MOM is influenced by surface boundary fluxes and transport, it may become colder than the freezing point of ice. In this case, the seawater is warmed back to the freezing point, with the heat required for this warming provided by the ice model. This adjustment process is known as *frazil formation*, as that is what the ice model does to the super-cooled seawater. From the ocean model perspective, the formation of frazil ice is a warming. Alternatively, it can be thought of as a re-partitioning of the surface boundary fluxes (cooling fluxes in this case) between the liquid ocean and sea ice.

$$\text{neut_rho_frazil} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{frazil}} \quad (36.510)$$

$$\text{wdian_rho_frazil} = \text{neut_rho_frazil} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.511)$$

$$\text{tform_rho_frazil} = \text{neut_rho_frazil} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.512)$$

36.11.7.2 Free surface or bottom pressure smoothing

$$\text{neut_rho_smooth} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) + \rho_{,S} \partial_t (S \rho dz) \right)_{\text{smooth}} \quad (36.513)$$

$$\text{neut_temp_smooth} = \frac{1}{\rho dz} \left(\rho_{,\Theta} \partial_t (\Theta \rho dz) \right)_{\text{smooth}} \quad (36.514)$$

$$\text{neut_salt_smooth} = \frac{1}{\rho dz} \left(\rho_{,S} \partial_t (S \rho dz) \right)_{\text{smooth}} \quad (36.515)$$

$$\text{wdian_rho_smooth} = \text{neut_rho_smooth} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.516)$$

$$\text{wdian_temp_smooth} = \text{neut_temp_smooth} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.517)$$

$$\text{wdian_salt_smooth} = \text{neut_salt_smooth} \left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \quad (36.518)$$

$$\text{tform_rho_smooth} = \text{neut_rho_smooth} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.519)$$

$$\text{tform_temp_smooth} = \text{neut_temp_smooth} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \quad (36.520)$$

$$\text{tform_salt_smooth} = \text{neut_salt_smooth} \left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right). \quad (36.521)$$

This contribution arises from the process of smoothing either the free surface (for Boussinesq simulations) or bottom pressure (for non-Boussinesq simulations) in order to reduce the B-grid checkerboard noise (see, e.g., [Killworth et al. \(1991\)](#) and [Griffies et al. \(2001\)](#)).

36.11.8 Summary of the process method for the ESM2M ocean

As for the kinematic method summarized in Section 36.10.13, we take the convention that each term is diagnosed as an Eulerian time tendency, using the same numerical operations as used for the prognostic equations of conservative temperature and salinity. We present here a specific example based on the ocean component of the ESM2M earth system model documented by [Dunne et al. \(2012, 2013\)](#).

36.11.8.1 Material time derivative

The finite volume estimate of the advective-form of the process material time derivative is given by

$$\left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{process level}} \approx$$

$$\begin{aligned} & \text{neut_rho_pbl_pme_pr} + \text{neut_rho_pbl_rn_pr} + \text{neut_rho_pbl_cl_pr} \\ & + \text{neut_rho_sw} + \text{neut_rho_sbc_temp} + \text{neut_rho_sbc_salt} \\ & + \text{neut_rho_diff_cbt} + \text{neutral_rho_kpp_nloc} \\ & + \text{neut_rho_ndiff} + \text{neut_rho_k33} \\ & + \text{neut_rho_mixdown} + \text{neut_rho_sigma} \\ & + \text{neut_rho_xmix} + \text{neut_rho_xinsert} \\ & + \text{neut_rho_smooth} + \text{neut_rho_frazil} \\ & + \text{neut_rho_bbc_temp} \end{aligned} \quad (36.522)$$

where we assumed liquid plus solid runoff are split into their own separate arrays, as per a realistic climate model with a liquid runoff and solid calving scheme. If these terms are combined, as commonly done in

ocean-ice models, we should instead diagnose

$$\text{neut_rho_pbl_rn_pr} + \text{neut_rho_pbl_cl_pr} \rightarrow \text{neut_rho_pbl_rv_pr}. \quad (36.523)$$

The same relations also hold for the density binned form

$$\left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right)_{\text{process layer}} \approx$$

$$\begin{aligned} & \text{neut_rho_pbl_pme_pr_on_nrho} + \text{neut_rho_pbl_rn_pr_on_nrho} + \text{neut_rho_pbl_cl_pr_on_nrho} \\ & + \text{neut_rho_sw_on_nrho} + \text{neut_rho_sbc_temp_on_nrho} + \text{neut_rho_sbc_salt_on_nrho} \\ & + \text{neut_rho_diff_cbt_on_nrho} + \text{neut_rho_kpp_nloc_on_nrho} \\ & + \text{neut_rho_ndiff_on_nrho} + \text{neut_rho_k33_on_nrho} \\ & + \text{neut_rho_mixdown_on_nrho} + \text{neut_rho_sigma_on_nrho} \\ & + \text{neut_rho_xmix_on_nrho} + \text{neut_rho_xinsert_on_nrho} \\ & + \text{neut_rho_smooth_on_nrho} + \text{neut_rho_frazil_on_nrho} \\ & + \text{neut_rho_bbc_temp_on_nrho}. \end{aligned} \quad (36.524)$$

36.11.8.2 Dianeutral transport from wdian diagnostics

The dianeutral mass transport, in units of kg s^{-1} , can be estimated using `wdian` both on levels and binned to neutral density layers. Following from the `neutral_rho` diagnostics above, we have

$$\left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right)_{\text{process level}} \approx$$

$$\begin{aligned} & \text{wdian_rho_pbl_pme_pr} + \text{wdian_rho_pbl_rn_pr} + \text{wdian_rho_pbl_cl_pr} \\ & + \text{wdian_rho_sw} + \text{wdian_rho_sbc_temp} + \text{wdian_rho_sbc_salt} \\ & + \text{wdian_rho_diff_cbt} + \text{wdian_rho_kpp_nloc} \\ & + \text{wdian_rho_ndiff} + \text{wdian_rho_k33} \\ & + \text{wdian_rho_mixdown} + \text{wdian_rho_sigma} \\ & + \text{wdian_rho_xmix} + \text{wdian_rho_xinsert} \\ & + \text{wdian_rho_smooth} + \text{wdian_rho_frazil} \\ & + \text{wdian_rho_bbc_temp}. \end{aligned} \quad (36.525)$$

The same relations also hold for the density binned dianeutral mass transport

$$\left(\frac{\rho dA_{(h)}}{|\rho_{,\Theta} \Theta_{,h} + \rho_{,S} S_{,h}|} \right) \left(\rho_{,\Theta} \frac{d^+\Theta}{dt} + \rho_{,S} \frac{d^+S}{dt} \right)_{\text{process layer}} \approx$$

$$\begin{aligned} & \text{wdian_rho_pbl_pme_pr_on_nrho} + \text{wdian_rho_pbl_rn_pr_on_nrho} + \text{wdian_rho_pbl_cl_pr_on_nrho} \\ & + \text{wdian_rho_sw_on_nrho} + \text{wdian_rho_sbc_temp_on_nrho} + \text{wdian_rho_sbc_salt_on_nrho} \\ & + \text{wdian_rho_diff_cbt_on_nrho} + \text{wdian_rho_kpp_nloc_on_nrho} \\ & + \text{wdian_rho_ndiff_on_nrho} + \text{wdian_rho_k33_on_nrho} \\ & + \text{wdian_rho_mixdown_on_nrho} + \text{wdian_rho_sigma_on_nrho} \\ & + \text{wdian_rho_xmix_on_nrho} + \text{wdian_rho_xinsert_on_nrho} \\ & + \text{wdian_rho_smooth_on_nrho} + \text{wdian_rho_frazil_on_nrho} \\ & + \text{wdian_rho_bbc_temp_on_nrho}. \end{aligned} \quad (36.526)$$

36.11.8.3 Dianeutral transport from tform diagnostics

The dianeutral mass transport, in units of kg s^{-1} , can also be estimated using `tform` diagnostics. Here, the results are only sensible when binned to neutral density layers. Following from the `wdian_rho` diagnostics

above, we have

$$\left(\frac{\mathcal{B} \rho dV}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} \frac{d^+ \Theta}{dt} + \rho_{,S} \frac{d^+ S}{dt} \right)_{\text{process layer}} \approx$$

$$\begin{aligned} & \text{tform_rho_pbl_pme_pr_on_nrho} + \text{tform_rho_pbl_rn_pr_on_nrho} + \text{tform_rho_pbl_cl_pr_on_nrho} \\ & + \text{tform_rho_sw_on_nrho} + \text{tform_rho_sbc_temp_on_nrho} + \text{tform_rho_sbc_salt_on_nrho} \\ & + \text{tform_rho_diff_cbt_on_nrho} + \text{tform_rho_kpp_nloc_on_nrho} \\ & + \text{tform_rho_ndiff_on_nrho} + \text{tform_rho_k33_on_nrho} \\ & + \text{tform_rho_mixdown_on_nrho} + \text{tform_rho_sigma_on_nrho} \\ & + \text{tform_rho_xmix_on_nrho} + \text{tform_rho_xinsert_on_nrho} \\ & + \text{tform_rho_smooth_on_nrho} + \text{tform_rho_frazil_on_nrho} \\ & + \text{tform_rho_bbc_temp_on_nrho}. \end{aligned} \quad (36.527)$$

36.12 Budget for locally referenced potential density

The diagnostics detailed in Sections 36.10 and 36.11 were used to obtain estimates of the advective-form for the material time derivative of locally referenced potential density, along with the associated dianeutral transport. Additionally, we have the kinematic formulation diagnosed in its flux-form (as used for the prognostic model equations), which is distinct from the advective-form in the surface grid cell (see Sections 36.10.5, 36.10.7, 36.10.9, 36.10.11). We may thus obtain the budget (i.e., time tendency) for locally referenced potential density. The following expression holds for the ESM2M coupled climate model where river water is split into liquid runoff and solid calving

$$\begin{aligned} \text{neut_rho_tendency} = & \text{neut_rho_advect} + \text{neut_rho_gm} + \text{neut_rho_submeso} \\ & + \text{neut_rho_pme} + \text{neut_rho_runoffmix} + \text{neut_rho_calvingmix} \\ & + \text{neut_rho_sw} + \text{neut_rho_sbc_temp} + \text{neut_rho_sbc_salt} \\ & + \text{neut_rho_diff_cbt} + \text{neut_rho_kpp_nloc} \\ & + \text{neut_rho_ndiff} + \text{neut_rho_k33} \\ & + \text{neut_rho_mixdown} + \text{neut_rho_sigma} \\ & + \text{neut_rho_xmix} + \text{neut_rho_xinsert} \\ & + \text{neut_rho_smooth} + \text{neut_rho_frazil} \\ & + \text{neut_rho_bbc_temp}. \end{aligned} \quad (36.528)$$

This budget is also available binned to neutral density classes through the on_nrho version of the above terms.

36.13 Diagnosing mass budgets for density layers

The purpose of this section is to detail the diagnostic methods available in MOM calculation of the accumulated formation equation (36.8), repeated here for convenience

$$\begin{aligned}
 \Phi(\gamma) &\equiv \int_{\gamma}^{\gamma_b} \mathcal{F}(\gamma) \delta\gamma. \\
 &= \int_{\gamma}^{\gamma_b} \left(\nu^{\dagger} + \frac{\partial \mathcal{M}}{\partial t} \right) \\
 &= \mathcal{G}(\gamma) - \mathcal{G}(\gamma_b) + \int_{\gamma}^{\gamma_b} \mathcal{E}.
 \end{aligned} \tag{36.529}$$

Note that the caveats regarding neutral density given in Section 36.5.2 are relevant for general density mass budgets. We ignore those details here. However, such details, as well as the more leading order issues of just how to define the binning operation (detailed in this section), make the layer diagnostics far less robust than the level diagnostics.

36.13.1 Time tendency for layer mass, $\mathcal{M}(\gamma)$

Once the mass of a layer is estimated, the time tendency can be computed as the difference across a chosen time step. It turns out that this calculation can be the source of some noise. As detailed in Section 36.13.1, the noise can be ameliorated in two different manners. First, we diagnose the tendency by taking the time mean of all forcing terms acting on the right hand side of the mass equation (36.1), and interpreting the masses as living on the temporal interface of the time average period. That is, the masses comprising the diagnosed time tendency live at half-integer time steps. The second means for computing the time tendency is to take the difference in the time averaged later mass (e.g., difference in annual means). The second difference will result in a tendency that is offset by a half-integer from the forcing terms. In a quasi-steady state, both tendencies will agree, and be small.

In general, the layer mass will evolve in time. Only by considering long time means in a simulation with modest drift will the mass of each layer remain close to constant. So when diagnosing the mass balance (36.1), or the streamfunction (36.7), it is important to determine how large the mass tendency is relative to the other terms.

There are two general approaches for estimating the mass of a layer, $\mathcal{M}(\gamma)$. Briefly, the first approach estimates, via linear interpolation, the lower and upper interfaces of the layer. The second approach bins the mass of a grid cell according to its neutral density class. Multiplication by the horizontal area yields the volume of that portion of the layer, and summing over the horizontal region within the domain of interest then leads to the total volume for the layer. If working in a pressure based non-Boussinesq model, the thickness of concern is actually the mass per area of the layer. The two approaches agree quite well, with differences arising in the case of unstably stratified waters, since the interpolation approach assumes stable stratification. We detail these methods in this section. We also expose a limitation of the binning approach used to measure layer masses, and propose a means to reduce the problems associated with this limitation.

36.13.1.1 Time averaging the time tendency

We start by noting a trivial point relevant for diagnosing the time tendency of layer mass. Namely, the time average of the layer mass time tendency is the difference in layer mass between the initial time step and the final time step

$$\frac{1}{T} \int_{-T/2}^{T/2} dt \left(\frac{\partial \mathcal{M}(\gamma)}{\partial t} \right) = \frac{\mathcal{M}(\gamma, T/2) - \mathcal{M}(\gamma, -T/2)}{T}, \tag{36.530}$$

where T is the length of the time interval, $\mathcal{M}(\gamma, \pm T/2)$ is the mass of the γ layer at time steps $\pm T/2$. The resulting time averaged tendency thus measures the mass difference between final and initial time steps. Depending on the layer population, this diagnosed time tendency can be very noisy. Indeed, it can be so noisy as to be of little use. The noise arises since computing the difference in the mass of a bin, where the bin class is constant in time, is an intrinsically noisy procedure, whereby mass moves around in bin space in a discrete manner subject to the arbitrary choice of bin classes. In particular, when filling an empty bin, there is a huge change in the mass over just a small time step, and this change corresponds to unphysically huge mass transport. Such huge jumps in mass arise solely from the arbitrary fixed bin classes, and do not reflect a physically relevant transfer of water into different layers.

36.13.1.2 Noise in the tendency of binned mass

Consider the layer mass equation (36.1), rewritten here in the generic form

$$\frac{\partial \mathcal{M}(t)}{\partial t} = \mathcal{H}(t), \quad (36.531)$$

where \mathcal{H} is shorthand for the forcing terms on the right hand side of equation (36.1). Because of the identity (36.530), the time averaged forcing, $\overline{\mathcal{H}}(\tau)$, forces a difference in the instantaneous layer mass. However, computation of the difference in the instantaneous layer mass using fixed density bins generally results in very noisy results for the left hand side to equation (36.531). In contrast, the right hand side forcing, $\overline{\mathcal{H}}(\tau)$, is generally far smoother. Hence, one cannot expect the time average of both sides to equation (36.531) to be equal when one uses fixed bins to diagnose both sides.

Fundamental to the problem of noise for the left hand side to equation (36.531) is that the mass contained within a particular density bin will jump when using fixed density bins. For example, if a bin originally was empty, the smooth diffusive transport of mass into that bin will result, at a particular time step, in a huge jump in the bin mass over a single time step. Taking the temporal difference of the bin mass will then lead to a huge jump in the mass tendency. Such jumps have no relevance to the otherwise smooth rearrangement of mass between layers. Instead, it arises as an artifact of the use of fixed density bins.

Nonetheless, we wish to use fixed bins in order to allow for the online watermass diagnostic to be useful across a wide suite of model processes. We thus need a means to smooth the binning noise. We choose a temporal averaging operator for this purpose, with the approach motivated from that used in the barotropic time stepping scheme used in MOM4 (see, for example, Section 12.7 of Griffies (2004)).

36.13.1.3 A smoothed mass tendency

Introduce the symbol Δt to denote the time step used for an update to model prognostic fields (i.e., the prognostic model tracer time step), with t the associated time label. Distinguish the model time step Δt from the generally longer time step $\Delta \tau \geq \Delta t$ associated with time averaged diagnostics, which are typically taken as daily, monthly, annual, or longer time means. The relation between the model time step and the diagnostic time step is given by

$$t_n = \tau + n \Delta t, \quad (36.532)$$

where $n = 0, N$ is the discrete model time step, N is the number of model time steps per diagnostic time step, and τ is the diagnostic time label. With these conventions, introduce the time averaged forcing term appearing on the right hand side to equation (36.531)

$$\overline{\mathcal{H}}(\tau + \Delta \tau) = \frac{1}{N} \sum_{n=1}^N \mathcal{H}(\tau + n \Delta t), \quad (36.533)$$

where the time average is performed according to the specified diagnostic time step; i.e., day, month, year, etc. Next, we *define* a diagnostic layer mass centered on a half integer diagnostic time step, $\widetilde{\mathcal{M}}(\tau + \Delta \tau/2)$, according to

$$\frac{\widetilde{\mathcal{M}}(\tau + \Delta \tau/2) - \widetilde{\mathcal{M}}(\tau - \Delta \tau/2)}{\Delta \tau} \equiv \overline{\mathcal{H}}(\tau). \quad (36.534)$$

That is, the half integer diagnostic layer mass $\widetilde{\mathcal{M}}(\tau + \Delta\tau/2)$ is forced by the time averaged forcing $\overline{\mathcal{H}}(\tau)$. Given the time averaged forcing $\overline{\mathcal{H}}(\tau)$, and an initial condition, we can time step the half integer layer mass if so desired. A useful initial condition is given by setting the initial half-integer mass equal to the integer mass at the first diagnostic time step. Note that since the forcing $\overline{\mathcal{H}}(\tau)$ is smooth, so too is the time tendency $(\widetilde{\mathcal{M}}(\tau + \Delta\tau/2) - \widetilde{\mathcal{M}}(\tau - \Delta\tau/2))/\Delta\tau$.

A check that the prescription (36.534) produces a sensible result is to compare the tendency to that obtained by taking the difference between integer time step mean layer masses according to

$$\frac{\overline{\mathcal{M}}(\tau + \Delta\tau) - \overline{\mathcal{M}}(\tau)}{\Delta\tau} \equiv \widetilde{\mathcal{H}}(\tau + \Delta\tau/2), \quad (36.535)$$

where

$$\overline{\mathcal{M}}(\tau + \Delta\tau) = \frac{1}{N} \sum_{n=1}^N \mathcal{M}(\tau + n \Delta t), \quad (36.536)$$

is the time averaged mass, and the right hand side of equation (36.535) defines a half integer diagnostic forcing. In a steady state, the tendencies (36.534) and (36.535) will be the same. In the more general case, they will differ according to the evolution of mass within a layer class.

The prescription (36.534) provides an indirect means to the calculation of the layer mass tendency. That is, we infer the mass tendency through diagnosing the time mean forcing. The self-consistency of this inference is examined by checking that it is consistent with the tendency (36.535) deduced from the difference in time mean layer masses. In brief, we jettison the use of the difference in the instantaneous layer mass given by equation (36.530), as that tendency is far too noisy for diagnostic use.

In the following, we detail two approaches available for computing the instantaneous layer mass, $\mathcal{M}(t)$. Either approach can be used to compute the time averaged mass (36.536), whose tendency (36.536) should correspond, though not generally equal, to the tendency of the half integer mass (36.534).

36.13.1.4 Estimating layer mass via interpolation

One approach to computing the mass tendency is to estimate the layer thickness via interpolation to determine the lower and upper interfaces of the layer, upon which multiplication by the horizontal area of a grid cell yields the volume of that portion of the layer. Summing over the horizontal region within the domain of interest then leads to the total volume for the layer. If working in a pressure based non-Boussinesq model, the thickness of concern is actually the mass per area of the layer. Once the thickness is estimated, the time tendency is computed as the difference in mass of the layer across a model time step.

The following diagnostics are related to computing $\mathcal{M}(\gamma)$ and $\partial_t \mathcal{M}(\gamma)$, with each diagnostic computed inside the `ocean_tracer.F90` module.

- `mass_nrho_layer`: estimate of $\mathcal{M}(\gamma)$ (units of kg). This diagnostic computes the time mean of the layer mass, allowing for offline calculations to be used for computing time tendencies based on differences of the time means (e.g., save annual mean `mass_nrho_layer` and compute the time tendency by differencing annual mean masses).
- `mass_nrho_tendency_layer`: estimate of $\partial_t \mathcal{M}(\gamma)$ (units of kg s^{-1}). This diagnostic computes the time tendency at each time step and then performs a time average of the tendency. Because of the identity (36.530), this time tendency can be very noisy, so it is not very useful. Instead, the time tendencies (36.534 and (36.535) are more useful.

We offer the following comments regarding this diagnostic.

- **PRO A:** This algorithm appears to produce results that have little sensitivity to details of the pre-defined density classes, with finer density bins producing cleaner results.
- **PRO B:** When the neutral density is stably stratified, the vertical sum of `mass_nrho_tendency_layer` equals to the mass entering through the boundaries. So this diagnostic respects mass conservation for stably stratified columns.

- **CON A:** The algorithm is based on a linear interpolation that is sensible only when the diagnosed neutral density layers are monotonically stacked in the vertical. In particular, the algorithm does not guarantee mass conservation when there are unstably stratified regions.

36.13.1.5 Binning the mass of a tracer cell to neutral density layers

A second method for computing the mass tendency of a density layer is to bin the mass of a tracer cell (or ρ_0 times the volume when working with a Boussinesq fluid) into neutral density classes. The evolution of this binned mass then determines the mass tendency for the layer. This method is not equivalent to the first method, since the binning approach is not equivalent to linear interpolation. The diagnostics available with this method include the following.

- **mass_t:** this is the mass of a tracer grid cell.
- **mass_t_on_nrho:** this is the mass of a tracer cell binned into neutral density layers. It thus provides an estimate of the mass of neutral density layers.
- **mass_t_tendency_on_nrho:** This is the time tendency of the layer mass, computed as the tendency of **mass_t_on_nrho**. This diagnostic computes the time tendency at each time step and then performs a time average of the tendency. Because of the identity (36.530), this time tendency can be very noisy, so it is not very useful. Instead, the time tendencies (36.534 and (36.535) are more useful.

We offer the following comments regarding this diagnostic approach.

- **PRO A:** Since we are binning the mass of a tracer cell according to density layers, the method works for arbitrary density stratification.
- **PRO B:** The vertical sum of **mass_t_tendency_on_nrho** equals to the mass entering through the boundaries. So this diagnostic respects mass conservation.
- **CON:** Binning generally produces noisy results when the density classes are sporadically populated, with finer density bins generally producing more noisy results. The results are therefore a function of the pre-defined density classes. However, performing a vertical indefinite integral, as when forming an accumulated formation streamfunction as in equation (36.8), will generally smooth the results.

36.13.1.6 Binning the time tendency for the mass per area (not recommended)

A spurious approach was discovered when working with the field **rho_dzt_tendency**. When multiplied by area of a grid cell, **rho_dzt_tendency** provides a measure of the mass tendency for a grid cell, with a nonzero tendency associated with mass convergence and divergence into the cell. However, this tendency is notably *not* what we aim to diagnose for the purpose of watermass transformation analysis. Indeed, **rho_dzt_tendency** is identically zero in a rigid lid geopotential model, whereas the neutral density layers certainly can evolve in such models. So again, it is inappropriate to consider mapping the area weighted **rho_dzt_tendency** to neutral density classes for the watermass transformation diagnostics.

36.13.1.7 Regarding water fluxes and vertical coordinates

An unsettling issue with the diagnosis of layer mass was uncovered when developing the diagnostics detailed in this section. Namely, consider the transfer of water across the ocean surface, absent interior mixing. In the case of a non-geopotential vertical coordinate, a water flux at the ocean surface will alter interior tracer concentrations, including salinity and temperature, even in the absence of mixing, so long as there is a nontrivial vertical stratification of the tracer concentration. This alteration of tracer concentration results from rearrangement (expansion or contraction) of the vertical coordinate surfaces due to changes in the surface height or bottom pressure.

A one-dimensional test case is sufficient to illustrate the issue. Consider for this purpose horizontally flat temperature surfaces with stable vertical stratification. Allow salinity to be uniform and the equation of state linear with just a temperature dependence. When passing water across the ocean surface, sea

level will change. For the geopotential vertical coordinate, it is just the surface grid cell that feels the sea level change. However, for all other MOM4 vertical coordinates, each cell at depth will experience the sea level change as well. In particular, as sea level changes, so too does the mass per area (non-Boussinesq) or volume per area (Boussinesq). A nonzero vertical velocity will be established to conserve mass for each grid cell. This vertical velocity will in turn advect tracer. As the salinity is uniform to begin with, there is no change in salinity concentration throughout the vertical, except for the top grid cell, where the impacts of the surface water flux impact the surface salinity. However, with a nonzero stratification in temperature, there will be advective transport of temperature, and so the temperature field will be modified.

There are various unsettling aspects to this test case. Notably, the temperature stratification has been modified throughout the full depth, yet we have no interior mixing. This alteration of temperature is unrelated to the spurious mixing from numerical advection described in [Griffies et al. \(2000b\)](#), where advective truncation errors generally become more egregious as eddy activity is enhanced. Instead, all that has occurred is to transfer fresh water across the ocean surface in this one-dimensional test case. The stratification change arises from motion of the coordinate surfaces, and the associated nonzero vertical advection velocity developed to conserve mass within each grid cell. The vertical velocity then leads to a nonzero advective temperature convergences, and hence to changes in temperature.

We acknowledge the somewhat unphysical nature to the above test case. Namely, the absence of mixing is unrealistic. Motion of water across the ocean surface generally occurs in the presence of nontrivial vertical mixing in the upper ocean. In particular, phase change for evaporating water and mixing of precipitating fresh water into a salty ocean are both irreversible. So the case of a geopotential model with its upper grid cell remaining isolated from static deeper cells becomes unphysical, especially in the case of a very fine upper cell thickness. Nonetheless, given these caveats, the test case presents an unsettling example of how the choice of vertical coordinates can directly impact the level of effective tracer mixing within a simulation, even when the flow regime is trivially laminar.

36.13.2 Surface mass transport, $\mathcal{E}(\gamma)$

The transport of water crossing the ocean surface, $\mathcal{E}(\gamma)$, is determined according to a binning of the surface water transport according to density classes. The following diagnostics are computed in the `ocean_sbc.F90` module.

- `mass_pmepr_on_nrho`: density binned net mass transport (units of kg s^{-1}) of water that crosses the ocean surface. A positive value indicates water enters the ocean.
- `mass_precip_on_nrho`: density binned mass transport (units of kg s^{-1}) of precipitating liquid plus precipitating frozen water crossing the ocean surface, including water exchanged with sea ice (i.e., melt). A positive value indicates water enters the ocean.
- `mass_evap_on_nrho`: density binned mass transport (units of kg s^{-1}) of evaporating vapor or condensing liquid water crossing the ocean surface. A positive value indicates water enters the ocean.
- `mass_river_on_nrho`: density binned mass transport (units of kg s^{-1}) of liquid river runoff and solid calving land ice entering the ocean. A positive value indicates water enters the ocean.
- `mass_melt_on_nrho`: density binned mass transport (units of kg s^{-1}) of water exchanged with the sea ice model through melting or forming sea ice. A positive value indicates water enters the ocean. Again, note that melt is also included as part of precipitation.

In general, the full mass flux into a density layer through the ocean surface is given by

$$\mathcal{E} = \text{mass_pmepr_on_nrho} \quad (36.537)$$

and note that the following identity holds for these diagnostics:

$$\text{mass_pmepr_on_nrho} = \text{mass_precip_on_nrho} + \text{mass_evap_on_nrho} + \text{mass_river_on_nrho}, \quad (36.538)$$

where, again, ice melt is included as part of precipitation.

36.13.3 Overturning streamfunction, $\Psi^\dagger(\gamma) = -\int_\gamma^{\gamma_b} \mathcal{V}^\dagger$

The overturning streamfunction is a diagnostic that is commonly computed in ocean simulations. Indeed, analysis of Ψ^\dagger as computed on model levels as well as binned into density layers is becoming quite common, as the two perspectives provide useful complementary information. There are presently three terms that contribute to the overturning streamfunction in MOM, with the two terms associated with eddy induced transport treated differently than the one term associated with the model resolved Eulerian transport. When computing terms in the accumulated watermass formation equation (36.8), it is important to recall the minus sign relation (36.14)

$$\Psi^\dagger(y, \gamma) = - \int_\gamma^{\gamma_b} \mathcal{V}^\dagger(y). \quad (36.539)$$

We start the diagnostic calculation by considering the level space specification of the meridional streamfunction Ψ^\dagger , which takes the form

$$\Psi^\dagger(s) = - \int_{\text{zonal range}} dx \int_{-H}^{z(s)} dz \rho (v + v_{\text{gm}}^* + v_{\text{submeso}}^*). \quad (36.540)$$

In this equation, the zonal integration occurs over a specified periodic or closed domain, such as the full longitudinal extent of the Southern Ocean or the region between two continents. Vertical integration occurs from the ocean bottom at a position $z = -H(x, y)$, up to the depth of the coordinate surface $z(s)$, where s is the value of the generalized level for each model level. The *in situ* density factor ρ renders a mass transport for mass conserving non-Boussinesq simulations, and it becomes the constant reference density ρ_o for volume conserving Boussinesq simulations. The meridional velocity component v is part of the model's resolved horizontal velocity field \mathbf{u} . We are most interested in meridional transport, but the overturning streamfunction can also be defined for zonal transport, in which case u replaces v .

There are two eddy induced velocity fields displayed in equation (36.540), with \mathbf{u}_{gm}^* associated with the Gent et al. (1995) parameterization of mesoscale eddies, and $\mathbf{u}_{\text{submeso}}^*$ associated with the Fox-Kemper et al. (2008b) parameterization of mixed layer submesoscale eddies. Both eddy induced velocity fields can be written as the vertical derivative of an eddy induced transport Υ (units of $\text{m}^2 \text{s}^{-1}$), so that

$$\begin{aligned} \rho \mathbf{u}_{\text{gm}}^* &= \partial_z (\rho \Upsilon_{\text{gm}}) \\ \rho \mathbf{u}_{\text{submeso}}^* &= \partial_z (\rho \Upsilon_{\text{submeso}}), \end{aligned} \quad (36.541)$$

where the ρ factors reduce to the constant reference density ρ_o for Boussinesq fluids. Hence, the contribution from the eddy induced transport to the overturning streamfunction takes the form

$$\begin{aligned} \Psi^{\text{eddy}}(s) &= - \int_{\text{zonal range}} dx \int_{-H}^{z(s)} dz \rho (v_{\text{gm}}^* + v_{\text{submeso}}^*) \\ &= - \int_{\text{zonal range}} dx \int_{-H}^{z(s)} dz \partial_z \left(\rho \Upsilon_{\text{gm}}^{(y)} + \rho \Upsilon_{\text{submeso}}^{(y)} \right) \\ &= - \int_{\text{zonal range}} dx \left(\rho \Upsilon_{\text{gm}}^{(y)} + \rho \Upsilon_{\text{submeso}}^{(y)} \right)_{z=z(s)}. \end{aligned} \quad (36.542)$$

In this equation, we wrote $\Upsilon^{(y)}$ for the meridional component of the eddy induced transport. This transport vanishes at the ocean bottom (and surface), thus leaving just the contribution from the interior transport at a depth $z = z(s)$. Given these results, we may write the full overturning streamfunction (36.540) in the form

$$\Psi^\dagger(s) = - \int_{\text{zonal range}} dx \int_{-H}^{z(s)} dz \rho v - \int_{\text{zonal range}} dx \left(\rho \Upsilon_{\text{gm}}^{(y)} + \rho \Upsilon_{\text{submeso}}^{(y)} \right)_{z=z(s)}. \quad (36.543)$$

The following diagnostics are computed in the module `ocean_adv_vel_diag.F90` for the overturning due to the Eulerian velocity.

- `tx_trans_nrho`: zonal mass transport through a grid cell, $\rho dy dz u$ (units of kg s^{-1}) binned according to neutral density classes.
- `ty_trans_nrho`: meridional mass transport through a grid cell $\rho dx dz v$ (units of kg s^{-1}) binned according to neutral density classes.

The following diagnostics are computed in the module `ocean_nphysics_util.F90` for the overturning due to the [Gent et al. \(1995\)](#) eddy induced velocity.

- `tx_trans_nrho_gm`: zonal eddy induced mass transport through a grid cell $\rho dy \Upsilon_{gm}^{(x)}$ (units of kg s^{-1}) mapped according to neutral density classes.
- `ty_trans_nrho_gm`: meridional eddy induced mass transport through a grid cell $\rho dx \Upsilon_{gm}^{(y)}$ (units of kg s^{-1}) mapped according to neutral density classes.

The following diagnostics are computed in the module `ocean_submesoscale1.F90` for the overturning due to the [Fox-Kemper et al. \(2008b\)](#) eddy induced velocity.

- `tx_trans_nrho_submeso`: zonal eddy induced mass transport through a grid cell $\rho dy \Upsilon_{submeso}^{(x)}$ (units of kg s^{-1}) mapped according to neutral density classes.
- `ty_trans_nrho_submeso`: meridional eddy induced mass transport through a grid cell $\rho dx \Upsilon_{submeso}^{(y)}$ (units of kg s^{-1}) mapped according to neutral density classes.

A calculation of the full meridional overturning streamfunction in Ferret takes the form

$$\begin{aligned} \Psi^+(\gamma) &= \text{ty_trans_nrho}[k = @rsum, i = @sum] - \text{ty_trans_nrho}[k = @sum, i = @sum] \\ &\quad + \text{ty_trans_nrho_gm}[i = @sum] + \text{ty_trans_nrho_submeso}[i = @sum] \\ \Psi^+(y, \gamma) &= - \int_{\gamma}^{\gamma_b} \Psi^+(y) d\gamma. \end{aligned} \tag{36.544}$$

Note that absence of a vertical sum for the two eddy induced transports, since their respective vertical integrals have been performed analytically according to equation (36.542). Subtracting the term

$$\text{ty_trans_nrho}[k = @sum, i = @sum] \tag{36.545}$$

accounts for the convention in Ferret whereby it integrates from the surface downward, rather than from the bottom upward. This step is required to ensure that the diagnosed streamfunction `psi` vanishes at the ocean bottom. Other analysis software, such as Matlab, may not require this added step, so long as the integration starts from zero at the bottom and goes from the bottom upwards.

We have had mixed success with the structure of the transports `ty_trans_nrho_gm` and `ty_trans_nrho_submeso` resulting from the MOM diagnostic calculation. One may find the following Ferret command as a useful alternative for either the [Gent et al. \(1995\)](#) or [Fox-Kemper et al. \(2008b\)](#) transport computed from MOM:

$$\text{let ty_trans_nrho_gm.new} = \text{ZAXREPLACE}(\text{ty_trans_gm}, \text{neutral_rho}, \text{ty_trans_nrho}), \tag{36.546}$$

where `ty_trans_gm` is the level-space version of the [Gent et al. \(1995\)](#) transport, `neutral_rho` is the diagnosed neutral density at each model grid point, and `ty_trans_nrho` provides Ferret with the density classes upon which to remap `ty_trans_gm`. The Ferret function `ZAXREPLACE` replaces the vertical axis of `ty_trans_gm` with the vertical axis from `ty_trans_nrho` according to the neutral density values in `neutral_rho`. There will be differences between `ty_trans_nrho_gm.new` and `ty_trans_nrho_gm` due to sampling. But in principle they should look quite simple. In practice, however, they may differ depending on other details that are not clear. It is thus useful to compare the two versions of the eddy induced overturning.

36.13.4 watermass transformation, $\mathcal{G}(\gamma)$

By definition, the transport crossing a density class is given by the surface integral (see equation (36.69))

$$\mathcal{G}(\gamma) = \int_{\mathcal{A}(\gamma)} \rho w^{(\gamma)} dA_{(\gamma)}. \quad (36.547)$$

The area integral extends over the density surface γ for the region south of the chosen latitude (see Figure 36.1). The algorithm detailed by Table 36.1 provides the means to compute the integral on the right hand side of equation (36.547).

Following the form given by equation (36.529), we need to compute the

$$\mathcal{G}(\gamma) - \mathcal{G}(\gamma_b) = \int_{\mathcal{A}(\gamma)} \rho w^{(\gamma)} dA_{(\gamma)} - \int_{\mathcal{A}(\gamma_b)} \rho w^{(\gamma_b)} dA_{(\gamma_b)}. \quad (36.548)$$

Either the kinematic or process methods can be used to compute contributions to this integral, thus facilitating physical interpretations of the causes for the transformation. For a realistic global model with a full suite of parameterizations, we have the following identity for arbitrary density interfaces, as represented using the kinematic approach (see Section 36.10.13)

$$\begin{aligned} & \left(\rho w^{(\gamma)} dA_{(\gamma)} - \rho w^{(\gamma_b)} dA_{(\gamma_b)} \right)_{\text{kinematic}} \approx \\ & \text{wdian_rho_tendency_on_nrho} - \text{wdian_rho_advect_on_nrho} \\ & - \text{wdian_rho_gm_on_nrho} - \text{wdian_rho_submeso_on_nrho} \\ & + \text{wdian_rho_pbl_pme_kn_on_nrho} \\ & + \text{wdian_rho_pbl_rn_kn_on_nrho} + \text{wdian_rho_pbl_cl_kn_on_nrho} \\ & - \text{wdian_rho_bbc_temp_on_nrho}. \end{aligned} \quad (36.549)$$

Note the subtraction of the geothermal heating term, which arises from the accumulated formation (see equation (36.529)). The equivalent process version, again for the ESM2M ocean component (Dunne et al., 2012, 2013) discussed in Section 36.11.8, is given by

$$\begin{aligned} & \left(\rho w^{(\gamma)} dA_{(\gamma)} - \rho w^{(\gamma_b)} dA_{(\gamma_b)} \right)_{\text{process}} \approx \\ & \text{wdian_rho_pbl_pme_pr_on_nrho} + \text{wdian_rho_pbl_rn_pr_on_nrho} + \text{wdian_rho_pbl_cl_pr_on_nrho} \\ & + \text{wdian_rho_sw_on_nrho} + \text{wdian_rho_sbc_temp_on_nrho} + \text{wdian_rho_sbc_salt_on_nrho} \\ & + \text{wdian_rho_diff_cbt_on_nrho} + \text{wdian_rho_kpp_nloc_on_nrho} \\ & + \text{wdian_rho_ndiff_on_nrho} + \text{wdian_rho_k33_on_nrho} \\ & + \text{wdian_rho_mixdown_on_nrho} + \text{wdian_rho_sigma_on_nrho} \\ & + \text{wdian_rho_xmix_on_nrho} + \text{wdian_rho_xinsert_on_nrho} \\ & + \text{wdian_rho_smooth_on_nrho} + \text{wdian_rho_frazil_on_nrho}. \end{aligned} \quad (36.550)$$

Note the absence of the geothermal heating term, since it is subtracted through the accumulated formation (see equation (36.529)). Another estimate of these transformations can be obtained through the tform diagnostic, in which case we have the kinematic expression

$$\begin{aligned} & \left(\rho w^{(\gamma)} dA_{(\gamma)} - \rho w^{(\gamma_b)} dA_{(\gamma_b)} \right)_{\text{kinematic}} \approx \\ & \text{tform_rho_tendency_on_nrho} - \text{tform_rho_advect_on_nrho} \\ & - \text{tform_rho_gm_on_nrho} - \text{tform_rho_submeso_on_nrho} \\ & + \text{tform_rho_pbl_pme_kn_on_nrho} \\ & + \text{tform_rho_pbl_rn_kn_on_nrho} + \text{tform_rho_pbl_cl_kn_on_nrho} \\ & - \text{tform_rho_bbc_temp_on_nrho} \end{aligned} \quad (36.551)$$

as well as the process expression

$$\left(\rho w^{(\gamma)} dA_{(\gamma)} - \rho w^{(\gamma_b)} dA_{(\gamma_b)} \right)_{\text{process}} \approx$$

$$\begin{aligned} & \text{tform_rho_pbl_pme_pr_on_nrho} + \text{tform_rho_pbl_rn_pr_on_nrho} + \text{tform_rho_pbl_cl_pr_on_nrho} \\ & + \text{tform_rho_sw_on_nrho} + \text{tform_rho_sbc_temp_on_nrho} + \text{tform_rho_sbc_salt_on_nrho} \\ & + \text{tform_rho_diff_cbt_on_nrho} + \text{tform_rho_kpp_nloc_on_nrho} \\ & + \text{tform_rho_ndiff_on_nrho} + \text{tform_rho_k33_on_nrho} \\ & + \text{tform_rho_mixdown_on_nrho} + \text{tform_rho_sigma_on_nrho} \\ & + \text{tform_rho_xmix_on_nrho} + \text{tform_rho_xinsert_on_nrho} \\ & + \text{tform_rho_smooth_on_nrho} + \text{tform_rho_frazil_on_nrho}. \end{aligned}$$

(36.552)

36.14 Inferring transformation from surface buoyancy fluxes

Many applications of the [Walín \(1982\)](#) layer transformation method have focused on the surface buoyancy fluxes, binned into density classes, and using these binned fluxes as a means to infer the interior watermass transformation ([Large and Nurser, 2001](#)). Returning to the layer mass equation (36.1), this approach is based on diagnosing the mass tendency $\partial \mathcal{M}(\gamma)/\partial t$ (usually assumed to be negligible in most applications); the surface mass flux $\mathcal{E}(\gamma)$ arising from precipitation, evaporation, runoff and ice melt; and contributions to the transformation $\mathcal{G}(\gamma)$ associated with surface buoyancy fluxes. From this information, one may infer interior values for the watermass transformation $\mathcal{G}(\gamma)$.

The inferential approach is unnecessary when using the deductive methods detailed in the previous sections, whereby explicit formulae for $\mathcal{G}(\gamma)$ were presented. However, the deductive approach is not available with observational data, where interior transport processes are not diagnosed to the degree required to perform a fully deductive watermass transformation analysis. Additionally, the inferential approach is a useful means for checking the integrity of the deductive diagnostics. Hence, we provide in this section a discussion of the boundary flux approach, and detail the MOM diagnostics available with this method.

This section summarizes the diagnostics available in MOM to determine how surface buoyancy fluxes affect a material change in density, when formulating the material change in the advective-form and using the process perspective. We are concerned here with fluxes binned according to neutral density classes, since we are interested in how the density census is modified through surface fluxes. All of these diagnostics are named

$$\text{tform_rho_pbl_name_on_nrho}, \quad (36.553)$$

where name denotes the associated flux, and pbl signifies that these terms are associated with the surface planetary boundary layer. Buoyancy fluxes associated with surface mass fluxes have already been introduced in Section 36.11.2. They are also included here for completeness.

36.14.1 Density forcing associated with surface water fluxes

The following diagnostics are associated with surface buoyancy fluxes arising from the passage of water across the ocean boundary. As water transported across the boundary carries both heat and salt (usually zero salinity), it contributes a nonzero buoyancy flux. The diagnostics are each computed in the modules

ocean_tracers/ocean_tracer.F90
ocean_param/sources/ocean_rivermix.F90

- `tform_rho_pbl_pme_pr_on_nrho`: This diagnostic, computed in the module `ocean_tracer.F90`, provides the process form contribution (see Section 36.11.2) due to the precipitation (liquid and frozen)

minus evaporation (units of kg s^{-1}). It is also presented in Section 36.11.2.1.

$$\text{tform_rho_pbl_pme_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{pme}} - \Theta_{k=1}) + \rho_{,S} (S_{\text{pme}} - S_{k=1}) \right) Q_{\text{pme}} \quad (36.554)$$

$$\text{tform_temp_pbl_pme_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{pme}} - \Theta_{k=1}) \right) Q_{\text{pme}} \quad (36.555)$$

$$\text{tform_salt_pbl_pme_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,S} (S_{\text{pme}} - S_{k=1}) \right) Q_{\text{pme}} \quad (36.556)$$

- `tform_rho_pbl_rv_pr_on_nrho`: This diagnostic, computed in the module `ocean_rivermix.F90`, provides the process contribution (see Section 36.11.2) due to liquid river runoff plus solid calving land ice (units of kg s^{-1}). It is also presented in Section 36.11.2.2.

$$\text{tform_rho_pbl_rv_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{river}} - \Theta_{k=1}) + \rho_{,S} (S_{\text{river}} - S_{k=1}) \right) Q_{\text{river}} \quad (36.557)$$

$$\text{tform_temp_pbl_rv_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{river}} - \Theta_{k=1}) \right) Q_{\text{river}} \quad (36.558)$$

$$\text{tform_salt_pbl_rv_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,S} (S_{\text{river}} - S_{k=1}) \right) Q_{\text{river}} \quad (36.559)$$

- `tform_rho_pbl_rn_pr_on_nrho`: This diagnostic, computed in the module `ocean_rivermix.F90`, provides the process contribution (see Section 36.11.2) due to liquid river runoff (units of kg s^{-1}). It is also presented in Section 36.11.2.3.

$$\text{tform_rho_pbl_rn_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{runoff}} - \Theta_{k=1}) + \rho_{,S} (S_{\text{runoff}} - S_{k=1}) \right) Q_{\text{runoff}} \quad (36.560)$$

$$\text{tform_temp_pbl_rn_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{runoff}} - \Theta_{k=1}) \right) Q_{\text{runoff}} \quad (36.561)$$

$$\text{tform_salt_pbl_rn_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,S} (S_{\text{runoff}} - S_{k=1}) \right) Q_{\text{runoff}} \quad (36.562)$$

- `tform_rho_pbl_cl_pr_on_nrho`: This diagnostic, computed in the module `ocean_rivermix.F90`, provides the process contribution (see Section 36.11.2) due to solid calving land ice (units of kg s^{-1}). It is also presented in Section 36.11.2.4.

$$\text{tform_rho_pbl_cl_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{calving}} - \Theta_{k=1}) + \rho_{,S} (S_{\text{calving}} - S_{k=1}) \right) Q_{\text{calving}} \quad (36.563)$$

$$\text{tform_temp_pbl_cl_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} (\Theta_{\text{calving}} - \Theta_{k=1}) \right) Q_{\text{calving}} \quad (36.564)$$

$$\text{tform_salt_pbl_cl_pr_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,S} (S_{\text{calving}} - S_{k=1}) \right) Q_{\text{calving}} \quad (36.565)$$

36.14.2 Density forcing associated with surface heat and salt fluxes

The following diagnostics are associated with surface buoyancy fluxes arising from the passage of salt and heat across the ocean. All fluxes are positive when entering the ocean (positive heat flux adds heat to the ocean; positive salt flux adds salt to the ocean). Each of the following diagnostics are computed inside the module

`ocean_core/ocean_sbc.F90`

- `tform_rho_pbl_flux_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to the net surface radiative and turbulent heat fluxes, plus salt fluxes due to sea ice interactions (units of kg s^{-1})

$$\text{tform_rho_pbl_flux_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\theta} (Q_{\text{sw}}^\Theta + Q_{\text{lw}}^\Theta + Q_{\text{latent}}^\Theta + Q_{\text{sensible}}^\Theta) + \rho_{,s} Q_{\text{sea ice}}^S \right). \quad (36.566)$$

This diagnostic is equivalent to `tform_rho_sbc_on_nrho` detailed in Section 36.11.1.3

$$\text{tform_rho_pbl_flux_on_nrho} = \text{tform_rho_sbc_on_nrho}. \quad (36.567)$$

- `tform_rho_pbl_heat_on_nrho`: This diagnostic provides the contribution due to the net surface radiative and turbulent heat fluxes (units of kg s^{-1})

$$\text{tform_rho_pbl_heat_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\theta} (Q_{\text{sw}}^\Theta + Q_{\text{lw}}^\Theta + Q_{\text{latent}}^\Theta + Q_{\text{sensible}}^\Theta) \right). \quad (36.568)$$

This diagnostic is equivalent to `tform_rho_sbc_temp_on_nrho` detailed in Section 36.11.1.1

$$\text{tform_rho_pbl_heat_on_nrho} = \text{tform_rho_sbc_temp_on_nrho}. \quad (36.569)$$

- `tform_rho_pbl_salt_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to the net surface salt fluxes due to sea ice interactions (units of kg s^{-1})

$$\text{tform_rho_pbl_salt_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,s} Q_{\text{sea ice}}^S \right). \quad (36.570)$$

This diagnostic is equivalent to `tform_rho_sbc_temp_on_nrho` detailed in Section 36.11.1.2

$$\text{tform_rho_pbl_salt_on_nrho} = \text{tform_rho_sbc_salt_on_nrho}. \quad (36.571)$$

- `tform_rho_pbl_lw_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to the net longwave radiation crossing the ocean surface (units of kg s^{-1}).

$$\text{tform_rho_pbl_lw_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\theta} Q_{\text{lw}}^\Theta \right). \quad (36.572)$$

- `tform_rho_pbl_lat_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to latent heating (due to liquid-vapor and solid-liquid phase changes) crossing the ocean surface (units of kg s^{-1}). Note that in some regions, the air temperature may drop below the wet bulb temperature, thus giving condensation rather than evaporation, and so causing latent heating to not be sign definite.

$$\text{tform_rho_pbl_lat_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\theta} Q_{\text{lat}}^\Theta \right). \quad (36.573)$$

- `tform_rho_pbl_sens_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to latent heating (due to liquid-vapor and solid-liquid phase changes) crossing the ocean surface (units of kg s^{-1}).

$$\text{tform_rho_pbl_sens_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\theta} Q_{\text{sens}}^\Theta \right). \quad (36.574)$$

- `tform_rho_pbl_sw_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to the net solar shortwave crossing the ocean surface (units of kg s^{-1}). This diagnostic does *not* consider penetrative shortwave effects

$$\text{tform_rho_pbl_sw_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\theta} Q_{\text{sw}}^\Theta \right). \quad (36.575)$$

- `tform_rho_pbl_adjheat_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to the net surface temperature restoring fluxes (units of kg s^{-1})

$$\text{tform_rho_pbl_adjheat_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,\Theta} Q_{\text{restore}}^{\Theta} \right). \quad (36.576)$$

- `tform_rho_pbl_adjsalt_on_nrho`: This diagnostic, computed in the module `ocean_sbc.F90`, provides the contribution due to the net surface salt restoring fluxes (units of kg s^{-1})

$$\text{tform_rho_pbl_adjsalt_on_nrho} = \left(\frac{\mathcal{B} dA}{\Delta \gamma^n} \right) \left(\rho_{,S} Q_{\text{restore}}^S \right). \quad (36.577)$$

36.15 Specifying the density classes for layer diagnostics

All diagnostics presented in “neutral density” classes are binned according to the user specified density layers. The neutral density at a model grid point is then used to determine where to bin the associated properties into neutral density space. The purpose of this subsection is to detail how the bins are determined and how the binning occurs.

36.15.1 Online calculation of neutral density

The neutral density computed online for the dianeutral transport diagnostic can be computed in one of the following ways.

- In early implementations of this diagnostic, the default approach followed the polynomial approximation given in the appendix to [McDougall and Jackett \(2005\)](#). This polynomial approximation is very convenient to use for an online model diagnostic. However, it suffers from inaccuracies that limit its utility. Under strong recommendation from Trevor McDougall, the MOM diagnostic presently defaults to potential density computed relative to 2000dbar as an approximation to the neutral density coordinate. Ongoing work aims to provide a more sensible global approximation to neutral density that is computable online without requiring a climatology.
- Another approach is to use potential density, which may be enabled by setting `neutral_density_potrho = .true.` and `neutral_density_mj = .false.` in the module `ocean_density.F90`. The pressure reference value is `potrho_press`, which is also a namelist in `ocean_density.F90`.

It remains a research question to determine an optimal online method for computing neutral density for dianeutral transport diagnostics. In particular, we are exploring the approach of [Klocker et al. \(2009\)](#).

36.15.2 Defining the density bins

The model provides a namelist in `ocean_density_nml` for the maximum and minimum value for the density class, and the number of bins included between these range boundaries. The following details these namelists.

- `neutralrho_min`: minimum neutral density for the binning (units kg m^{-3})
- `neutralrho_max`: maximum neutral density for the binning (units kg m^{-3})
- `layer_nk`: number of bins used for defining the neutral density classes.

These namelist settings are then used to setup `layer_nk` neutral density bins, whose boundaries or edges are denoted by γ^{bound} . The boundaries γ^{bound} correspond to the intervals $\delta\gamma/2$ shown in Figure 36.1. The `layer_nk+1` boundaries of the `layer_nk` bins are determined according to

$$\gamma^{\text{bound}}(k_{\gamma} = 1) = \text{neutralrho_min} \quad (36.578)$$

$$\gamma^{\text{bound}}(k_{\gamma}) = \gamma^{\text{bound}}(k_{\gamma} - 1) + \frac{\text{neutralrho_max} - \text{neutralrho_min}}{\text{layer_nk}} \quad \text{for } k_{\gamma} = 2, \text{layer_nk}+1. \quad (36.579)$$

For example, the CM2 climate models run at GFDL using MOM have 50 vertical grid levels. When binning according to σ_{2000} , we choose the following for its density bins

- `neutralrho_min` = 1028 kg m⁻³
- `neutralrho_max` = 1038 kg m⁻³
- `layer_nk` = 80.

Many choices for density range will not resolve processes that occur in certain inland seas (e.g., Baltic Sea or Hudson Bay) where fresh water makes the neutral density lighter than 1028 kg m⁻³. It may also miss certain deep processes with neutral density larger than 1038 kg m⁻³. So the user should choose the range based on the region of interest for the analysis.

The use of refined density classes will allow for finer resolution of transport in density space. But adding bins comes with a price. Furthermore, at some point there are reduced if not negative benefits. The cost is associated with the memory required for the online calculation (i.e., more bins means more memory to hold the binned fields) and archive storage for the binned output. The reduced and possibly negative benefits arise since adding more bins will reduce the population of any particular density class. At some point, with too many bins, the reduced population leads to noisy results.

36.15.3 Convention for the binning

The fields available in neutral density layers are a function of density, and written in the form $\mathcal{F}(\gamma)$. The discrete version of $\mathcal{F}(\gamma)$ is determined at each model time step according to the following binning procedure

$$\mathcal{F}(i, j, k_\gamma) = \sum_k \mathcal{F}(i, j, k) \Pi[\gamma^{\text{bound}}(k_\gamma) \leq \gamma(i, j, k) < \gamma^{\text{bound}}(k_\gamma + 1)] \quad (36.580)$$

where Π is a binning function that is unity when $\gamma(i, j, k)$ is inside the specified range, and vanishes when outside. There are two choices we may make for densities $\gamma(i, j, k)$ that fall outside of the maximum and minimum of the predefined density range.

- Populate the nearest bin. That is, if $\gamma(i, j, k) < \gamma^{\text{bound}}(k_\gamma = 1)$ (i.e., neutral density is lighter than any predefined density class), then $\mathcal{F}(i, j, k_\gamma = 1)$ is populated. If $\gamma(i, j, k) > \gamma^{\text{bound}}(k_\gamma = \text{layer_nk})$ (i.e., neutral density is denser than any predefined density class), then $\mathcal{F}(i, j, k_\gamma = \text{layer_nk})$ is populated.
- Ignore points outside of the predefined density range.

We choose the first method, where all cells are binned, even if falling outside of the bin range. This choice is consistent with the traditional approach used to bin transport in MOM, where it is important to bin all of the transport, including that which may occur for classes outside of the predefined range. However, this approach can lead to an overpopulation of the lightest or densest density class when there are waters that sit outside of the range. It is for this reason that we recommended that the maximum and minimum density bins both be removed from an analysis of the dianeutral transport diagnostic.

36.15.4 Binning versus remapping

When examining the results from transport binning, it is important to understand that binning is not the same as remapping. For example, consider a nonzero transport that has no depth dependence. When remapped to density space, the transport value will remain the same and there will be no density dependence, regardless of how the density layers are stratified. In contrast, when binned to density space, the binned transport will be a function of the density stratification. Namely, there will be more contribution to density bins that occupy a larger depth range, such as weakly stratified layers, and less contribution to those layers that are within a strongly stratified region. The result from binning is relevant when measuring the net mass transport within a particular density class.

36.16 Known limitations

The following summarizes some of the known limitations of this diagnostic framework.

36.16.1 Disagreements between *wdian* and *tform*

There are two general ways proposed in this chapter for diagnosing the dianeutral transformation of water: the *wdian* approach of Section 36.4 and the *tform* method of Section 36.5. Both approaches should agree, as they do in the continuum. However, discrete numerical truncations preclude agreement in the model. The disagreements can be quite large especially in regions of weak stratification, in which case the division by the density gradient required for the *wdian* approach can lead to unphysically large values of the associated transformation diagnostics. One means to regularize the *wdian* diagnostics is to smooth the stratification prior to computing the diagnostics. Such smoothing is indeed fundamental to how one estimates vertical mixing processes from observed data. Pursuit of such smoothing operations remains incomplete for the MOM *wdian* diagnostics.

In contrast, the *tform* method appears to be more robust than the *wdian* diagnostics, since there is no need to divide by the density gradient. Furthermore, agreement between the *tform* approach and the traditional overturning streamfunction Ψ (Section 36.2.4) is far better using the *tform* approach than the *wdian* approach. Consequently, in general we make more use of the *tform* diagnostics when aiming to perform quantitative estimates of water mass transformation and to close the mass budget detailed in Section 36.2.

36.16.2 Inserting river water into the ocean

Liquid runoff and solid calving are generally inserted into the upper few grid cells of a realistic ocean model. For example, in the ESM2M ocean component of Dunne et al. (2012, 2013), as well as the earlier CM2.1 ocean component of Delworth et al. (2006), the liquid and solid runoff is inserted into the top four model grid cells as detailed in Section 3.6 of Griffies et al. (2005). However, the kinematic formulation of the advective form, as presented in Sections 36.10.8, 36.10.10, 36.10.12, assumes all of the liquid and solid runoff are inserted to the top model grid cell. This assumption precludes the kinematic and process versions of the dianeutral transport from agreeing to numerical roundoff in those cells where runoff is actually inserted. Since the process diagnostic computes liquid and solid runoff contributions correctly, it is the preferred method for computing contributions to watermass transformation arising from runoff.

36.16.3 Difficulty closing the mass budgets

Section 36.13 summarizes the requirements for computing the mass budget according to density classes. Our attempts to close the mass budget remain unsuccessful. That is, the diagnostic of the mass transport does not agree with the diagnostic of the transformation. The following possible reasons for this disagreement require further research to gain clarity on the issues.

- The polynomial computation of neutral density is problematic. Even if it was perfect, there is a fundamental difference between locally referenced potential density and a polynomial neutral density variable. The differences necessitate the introduction of the \mathcal{B} factor in equation (36.77). We have yet to fully explore this approach. As discussed in Ludicone et al. (2008), details of the density binning can be of leading order importance, especially in the Southern Ocean.
- An alternative to neutral density is to simply use the potential density referenced to a chosen pressure level. We have pursued this approach, preferring 2000dbar reference level. Again, this choice leads to problems in the Southern Ocean, with those problems impacting all regions to the north.

MIXED LAYER DEPTH DIAGNOSTICS

Contents

37.1	The mixed layer depth	541
37.2	Tracer budgets within the mixed layer	542

The purpose of this chapter is to detail MOM5 diagnostics used to compute the mixed layer depth and budgets for tracers within the mixed layer. The following MOM module is directly connected to the material in this chapter:

ocean_diag/ocean_tracer_diag.F90

37.1 The mixed layer depth

The mixed layer depth is based on measuring the ocean stability in a manner similar to that used for the KPP boundary layer scheme (see Section 18.5.5). To determine whether vertical transfer is favored requires a thought experiment, in which a surface ocean fluid parcel is displaced downward without changing its temperature or salinity, but feeling the local *in situ* pressure. If the density of this displaced parcel is sufficiently far from the local *in situ* density, then the displacement is not favored, and we are thus beneath the mixed layer and into the stratified interior. What determines “sufficiently far” is subjective, with convention determining the precise value.

Mathematically, we compute the difference between the following two densities

$$\rho_{\text{displaced from surface}} = \rho[S(k=1), \Theta(k=1), p(k)] \quad (37.1a)$$

$$\rho_{\text{local}} = \rho[S(k), \Theta(k), p(k)], \quad (37.1b)$$

and convert that density difference to a buoyancy difference

$$\delta B = - \left(\frac{g(\rho_{\text{displaced from surface}} - \rho_{\text{local}})}{\rho_{\text{local}}} \right). \quad (37.2)$$

This buoyancy difference is computed from the surface down to the first depth at which $\delta B > \Delta B_{\text{crit}}$, where the MOM5 default is

$$\Delta B_{\text{crit}} = 0.0003 \text{ m s}^{-2}, \quad (37.3)$$

with this value also used in Conkright et al. (2002). The mixed layer depth, H_{mld} is then approximated by interpolating between the depth where $\delta B > \Delta B_{\text{crit}}$ and the shallower depth.

For purposes of computing budgets for tracers within the mixed layer, we compute the grid level where the mixed layer depth sits. Doing so provides a discrete approximation, K_{mld} , to the continuous mixed layer depth.

37.2 Tracer budgets within the mixed layer

SUBDUCTION DIAGNOSTICS

Contents

38.1	Kinematics of flow across a surface	543
38.1.1	General form of the dia-surface mass transport	543
38.1.2	Surface area elements	544
38.1.3	Oceanographic examples of surfaces	545
38.1.4	The dia-surface mass transport and dia-surface velocity component	546
38.1.5	Oceanographic examples of dia-surface mass transport	546
38.2	MOM subduction diagnostic calculation	548
38.2.1	Algorithm for subduction diagnostic	548
38.2.2	Available diagnostics	549

The purpose of this chapter is to detail diagnostic methods available online in MOM to compute the rate that seawater is transported across the mixed layer base, with notable applications of these methods to quantifying rates that water is transferred across the mixed layer base. The subduction rate is computed as in [Cushmin-Roisin \(1987\)](#), along with the option to bin this instantaneous rate into density classes as recommended by [Kwon et al. \(2013\)](#).

The following MOM module is directly connected to the material in this chapter:

ocean_diag/ocean_tracer_diag.F90

38.1 Kinematics of flow across a surface

Following the discussion of dia-surface transport in Section [36.3.2](#), we develop a kinematic framework to measure the rate that seawater crosses a surface, where the surface is generally moving. The surface of interest in this chapter is the mixed layer base, whereas in Chapter [36](#) we were concerned with density surfaces. The kinematic formulation is identical, so we follow the geometric presentation given in Section 6.7 of [Griffies \(2004\)](#), as well as Section 2.2 of [Griffies and Adcroft \(2008\)](#). Some of the material here can also be found in [Viúdez \(2000\)](#).

38.1.1 General form of the dia-surface mass transport

We assume that surfaces of interest can be specified, at least locally, by a function $s = s(x, y, z, t)$ (see Figure [38.1](#)). An orientation for the surface is specified by computing a normal direction

$$\hat{\mathbf{n}}(s) = \frac{\nabla s}{|\nabla s|}, \quad (38.1)$$

which points in the direction of increasing s . Given the velocity of a fluid parcel,

$$\mathbf{v} = (\mathbf{u}, w), \quad (38.2)$$

the *in situ* density ρ , and the velocity of a point on the surface, $\mathbf{v}^{(\text{ref})}$, we measure the mass per time of fluid moving across a surface by computing

$$\text{mass per time of fluid across surface} = \rho dA_{(s)} \hat{\mathbf{n}}(s) \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}), \quad (38.3)$$

where $dA_{(s)}$ is an infinitesimal area element on the surface. This fundamental definition provides a baseline from which to develop diagnostics for subduction (this chapter) and diapycnal transport (Chapter 36).

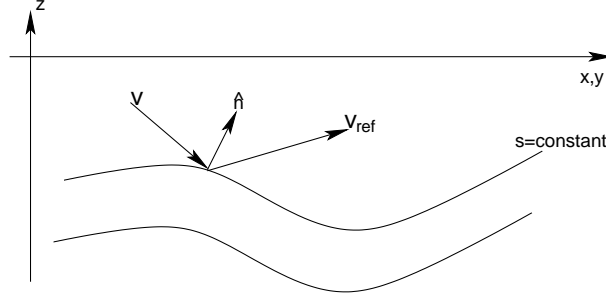


Figure 38.1: A schematic of an undulating surface, with normal direction $\hat{\mathbf{n}}$ indicated on one of the surfaces. Also shown is the orientation of the velocity of a fluid parcel \mathbf{v} and the velocity $\mathbf{v}^{(\text{ref})}$ of a reference point living on the surface.

38.1.2 Surface area elements

For any particular orientation of the normal direction $\hat{\mathbf{n}}(s)$, it is possible to project the local area element $dA_{(s)}$ onto at least one of the three coordinate planes perpendicular to the three unit directions $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, and $\hat{\mathbf{z}}$. This projection provides the means for a practical calculation of the area element. For example, if the field s is a density-like field that exhibits a stable vertical stratification, then $|\partial_z s|$ is nonzero. Indeed, it generally has the largest magnitude of the three spatial derivatives ($\partial_x s, \partial_y s, \partial_z s$). In this case, $\hat{\mathbf{n}}(s) \approx -\hat{\mathbf{z}}$ (density increasing downward), so the area element $dA_{(s)}$ is nearly equal to

$$dA_{(z)} = dx dy, \quad (38.4)$$

which is the area element in the plane perpendicular to the vertical direction $\hat{\mathbf{z}}$. Our intuition based on this common case in the ocean interior can be made precise by an expression from differential geometry that relates two area elements. For this purpose, make use of the equation (6.58) in Griffies (2004), in which we have the exact relation

$$dA_{(s)} = |\partial_s Z| |\nabla s| dA_{(z)}. \quad (38.5)$$

In this equation, we introduced the depth of an s -surface, which has the following functional dependence

$$Z = Z(x, y, s, t). \quad (38.6)$$

We choose the capital Z to distinguish the depth of a particular surface of constant s from the depth z of an arbitrary position in the ocean. Nonetheless, note that the inverse function is given by

$$\frac{\partial s}{\partial z} = \left(\frac{1}{\partial Z / \partial s} \right). \quad (38.7)$$

Again, for the highly stratified ocean interior with s chosen as a density-like field,

$$|\nabla s| \approx |\partial_z s| = |\partial Z / \partial s|^{-1}, \quad (38.8)$$

thus making $dA_{(s)} \approx dA_{(z)}$. The relation (38.5) holds in general, so long as the vertical derivative $\partial_z s$ remains nonzero. It thus provides for a general method to compute the area element $dA_{(s)}$ in regions of nonzero vertical stratification of s iso-surfaces.

In regions where $\partial_z s$ becomes tiny, such as the surface mixed layer for s based on density, then alternative measures for the area element must be used. We detail such cases in Section 36.3.2. The bottomline is one merely needs to consider area elements in such cases that are projected onto either of the two vertical planes spanned by the unit vectors (\hat{x}, \hat{z}) or (\hat{y}, \hat{z}) , rather than the horizontal plane spanned by (\hat{x}, \hat{y}) .

38.1.3 Oceanographic examples of surfaces

Consider the following examples of surfaces and their geometric specification.

- **OCEAN BOTTOM:** Let $Z = -h^{(\text{topo})}(x, y)$ represent the ocean bottom depth, which is assumed to be static and to possess no overturns. The bottom is geometrically specified through the relation

$$s(x, y, z) = z + h^{(\text{topo})}(x, y) = 0. \quad (38.9)$$

The corresponding normal vector is given by

$$\hat{\mathbf{n}}^{(\text{topo})} = \frac{\nabla(z + h^{(\text{topo})})}{|\nabla(z + h^{(\text{topo})})|}, \quad (38.10)$$

and the area element is (with $\partial_s Z = 1$ in equation (38.5))

$$dA_{(\text{topo})} = |\nabla(z + h^{(\text{topo})}(x, y))| dA_{(z)}. \quad (38.11)$$

- **FREE SURFACE:** Let $Z = \eta(x, y, t)$ represent the deviation of the ocean surface from a resting state at $Z = 0$. Writing the ocean surface as this function of horizontal position and time presumes an averaging has occurred to remove the more general overturns and breaking waves present on the real ocean surface. Having made this assumption, we can geometrically specify the free surface by the relation

$$s(x, y, z, t) = z - \eta(x, y, t) = 0. \quad (38.12)$$

The corresponding normal vector is given by

$$\hat{\mathbf{n}}^{(\eta)} = \frac{\nabla(z - \eta)}{|\nabla(z - \eta)|}, \quad (38.13)$$

and the area element is

$$dA_{(\eta)} = |\nabla(z - \eta)| dA_{(z)}. \quad (38.14)$$

- **MIXED LAYER BASE:** Let $Z = -h^{(\text{mld})}(x, y, t)$ represent the depth at the mixed layer base, determined, for example, by a density threshold criteria. As for the ocean free surface, the mixed layer base is geometrically specified by the relation

$$s(x, y, z, t) = z + h^{(\text{mld})}(x, y, t) = 0. \quad (38.15)$$

The corresponding normal vector is given by

$$\hat{\mathbf{n}}^{(\text{mld})} = \frac{\nabla(z + h^{(\text{mld})})}{|\nabla(z + h^{(\text{mld})})|}, \quad (38.16)$$

and the area element is

$$dA_{(\text{mld})} = |\nabla(z + h^{(\text{mld})})| dA_{(z)}. \quad (38.17)$$

- ISOPYCNAL DEPTH: Now let $Z = -h(x, y, \gamma, t)$ represent the depth of an isopycnal surface specified by a density γ . As a shorthand, we write $h(x, y, \gamma, t) = h^{(\gamma)}$. As for the mixed layer base, an isopycnal surface is geometrically specified by the relation

$$s(x, y, z, t) = z + h(x, y, \gamma, t) = 0. \quad (38.18)$$

The corresponding normal vector is given by

$$\hat{\mathbf{n}}^{(\gamma)} = \frac{\nabla(z + h^{(\gamma)})}{|\nabla(z + h^{(\gamma)})|}, \quad (38.19)$$

and the area element is

$$dA_{(\gamma)} = |\nabla(z + h^{(\gamma)})| dA_{(z)}. \quad (38.20)$$

38.1.4 The dia-surface mass transport and dia-surface velocity component

We now have an understanding of how to compute the area element $dA_{(s)}$ appearing in equation (38.3). Next, consider the normal projection of the surface reference velocity, which is given by (see Section 6.7 of Griffies (2004), as well as Section 2.2 of Griffies and Adcroft (2008))

$$\hat{\mathbf{n}}(s) \cdot \mathbf{v}^{(\text{ref})} = - \left(\frac{\partial s / \partial t}{|\nabla s|} \right). \quad (38.21)$$

That is, the normal projection is nonzero so long as there is a time dependence of the surface itself.

Combining the result (38.21) with the expression (38.5) for the area element $dA_{(s)}$, and (38.1) for the normal direction $\hat{\mathbf{n}}(s)$, leads to the mass transport expression

$$\begin{aligned} \text{mass per time of fluid across surface} &= \rho dA_{(s)} \hat{\mathbf{n}}(s) \cdot (\mathbf{v} - \mathbf{v}^{(\text{ref})}) \\ &= \rho dA_{(z)} \left| \frac{\partial Z}{\partial s} \right| \frac{ds}{dt}. \end{aligned}$$

(38.22)

This equation provides an explicit expression for the mass per time of fluid penetrating a locally defined surface or tangent plane. Again, this surface can generally be moving (i.e., $\mathbf{v}^{(\text{ref})}$ can generally be nonzero). Positive mass transport indicates fluid moving across the surface in the direction of $\hat{\mathbf{n}}(s)$; e.g., into denser regions for the case where s is locally referenced potential density. Following from the discussion in Section 36.3.2, we find it convenient to define the *dia-surface velocity component* according to

$$\begin{aligned} w^{(z)} &= \left| \frac{\partial Z}{\partial s} \right| \frac{ds}{dt} \\ &= \frac{(\text{VOLUME/TIME}) \text{ FLUID PENETRATING SURFACE, IN DIRECTION OF INCREASING } s}{\text{HORIZONTAL SURFACE AREA IN X-Y PLANE}}. \end{aligned} \quad (38.23)$$

In words, $w^{(z)}$ measures the volume per time of fluid penetrating a locally defined surface or tangent plane, as defined by the normal vector $\hat{\mathbf{n}}(s)$, divided by the horizontal projection of the local area on that surface. Positive $w^{(z)}$ indicates fluid moving across the surface in the direction of $\hat{\mathbf{n}}(s)$; e.g., into denser regions for the case where s is locally referenced potential density. Given this definition of $w^{(z)}$, the dia-surface mass transport is given by the tidy expression

$$\text{mass per time of fluid across surface} = \rho dA_{(z)} w^{(z)}.$$

(38.24)

38.1.5 Oceanographic examples of dia-surface mass transport

We return now to the examples from Section 38.1.3 to exhibit specific forms of the dia-surface mass transport.

- **OCEAN BOTTOM:** The ocean bottom with $Z = -h^{(\text{topo})}(x, y)$ is generally assumed to allow zero flow through the solid earth boundary. Hence, the dia-surface transport reduces to the no-normal flow boundary condition

$$\hat{\mathbf{n}}^{(\text{topo})} \cdot \mathbf{v} = 0 \quad \text{at } z = -H(x, y), \quad (38.25)$$

which takes on the following familiar form

$$\mathbf{u} \cdot \nabla H + w = 0 \quad \text{at } z = -H(x, y). \quad (38.26)$$

- **FREE SURFACE:** At the ocean free surface with $s(x, y, z, t) = z - \eta(x, y, t) = 0$, the dia-surface mass transport takes the form

$$-Q_m \equiv \rho dA_{(z)} \left(\frac{d(z - \eta)}{dt} \right) \quad \text{at } z = \eta(x, y, t), \quad (38.27)$$

where the surface mass transport Q_m (dimensions of mass per time), arising from precipitation, evaporation, and runoff, is defined to be positive for mass entering the ocean. Rearrangement of this result leads to a more familiar expression for the surface kinematic boundary condition

$$(\partial_t + \mathbf{u} \cdot \nabla) \eta = w + \frac{Q_m}{\rho dA_{(z)}} \quad \text{at } z = \eta(x, y, t). \quad (38.28)$$

- **MIXED LAYER BASE:** At the base of the mixed layer with $s(x, y, z, t) = z + h^{(\text{mld})}(x, y, t) = 0$, the dia-surface mass transport takes the form

$$-\mathcal{S}^{(\text{subduction})} \equiv \rho dA_{(z)} \left(\frac{d(z + h^{(\text{mld})})}{dt} \right) \quad \text{at } z = -h^{(\text{mld})}(x, y, t), \quad (38.29)$$

where the mass transport $\mathcal{S}^{(\text{subduction})}$ (dimensions of mass per time) is positive for fluid moving downward beneath the mixed layer base into the pycnocline (subduction) and negative for water moving into the mixed layer (obduction). Expanding the material time derivative leads to

$$-\left(\frac{\mathcal{S}^{(\text{subduction})}}{\rho dA_{(z)}} \right) = w + (\partial_t + \mathbf{u} \cdot \nabla) h^{(\text{mld})} \quad \text{at } z = -h^{(\text{mld})}(x, y, t), \quad (38.30)$$

where again we define

$$\mathcal{S}^{(\text{subduction})} > 0 \quad \text{subduction} \quad (38.31)$$

$$\mathcal{S}^{(\text{subduction})} < 0 \quad \text{obduction}. \quad (38.32)$$

This definition of subduction rate corresponds to that given by [Cushman-Roisin \(1987\)](#).

- **ISOPYCNAL DEPTH :** At the depth of an isopycnal with $s(x, y, z, t) = z + h^{(\gamma)} = 0$, the dia-surface mass transport takes the form

$$-\mathcal{S}^{(\gamma)} \equiv \rho dA_{(z)} \left(\frac{d(z + h^{(\gamma)})}{dt} \right) \quad \text{at } z = -h^{(\gamma)}(x, y, \gamma, t), \quad (38.33)$$

where the mass transport $\mathcal{S}^{(\gamma)}$ (dimensions mass per time) is positive for fluid moving downward beneath the isopycnal depth $z = -h^{(\gamma)}$. Expanding the material time derivative leads to

$$-\left(\frac{\mathcal{S}^{(\gamma)}}{\rho dA_{(z)}} \right) = w + \frac{\partial h^{(\gamma)}}{\partial t} + \mathbf{u} \cdot \nabla h^{(\gamma)} \quad \text{at } z = -h^{(\gamma)}(x, y, \gamma, t). \quad (38.34)$$

This definition of mass transport corresponds to the kinematic expression for dia-surface mass transport detailed in Chapter 36, where we are interested in transport across a locally referenced potential density surface (see also equation (6.73) in [Griffies \(2004\)](#)).

38.2 MOM subduction diagnostic calculation

The subduction diagnostic in MOM is based on the definition (38.30). In this section, we outline the code algorithm and the available diagnostic fields.

38.2.1 Algorithm for subduction diagnostic

In brief, the algorithm consists of the following steps.

- Compute the mixed layer depth, $h^{(mld)}(x, y, t)$, based on some density increment criteria, with the increment set according to a namelist. The depth is naturally placed at the tracer point.
- Optionally smooth the diagnosed mixed layer depth. Note that smoothing is enabled via a namelist option; it incurs a trivial computational cost.
- Compute the time tendency $\partial_t h^{(mld)}$ based on the discrete time step difference between mixed layer depth

$$\frac{\partial h^{(mld)}}{\partial t} \approx \frac{h^{(mld)}(t + \Delta t) - h^{(mld)}(t)}{\Delta t}, \quad (38.35)$$

with Δt the model tracer time step.

- Compute the horizontal gradient of the mixed layer depth, $\nabla h^{(mld)}$. The zonal derivative is placed on the zonal face of a tracer cell, and the meridional derivative at the meridional position.
- Horizontally interpolate the zonal velocity component to the zonal tracer cell face, and the meridional velocity component to the meridional tracer cell face, both interpolations requiring two point averages on the B-grid (see Figure 9.5). The result is an estimate for the horizontal C-grid velocity components

$$u_{\text{east}}(i, j, k) = \frac{u(i, j, k) \text{dun}(i, j - 1) + u(i, j - 1, k) \text{dus}(i, j)}{\text{dus}(i, j) + \text{dun}(i, j - 1)} \quad (38.36)$$

$$v_{\text{north}}(i, j, k) = \frac{v(i - 1, j, k) \text{duw}(i, j) + v(i, j, k) \text{due}(i - 1, j)}{\text{duw}(i, j) + \text{due}(i - 1, j)}. \quad (38.37)$$

- Vertically interpolate the velocity ($u_{\text{east}}, v_{\text{north}}, w$) to the mixed layer base to determine ($u_{\text{east}}^{\text{mld}}, v_{\text{north}}^{\text{mld}}, w^{\text{mld}}$). There are various conditional cases to consider.

- If $h^{(mld)}(i, j) \leq \text{depth_zt}(i, j, k = 1)$, then no interpolation needed:

$$(u_{\text{east}}^{\text{mld}}, v_{\text{north}}^{\text{mld}}, w^{\text{mld}})(i, j) = (u_{\text{east}}, v_{\text{north}}, w)(i, j, k = 1) \quad \text{if } h^{(mld)}(i, j) \leq \text{depth_zt}(i, j, k = 1). \quad (38.38)$$

- If $h^{(mld)} = \text{depth_zt}(i, j, k)$ then no interpolation needed:

$$(u_{\text{east}}^{\text{mld}}, v_{\text{north}}^{\text{mld}}, w^{\text{mld}})(i, j) = (u_{\text{east}}, v_{\text{north}}, w)(i, j, k) \quad \text{if } h^{(mld)}(i, j) \leq \text{depth_zt}(i, j, k). \quad (38.39)$$

- If $\text{depth_zt}(i, j, k - 1) < h^{(mld)} < \text{depth_zt}(i, j, k)$ then vertical interpolation for each velocity component

$$\psi^{\text{mld}}(i, j) = \frac{\psi(i, j, k - 1) \text{dzup}(i, j, k) + \psi(i, j, k) \text{dztl}(i, j, k - 1)}{\text{dzup}(i, j, k) + \text{dztl}(i, j, k - 1)} \quad (38.40)$$

- If $h^{(mld)}(i, j) \geq \text{depth_zt}(i, j, k = \text{kmt}(i, j))$, then no interpolation needed:

$$(u_{\text{east}}^{\text{mld}}, v_{\text{north}}^{\text{mld}}, w^{\text{mld}})(i, j) = (u_{\text{east}}, v_{\text{north}}, w)(i, j, k = \text{kmt}) \quad \text{if } h^{(mld)}(i, j) \geq \text{depth_zt}(i, j, k = \text{kmt}). \quad (38.41)$$

- Estimate horizontal advection of the mixed layer base via

$$\mathbf{u} \cdot \nabla h^{(\text{mld})} \approx \mathbf{u}_{\text{east}}^{\text{mld}} \left(\frac{h^{(\text{mld})}(\mathbf{i} + 1, \mathbf{j}) - h^{(\text{mld})}(\mathbf{i}, \mathbf{j})}{dx_{te}(\mathbf{i}, \mathbf{j})} \right) + \mathbf{v}_{\text{north}}^{\text{mld}} \left(\frac{h^{(\text{mld})}(\mathbf{i}, \mathbf{j} + 1) - h^{(\text{mld})}(\mathbf{i}, \mathbf{j})}{dy_{tn}(\mathbf{i}, \mathbf{j})} \right). \quad (38.42)$$

- Compute the subduction mass flux rate via an estimate of equation (38.30)

$$\mathcal{S}^{(\text{subduction})}(\mathbf{i}, \mathbf{j}) \approx -\rho^{\text{mld}} dx_{te}(\mathbf{i}, \mathbf{j}) dy_{tn}(\mathbf{i}, \mathbf{j}) \left(\mathbf{w}^{\text{mld}} + \frac{\partial h^{(\text{mld})}}{\partial t} + \mathbf{u}^{(\text{mld})} \cdot \nabla h^{(\text{mld})} \right)(\mathbf{i}, \mathbf{j}), \quad (38.43)$$

where we use the estimate (38.35) for the time derivative, and equation (38.42) for the horizontal advection. The density $\rho^{\text{mld}} = \rho_o$ for a Boussinesq fluid, whereas it equals the vertically interpolated *in situ* density for a non-Boussinesq fluid.

- A final step is the optional binning of $\mathcal{S}^{(\text{subduction})}(\mathbf{i}, \mathbf{j})$ according to a chosen density classification, so to record the subduction as a function of density class in a manner similar to the binning of surface fluxes into density bins discussed in Section 36.14. This binning then allows for the diagnostic to be computed according to the recommendations of Kwon et al. (2013).

38.2.2 Available diagnostics

The following diagnostics are available related to the subduction calculation

$$\text{subduction} = \mathcal{S}^{(\text{subduction})} \quad (38.44)$$

$$\text{subduction_dhdt} = -\rho^{\text{mld}} dx_{te}(\mathbf{i}, \mathbf{j}) dy_{tn}(\mathbf{i}, \mathbf{j}) \left(\frac{\partial h^{(\text{mld})}}{\partial t} \right) \quad (38.45)$$

$$\text{subduction_horz} = -\rho^{\text{mld}} dx_{te}(\mathbf{i}, \mathbf{j}) dy_{tn}(\mathbf{i}, \mathbf{j}) \left(\mathbf{u}^{(\text{mld})} \cdot \nabla h^{(\text{mld})} \right) \quad (38.46)$$

$$\text{subduction_vert} = -\rho^{\text{mld}} dx_{te}(\mathbf{i}, \mathbf{j}) dy_{tn}(\mathbf{i}, \mathbf{j}) \mathbf{w}^{\text{mld}}. \quad (38.47)$$

Each of these fields has a corresponding diagnostic produced according to binning into neutral density classes

$$\text{subduction_nrho} = \text{rebin_onto_rho}(\text{subduction}) \quad (38.48)$$

$$\text{subduction_dhdt_nrho} = \text{rebin_onto_rho}(\text{subduction_dhdt}) \quad (38.49)$$

$$\text{subduction_horz_nrho} = \text{rebin_onto_rho}(\text{subduction_horz}) \quad (38.50)$$

$$\text{subduction_vert_nrho} = \text{rebin_onto_rho}(\text{subduction_vert}) \quad (38.51)$$

DIAGNOSING THE CONTRIBUTIONS TO SEA LEVEL EVOLUTION

Contents

39.1	Mass conservation for seawater and tracers	553
39.1.1	Continuum fluid	553
39.1.2	Kinematic boundary conditions	553
39.1.2.1	Bottom kinematic boundary condition	554
39.1.2.2	Surface kinematic boundary condition	554
39.1.3	Averaged mass and tracer equations	555
39.1.3.1	Form implied by mathematical correspondence	556
39.1.3.2	Residual mean velocity and quasi-Stokes velocity	556
39.1.3.3	Comments on particular averaging methods	558
39.1.3.4	Notation convention for remainder of chapter	558
39.1.4	Material changes of <i>in situ</i> density	558
39.1.5	General and simplified forms of the ocean equilibrium thermodynamics	559
39.2	Kinematic equations for sea level evolution	559
39.2.1	Kinematic sea level equation: Version I	560
39.2.2	Kinematic sea level equation: Version II	560
39.2.3	How we make use of the kinematic sea level equations	563
39.3	The non-Boussinesq steric effect	563
39.3.1	Defining the non-Boussinesq steric effect	563
39.3.2	Vertical motion across pressure surfaces	564
39.3.3	An expanded form of the kinematic sea level equation	565
39.3.3.1	Exposing the boundary fluxes of temperature and salinity	566
39.3.3.2	Expanded form of the kinematic sea level equation	567
39.3.3.3	Buoyancy fluxes and mass fluxes	567
39.3.3.4	Interior sources	568
39.3.3.5	Concerning the absence of advective temperature and salt fluxes	568
39.3.4	Boundary fluxes of heat and salt	569
39.3.4.1	Boundary heat fluxes	569
39.3.4.2	Boundary salt fluxes and sea ice	570
39.4	Evolution of global mean sea level	570
39.4.1	Preliminaries	571
39.4.2	Global mean sea level and the non-Boussinesq steric effect	571

39.4.3	Global mean sea level and the global steric effect	572
39.4.4	Relating global steric to non-Boussinesq steric	573
39.5	Vertical diffusion and global mean sea level	573
39.5.1	Contributions from vertical diffusion	574
39.6	Neutral diffusion and global mean sea level	574
39.6.1	Neutral directions and neutral tangent plane	574
39.6.2	Fluxes computed with locally orthogonal coordinates	575
39.6.3	Fluxes computed with projected neutral coordinates	575
39.6.4	Compensating neutral diffusive fluxes of temperature and salinity	576
39.6.5	The cabbeling and thermobaricity parameters	576
39.6.6	Physical aspects of cabbeling	576
39.6.7	Physical aspects of thermobaricity	577
39.7	Parameterized quasi-Stokes transport and global mean sea level	577
39.7.1	Formulation with buoyancy impacted by quasi-Stokes transport	577
39.7.2	Effects on global mean sea level	579
39.8	MOM sea level diagnostics: Version I	580
39.8.1	Surface buoyancy fluxes	580
39.8.2	Surface mass fluxes	581
39.8.3	Bottom heat flux	583
39.8.4	River insertion of liquid and solid water	583
39.8.5	River insertion of liquid water	583
39.8.6	River insertion of solid water	584
39.8.7	Heating of liquid ocean due to frazil formation	584
39.8.8	Motion across pressure surfaces	584
39.8.9	Mixing associated with vertical diffusion	585
39.8.10	Mixing associated with neutral diffusion	585
39.8.11	Parameterized eddy advection from GM	587
39.8.12	Parameterized eddy advection from submesoscale parameterization	588
39.8.13	Parameterized horizontal diffusion from submesoscale parameterization	589
39.8.14	Penetrative shortwave radiation	589
39.8.15	Sigma transport	590
39.8.16	Mixdownslope	590
39.8.17	KPP nonlocal mixing	590
39.8.18	Cross land mixing	591
39.8.19	Cross land insertion	591
39.8.20	Smoothing of free surface or bottom pressure	591
39.9	MOM sea level diagnostics: Version II	592

The purpose of this chapter is to discuss the suite of diagnostics in MOM to quantify how various terms contribute to sea level evolution. The discussion here is an abbreviated version of that given by [Griffies and Greatbatch \(2012\)](#). Some of this material repeats similar discussions in Chapter 2. However, we find it useful to include such duplication here to enable the present chapter to be self-contained.

The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_barotropic.F90
ocean_core/ocean_sbc.F90
ocean_core/ocean_bbc.F90
```

as well as many other modules associated with parameterized subgrid scale physical processes.

39.1 Mass conservation for seawater and tracers

The purpose of this section is to formulate the equations for conservation of seawater mass and tracer mass. The seawater mass conservation equation is the basis for the kinematic equations derived in Section 39.2 for sea level. The tracer mass conservation equation describes the evolution of scalar trace constituents in seawater such as salt and biogeochemical tracers. Additionally, the evolution equation for conservative temperature satisfies the mathematically identical advection-diffusion equation (McDougall, 2003).

39.1.1 Continuum fluid

Eulerian expressions for the conservation of seawater mass and tracer mass are given by

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{v}) \quad (39.1)$$

$$\frac{\partial (\rho C)}{\partial t} = -\nabla \cdot (\rho \mathbf{v} C + \mathbf{J}_{\text{mdiff}}), \quad (39.2)$$

which have equivalent Lagrangian expressions

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{v} \quad (39.3)$$

$$\rho \frac{dC}{dt} = -\nabla \cdot \mathbf{J}_{\text{mdiff}}. \quad (39.4)$$

In these equations,

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla \quad (39.5)$$

is the material time derivative computed by an observer moving with the fluid parcel's center of mass velocity \mathbf{v} , whereas ∂_t is the Eulerian time derivative computed at a fixed spatial point. The tracer concentration, C , is the mass of a trace constituent within a fluid parcel, per mass of seawater within the parcel. Consequently, the product ρC is the mass of tracer per volume of seawater, with ρ the *in situ* seawater mass density. The flux $\mathbf{J}_{\text{mdiff}}$ arises from local gradients in the tracer field being acted upon by a nonzero molecular diffusivity. For passive tracers (those tracers not impacting density), this molecular diffusion flux vanishes when the tracer concentration is uniform, in which case the expression for tracer mass conservation (39.2) reduces to the seawater mass conservation (39.1). This *compatibility condition* is fundamental to the continuum mass and tracer equations, and it follows since we choose to measure the motion of fluid parcels using the parcel's center of mass velocity \mathbf{v} (e.g., see Section II.2 of DeGroot and Mazur (1984), Section 8.4 of Chaikin and Lubensky (1995), or Section 3.3 of Müller (2006)). For the active tracers (those tracers impacting density) temperature and salinity, there is generally a nonzero molecular flux of each field arising from gradients in the other field, with this process known the Soret and Dufour effect (Landau and Lifshitz, 1987). As this *cross-diffusion* effect is swamped by larger eddy fluxes considered in the following, we do not consider it further.

39.1.2 Kinematic boundary conditions

Kinematic boundary conditions arise from the geometric constraints imposed by the ocean bottom and surface, with such constraints impacting the budgets of mass over a column of seawater. In formulating the kinematic boundary conditions, we assume the ocean bottom to be static and impermeable (i.e., closed to mass fluxes), whereas the ocean surface is dynamic and open to mass fluxes. We provide some detail regarding the derivation of these boundary conditions since they expose issues essential for understanding the evolution of global mean sea level.

39.1.2.1 Bottom kinematic boundary condition

At the rigid ocean bottom, the kinematic boundary condition states that the geometric expression for the ocean bottom

$$z + H(x, y) = 0 \quad \text{ocean bottom,} \quad (39.6)$$

remains fixed in time for all parcels situated at the bottom, so that

$$\frac{d(z + H)}{dt} = 0 \quad \text{at } z = -H(x, y), \quad (39.7)$$

where the coordinates x, y denote the lateral position of a fluid parcel in the ocean. An equivalent statement is that there is no normal flow of fluid at the ocean bottom, so that $\mathbf{v} \cdot \hat{\mathbf{n}} = 0$, with the outward normal given by

$$\hat{\mathbf{n}} = -\left(\frac{\nabla(z + H)}{|\nabla(z + H)|} \right) \quad \text{at } z = -H(x, y). \quad (39.8)$$

In either case, we may write the bottom kinematic boundary condition as

$$w + \mathbf{u} \cdot \nabla H = 0 \quad \text{at } z = -H(x, y), \quad (39.9)$$

where

$$\mathbf{v} = (\mathbf{u}, w) \quad (39.10)$$

is the fluid velocity field, with \mathbf{u} the horizontal velocity.

39.1.2.2 Surface kinematic boundary condition

We assume that the ocean surface can be written geometrically as

$$z - \eta(x, y, t) = 0 \quad \text{ocean surface,} \quad (39.11)$$

which means that there are no overturns; i.e., we filter out breaking surface waves. The mass per time of material crossing the surface is written as

$$\text{mass per time through surface} = \mathcal{Q}_m dA_\eta, \quad (39.12)$$

where dA_η is the infinitesimal area element on the ocean surface, and \mathcal{Q}_m is the mass per time per surface area of material crossing the surface. This mass flux can be equivalently written as the normal projection of the relative velocity at the ocean surface, multiplied by the surface density

$$\mathcal{Q}_m dA_\eta = \rho (\mathbf{v} - \mathbf{v}_\eta) \cdot \hat{\mathbf{n}} dA_\eta \quad \text{at } z = \eta, \quad (39.13)$$

where

$$\hat{\mathbf{n}} = \frac{\nabla(z - \eta)}{|\nabla(z - \eta)|} \quad \text{at } z = \eta \quad (39.14)$$

is the outward normal at the ocean surface, and

$$\mathbf{v}_\eta \cdot \hat{\mathbf{n}} = \frac{\partial_t \eta}{|\nabla(z - \eta)|} \quad \text{at } z = \eta \quad (39.15)$$

is the normal velocity of a material point fixed to the ocean surface. The surface kinematic boundary condition given by equation (39.13) says there is a nonzero projection of the relative velocity onto the surface outward normal, $(\mathbf{v} - \mathbf{v}_\eta) \cdot \hat{\mathbf{n}}$, only when there is a nonzero mass flux through the undulating surface. This equation is written in a manner to expose the geometric nature of the boundary condition. The following develops a related form that is somewhat more familiar.

Given that the surface has no overturns, we can equivalently write the mass flux in the more convenient form

$$\text{mass per time through surface} = \mathcal{Q}_m dA, \quad (39.16)$$

where dA is the horizontal area element obtained by projecting the surface area element dA_η onto the horizontal plane, and Q_m is the mass per time per horizontal area of material crossing the surface. The two area elements are related by the expression (see Section 20.13.2 of [Griffies \(2004\)](#))

$$dA_\eta = |\nabla(z - \eta)| dA. \quad (39.17)$$

We now return to the boundary condition (39.13), yet replace the mass flux Q_m with Q_m , use the area relation (39.17), and write the normal material velocity at the ocean surface in the form (39.15), with these steps yielding the kinematic boundary condition (e.g., Section 3.4 of [Griffies \(2004\)](#))

$$\rho(\partial_t + \mathbf{u} \cdot \nabla)\eta = Q_m + \rho w \quad \text{at } z = \eta \quad (39.18)$$

or the equivalent Lagrangian expression

$$\rho \left(\frac{d(z - \eta)}{dt} \right) = -Q_m \quad \text{at } z = \eta. \quad (39.19)$$

As a self-consistency check, note that the mass per horizontal area in a fluid column,

$$\text{mass per horizontal area} = \int_{-H}^{\eta} \rho dz, \quad (39.20)$$

is altered by the convergence of mass to the column via ocean currents, and mass entering through the ocean surface, so that

$$\partial_t \left(\int_{-H}^{\eta} \rho dz \right) + \nabla \cdot \left(\int_{-H}^{\eta} \rho \mathbf{u} dz \right) = Q_m. \quad (39.21)$$

Leibniz's Rule can be used to move the time and space derivatives across the integral sign, with the bottom kinematic boundary condition (39.9) and Eulerian form of the mass continuity equation (39.1) recovering the surface kinematic boundary condition (39.18).

Matter entering the ocean is predominantly in the form of fresh water plus trace constituents, such as salt and biogeochemical matter,

$$Q_m = Q_w + Q^{(S)} \quad (39.22)$$

where Q_w is the mass flux of fresh water, and $Q^{(S)}$ is the mass flux of salt or other trace constituents. A nonzero salt flux for climate purposes is generally limited to regions under sea ice. In general, the salt and trace constituent surface mass flux is far smaller than the mass flux from fresh water. Hence, the matter flux is often approximated just with the fresh water flux. In this way, the exchange of tracer mass across the ocean surface generally does not impact the ocean mass in climate models. More realistic river models, and sea ice models that carry the mass of tracers, will necessitate removing this assumption from ocean models.

The surface kinematic boundary condition can be rearranged into a prognostic equation for sea level

$$\partial_t \eta = Q_m / \rho + (\hat{\mathbf{n}} \cdot \mathbf{v}) |\nabla(z - \eta)| \quad \text{at } z = \eta. \quad (39.23)$$

Hence, mass entering through the free surface ($Q_m > 0$) contributes to a positive sea level tendency, as does a three dimensional velocity that has a nonzero projection “upwards” ($\hat{\mathbf{n}} \cdot \mathbf{v} > 0$). The expression (39.23), though useful geometrically, does not provide the necessary means for partitioning sea level evolution into physical processes. For this purpose, we pursue the development of alternative sea level equations in Section 39.2.

39.1.3 Averaged mass and tracer equations

The seawater mass conservation equation (39.1), and the tracer mass conservation equation (39.2), arise from a continuum formulation of fluid mechanics (e.g., [Batchelor \(1967\)](#) and [Landau and Lifshitz \(1987\)](#)).

The finite sized grid (or, less commonly, a finite spectral representation) used in a numerical ocean model introduces a cutoff scale absent from the continuum. Formulating the equations discretized by an ocean model requires an averaging operation in which the continuum equations are averaged over scales smaller than the grid. A discrete ocean model then provides an approximation to the averaged equations by using various methods from computational physics. When averaging nonlinear products, such as $\rho \mathbf{v} C$, correlations appear between space and time fluctuations that are unresolved by the grid. Ideally, these correlations can be organized into the form of a subgrid scale flux divergence. Parameterizing the subgrid scale flux divergence is nontrivial, with no universal approach available.

39.1.3.1 Form implied by mathematical correspondence

We only require general properties of subgrid scale fluxes, rather than formulations specific to a particular physical process. A very convenient property we desire is that the averaged mass and tracer equations are written in the same mathematical form as the corresponding continuum equations. More precisely, the terms appearing in the averaged equations correspond to, though are generally distinct from, their continuum analogs. In turn, we insist that the compatibility condition between the continuum budgets for seawater mass and tracer mass be maintained for the averaged equations. Maintaining this direct correspondence between continuum and averaged equations facilitates straightforward physical interpretations of the averaged equations, which follow from the interpretation of the continuum equations.

Maintenance of the same mathematical form as the continuum mass conservation equation (39.1) and tracer conservation equation (39.2) allows us to write the averaged mass and tracer conservation equations in the form

$$\frac{\partial \rho_a}{\partial t} = -\nabla \cdot (\rho_a \mathbf{v}^\dagger) \quad (39.24)$$

$$\frac{\partial (\rho_a C_a)}{\partial t} = -\nabla \cdot (\rho_a \mathbf{v}^\dagger C_a + \mathbf{J}_{\text{eddydiff}}), \quad (39.25)$$

or the equivalent Lagrangian expressions

$$\frac{d^\dagger \rho_a}{dt} = -\rho_a \nabla \cdot \mathbf{v}^\dagger \quad (39.26)$$

$$\rho_a \frac{d^\dagger C_a}{dt} = -\nabla \cdot \mathbf{J}_{\text{eddydiff}}, \quad (39.27)$$

where the “a” subscript signifies an averaged quantity. In ocean circulation models, the subgrid scale eddy tracer diffusive flux, $\mathbf{J}_{\text{eddydiff}}$, is generally far larger than the corresponding molecular flux, $\mathbf{J}_{\text{mdiffuse}}$. Nonetheless, as for the molecular flux, compatibility between the averaged mass and tracer equations is maintained so long as the eddy flux vanishes when the tracer concentration, C_a , is spatially uniform. Compatibility is maintained by the commonly used flux-gradient relation for the parameterized subgrid scale tracer diffusive flux.

39.1.3.2 Residual mean velocity and quasi-Stokes velocity

Along with an eddy diffusive flux, $\mathbf{J}_{\text{eddydiff}}$, we introduced to the averaged equations the *residual mean* velocity, \mathbf{v}^\dagger , which advects seawater mass and tracer mass, thus bringing the material time derivative to the form

$$\frac{d^\dagger}{dt} = \frac{\partial}{\partial t} + \mathbf{v}^\dagger \cdot \nabla. \quad (39.28)$$

The residual mean velocity is generally partitioned into two pieces

$$\mathbf{v}^\dagger = \mathbf{v}_a + \mathbf{v}^*, \quad (39.29)$$

where \mathbf{v}_a is the averaged velocity directly represented by the numerical model, and \mathbf{v}^* is an eddy induced or quasi-Stokes velocity that requires a parameterization before being represented by the model. Proposed parameterizations of \mathbf{v}^* used by ocean circulation models (e.g., [Gent and McWilliams \(1990\)](#), [Gent et al.](#)

(1995), McDougall and McIntosh (2001), Fox-Kemper et al. (2008b)) all satisfy the non-divergence property in the ocean interior

$$\nabla \cdot (\rho_a \mathbf{v}^*) = 0 \quad \text{ocean interior} \quad (39.30)$$

and the no-normal flow condition at ocean boundaries

$$\hat{\mathbf{n}} \cdot \mathbf{v}^* = 0 \quad \text{ocean boundaries.} \quad (39.31)$$

The non-divergence condition (39.30) means that the quasi-Stokes mass transport can be written as the curl of a vector streamfunction

$$\rho_a \mathbf{v}^* = \nabla \wedge \rho_a \Psi, \quad (39.32)$$

so that the tracer equation (39.27) can be written as

$$\rho_a \frac{dC_a}{dt} = -\nabla \cdot (\mathbf{J}_{\text{eddydiff}} + \mathbf{J}_{\text{skew}}). \quad (39.33)$$

In this equation,

$$\mathbf{J}_{\text{skew}} = -\nabla C_a \wedge \rho_a \Psi \quad (39.34)$$

is a skew tracer flux, which differs from the advective tracer flux through a non-divergent curl

$$\rho_a \mathbf{v}^* C_a = -\nabla C_a \wedge \rho_a \Psi + \nabla \wedge (\rho_a \Psi C_a), \quad (39.35)$$

thus making the divergence of the advective flux equal to the divergence of the skew flux

$$\nabla \cdot (\rho_a \mathbf{v}^* C_a) = -\nabla \cdot (\nabla C_a \wedge \rho_a \Psi). \quad (39.36)$$

The following presents three more implications of the non-divergence and no-normal flow properties (39.30) and (39.31) that will be utilized in this paper.

- The averaged seawater mass continuity equations hold whether one uses the residual mean velocity \mathbf{v}^\dagger , as in equations (39.24) and (39.26), or the mean velocity \mathbf{v}_a

$$\frac{\partial \rho_a}{\partial t} = -\nabla \cdot (\rho_a \mathbf{v}_a) \quad (39.37)$$

$$\frac{d\rho_a}{dt} = -\rho_a \nabla \cdot \mathbf{v}_a. \quad (39.38)$$

- The kinematic boundary conditions from Section 39.1.2 remain identical whether using \mathbf{v}^\dagger or \mathbf{v}_a .
- Vertical integration of the non-divergence condition (39.30) over a seawater column, and use of the no-flux boundary condition (39.31), yields

$$\nabla \cdot \left(\int_{-H}^{\eta} dz \rho_a \mathbf{u}^* \right) = 0, \quad (39.39)$$

where \mathbf{u}^* is the horizontal component to the eddy induced velocity. The Gent et al. (1995) mesoscale eddy parameterization and the Fox-Kemper et al. (2008b) submesoscale eddy parameterization achieve this property by satisfying $\int_{-H}^{\eta} dz \rho_a \mathbf{u}^* = 0$. The eddy induced mass transport for these parameterizations thus has zero vertically integrated component.

These three properties mean that it is a matter of convenience what form of the mass conservation equation we choose when formulating the kinematic sea level equations in Section 39.2. We prefer the starting point offered by equations (39.37) and (39.38), in which the quasi-Stokes velocity is absent. This choice then places impacts of the quasi-Stokes transport directly onto buoyancy, as detailed in Section 39.7, rather than on the material evolution of pressure (Section 39.3.2).

39.1.3.3 Comments on particular averaging methods

Although we are not concerned with details of the averaging required to reach equations (39.24) and (39.25), it is important to note that methods exist to write the averaged scalar equations in precisely these forms. Examples include the density weighted averaging of Hesselberg (1926) (see also McDougall et al. (2002) and Chapter 8 of Griffies (2004)); the isopycnal thickness weighted methods of DeSzoek and Bennett (1993) and McDougall and McIntosh (2001) (see also Chapter 9 of Griffies (2004)); and the combined density and thickness weighted methods of Greatbatch and McDougall (2003). In contrast, the approach of Eden et al. (2007) leads to different averaged mass and tracer equations than (39.24) and (39.25), since they introduce a different eddy induced velocity for each tracer. However, so long as the eddy induced velocity for each tracer satisfies the non-divergence condition (39.30) and the no-normal flow boundary condition (39.31), their approach should maintain the compatibility condition between mass and tracer equations, and thus it falls within the framework of the present considerations.

39.1.3.4 Notation convention for remainder of chapter

Now that we have introduced a convention for the model fields, which are the result of a particular averaging procedure, we dispense with the “a” subscript in order to reduce notational clutter. In subsequent discussions, all equations and fields will refer to their averaged forms.

39.1.4 Material changes of *in situ* density

The *in situ* density, ρ , is a function of three intensive fluid properties

$$\rho = \rho(\Theta, S, p), \quad (39.40)$$

with Θ the conservative temperature (McDougall, 2003; IOC et al., 2010), S the salinity, and p the pressure. We prefer the conservative temperature as it reflects more accurately on the conservative nature of enthalpy transport in the ocean as compared to the alternative potential temperature.

The equation of state (39.40) leads to the material time evolution of *in situ* density

$$\begin{aligned} \frac{d\rho}{dt} &= \frac{\partial \rho}{\partial \Theta} \frac{d\Theta}{dt} + \frac{\partial \rho}{\partial S} \frac{dS}{dt} + \frac{\partial \rho}{\partial p} \frac{dp}{dt} \\ &= -(\rho\alpha) \frac{d\Theta}{dt} + (\rho\beta) \frac{dS}{dt} + \frac{1}{c_{\text{sound}}^2} \frac{dp}{dt}. \end{aligned} \quad (39.41)$$

These equations introduced the thermal expansion coefficient,

$$\alpha = -\frac{1}{\rho} \left(\frac{\partial \rho}{\partial \Theta} \right)_{p,S} \quad (39.42)$$

the haline contraction coefficient

$$\beta = \frac{1}{\rho} \left(\frac{\partial \rho}{\partial S} \right)_{p,\Theta} \quad (39.43)$$

and the squared speed of sound

$$c_{\text{sound}}^2 = \left(\frac{\partial p}{\partial \rho} \right)_{S,\Theta}. \quad (39.44)$$

The material evolution of density is thus partitioned into the evolution of buoyancy (via changes in temperature and salinity) and the evolution of pressure. Buoyancy remains unchanged by processes that are both adiabatic and isohaline, as well as processes where diabatic and non-isohaline effects perfectly cancel. Pressure evolution arises from vertical motion across pressure surfaces which, in a hydrostatic fluid, is equivalent to the vertical motion of mass.

Mass conservation in the form of equation (39.3), along with material evolution of density (39.41), render the balance

$$-\rho \nabla \cdot \mathbf{v} = -(\rho\alpha) \frac{d\Theta}{dt} + (\rho\beta) \frac{dS}{dt} + \frac{1}{c_{\text{sound}}^2} \frac{dp}{dt}. \quad (39.45)$$

To help further develop our understanding of this result, consider the mass of a fluid parcel of volume δV written in the form $M = \rho \delta V$. Mass conservation for this parcel means that as the parcel volume increases, the density must decrease, so that

$$\frac{1}{\rho} \frac{d\rho}{dt} = -\frac{1}{\delta V} \frac{d\delta V}{dt}. \quad (39.46)$$

Correspondingly, from the mass continuity equation (39.3), the volume of a fluid parcel increases when moving through regions of fluid with a divergent velocity field

$$\frac{1}{\delta V} \frac{d\delta V}{dt} = \nabla \cdot \mathbf{v}. \quad (39.47)$$

Substitution into equation (39.45) then yields

$$-\frac{\rho}{\delta V} \frac{d\delta V}{dt} = -(\rho\alpha) \frac{d\Theta}{dt} + (\rho\beta) \frac{dS}{dt} + \frac{1}{c_{\text{sound}}^2} \frac{dp}{dt}. \quad (39.48)$$

This balance states that the volume of a fluid parcel increases (negative left hand side) as the buoyancy of the parcel increases, such as occurs with heating in regions of $\alpha > 0$, or freshening in regions of $\beta > 0$. Volume also increases as the pressure of the parcel decreases ($dp/dt < 0$). We have many opportunities to return to this balance, and its relatives, in the subsequent development of sea level equations, which represent the accumulation of volume changes throughout a seawater column.

39.1.5 General and simplified forms of the ocean equilibrium thermodynamics

The thermal expansion coefficient α , haline contraction coefficient β , and squared speed of sound c_{sound}^2 are properties of the equilibrium thermodynamics of seawater, with [IOC et al. \(2010\)](#) summarizing the state of the science. In the ocean, these fields are nonlinear functions of the conservative temperature, salinity, and pressure, and this dependence leads to the processes of cabbeling and thermobaricity, in which parcels move dianeutrally ([McDougall, 1987b](#)). Elements of these processes are described in Section 39.6, where we illustrate how they impact on global mean sea level through the non-Boussinesq steric effect. Additionally, as shown in Section 39.3, the values for α , β , and c_{sound}^2 introduce scales that further influence how physical processes and boundary fluxes impact global mean sea level through the non-Boussinesq steric effect.

Under certain idealized situations, we may approximate α , β , and ρc_{sound}^2 to be constants independent of the ocean state. This approximation simplifies the equilibrium thermodynamics to help characterize where the more general ocean thermodynamics is fundamental. In particular, as shown in Section 39.6, the simplified thermodynamics associated with constant α , β , and ρc_{sound}^2 eliminates the processes of cabbeling and thermobaricity. Furthermore, with constant α , β , and ρc_{sound}^2 , mass conservation in the form of equation (39.48) indicates that the material evolution for the volume of a fluid parcel is a linear function of material changes in conservative temperature, salinity, and pressure. In the special case where density is a function just of conservative temperature, a constant $\alpha = -\rho^{-1} \partial \rho / \partial \Theta$ leads to $\rho(\Theta) = \rho_o e^{-\alpha(\Theta - \Theta_o)}$, where $\rho_o = \rho(\Theta_o)$. For conservative temperatures near to the reference value Θ_o , density is given by the linear equation of state

$$\rho(\Theta) \approx \rho_o [1 - \alpha(\Theta - \Theta_o)]. \quad (39.49)$$

In turn, a constant α with a linear equation of state corresponds to $\alpha = -\rho_o^{-1} \partial \rho / \partial \Theta$.

39.2 Kinematic equations for sea level evolution

We formulate here the kinematic evolution equations for sea level, again under the assumptions of (1) fixed geoid and ocean bottom, (2) constant horizontal area of the ocean, (3) space-time independent gravitational acceleration. All mathematical symbols refer to the averaged quantities appropriate for an ocean model.

39.2.1 Kinematic sea level equation: Version I

Integrating the material or Lagrangian form of the mass continuity equation (39.38) over the depth of the ocean, using the kinematic boundary conditions (Section 39.1.2), and Leibniz's rule for moving the derivative operator across an integral, yields Version I of the *kinematic sea level equation*

$$\frac{\partial \eta}{\partial t} = \frac{Q_m}{\rho(\eta)} - \nabla \cdot \mathbf{U} - \int_{-H}^{\eta} dz \frac{1}{\rho} \frac{d\rho}{dt}, \quad (39.50)$$

with

$$\mathbf{U} = \int_{-H}^{\eta} \mathbf{u} dz \quad (39.51)$$

the vertically integrated horizontal velocity, and

$$\rho(\eta) = \rho(x, y, z = \eta(x, y, t), t) \quad (39.52)$$

the ocean density at the free surface. Equation (39.50) partitions the evolution of sea level into the following three physical processes, with Figure 39.1 providing a schematic.

1. **BOUNDARY MASS FLUXES:** The transport of mass across the ocean surface is converted to a volume flux through multiplication by the specific volume $\rho(\eta)^{-1}$ at the ocean surface. Transport of mass across the ocean surface in regions of large surface specific volume $\rho(\eta)^{-1}$ (e.g., warm and freshwater) leads to greater sea level tendencies than the transport in regions of small surface specific volume (e.g., cold and salty regions).
2. **DYNAMIC:** The convergence of vertically integrated horizontal currents onto a fluid column redistributes ocean volume, and as such it imparts a sea level tendency. To help understand how flow convergence arises, decompose the vertically integrated velocity as $\mathbf{U} = d \bar{\mathbf{u}}^z$, where $d = H + \eta$ is the thickness of the fluid column, and $\bar{\mathbf{u}}^z$ is the vertically averaged horizontal velocity. Converging flow in a flat bottom ocean is equivalent to convergence of the vertically averaged flow, so that $-\nabla \cdot \mathbf{U} = -d \nabla \cdot \bar{\mathbf{u}}^z > 0$. Flow convergence occurs also when the vertically averaged flow slows down when approaching a region in the fluid, or flow convergence occurs when vertically constant flow impinges on a shoaling bottom, so that $-\nabla \cdot \mathbf{U} = -\nabla(d) \cdot \bar{\mathbf{u}}^z > 0$.

Globally integrating the sea level equation (39.50) eliminates the effects of ocean volume transport. Hence, in this formulation of the sea level equation, ocean currents redistribute ocean volume, yet do not directly impact global mean sea level. We find this property of the sea level equation (39.50) especially convenient for studying the evolution of global mean sea level.

3. **NON-BOUSSINESQ STERIC:** Material time changes of the *in situ* density integrated over a column of seawater lead to the expansion or contraction of the fluid column. Material changes in density arise from buoyancy fluxes at the ocean boundaries, convergence of buoyancy fluxes in the ocean interior, and processes associated with the equilibrium thermodynamics of the ocean as embodied by the equation of state (equation (39.40)).

39.2.2 Kinematic sea level equation: Version II

The second version of the kinematic sea level equation is formulated from the mass budget for a fluid column, rearranged to the form

$$\partial_t \left(\int_{-H}^{\eta} \rho dz \right) = Q_m - \nabla \cdot \left(\int_{-H}^{\eta} \rho \mathbf{u} dz \right). \quad (39.53)$$

The left hand side of this equation expresses the time tendency for the mass per unit horizontal area in the column, with this tendency affected by mass transported across the ocean surface, and mass transported

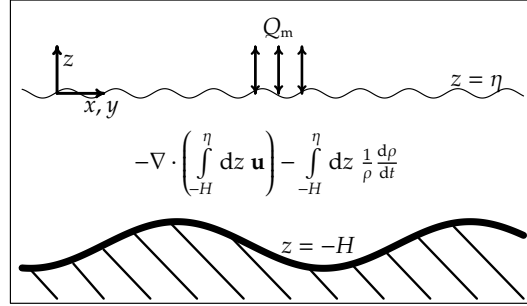


Figure 39.1: Shown here is a schematic ocean basin illustrating the ocean fluid dynamic and boundary processes that impact sea level according to the kinematic sea level equation (39.50). The surface mass flux Q_m arises from exchange of mass across the permeable sea surface $z = \eta(x, y, t)$ via precipitation, evaporation, runoff, and melt. This mass flux is converted to a volume flux through dividing by the surface ocean density $\rho(z = \eta)$. The non-Boussinesq steric effect, $-\int_{-H}^{\eta} \rho^{-1} d\rho/dt$, arises from material changes in the *in situ* density, as integrated from the impermeable bottom at $z = -H(x, y)$ to the ocean surface. Both the surface mass flux and non-Boussinesq steric effect can change the net global mean sea level. The third process impacting sea level arises from the convergence of depth integrated currents, $-\nabla \cdot \mathbf{U}$, which redistributes volume without impacting global mean sea level.

into the column by ocean currents. This mass budget can be written in a modified form by introducing the column averaged density

$$\bar{\rho}^z = \frac{1}{H + \eta} \left(\int_{-H}^{\eta} \rho dz \right), \quad (39.54)$$

and the column integrated horizontal mass transport

$$\mathbf{U}^\rho = \int_{-H}^{\eta} \rho \mathbf{u} dz, \quad (39.55)$$

in which case the mass budget becomes

$$\partial_t [(H + \eta) \bar{\rho}^z] = Q_m - \nabla \cdot \mathbf{U}^\rho. \quad (39.56)$$

Expanding the time derivative renders Version II of the sea level equation

$$\frac{\partial \eta}{\partial t} = \frac{Q_m - \nabla \cdot \mathbf{U}^\rho - (H + \eta) \partial \bar{\rho}^z / \partial t}{\bar{\rho}^z}. \quad (39.57)$$

Equation (39.57) is mathematically equivalent to Version I of the sea level equation (39.50). However, there is a difference in emphasis that provides some complement to Version I. We thus find it useful to again discuss the three physical processes that impact sea level as revealed by equation (39.57).

1. **BOUNDARY FLUXES:** As for Version I, the boundary contribution arises from mass fluxes crossing the ocean boundaries. However, for Version II, the conversion of mass to volume flux occurs through the column mean density, $\bar{\rho}^z$, rather than the surface density $\rho(\eta)$.
2. **DYNAMIC:** Version I considers the dynamic effects from the transport of volume by the vertically integrated currents. Version II considers the transport of mass by the vertically integrated currents, which is then converted to a sea level tendency through dividing by the vertically averaged density. Since the mass flux must be converted to a volume flux through division by the spatially dependent depth mean density, $\bar{\rho}^z$, a global area integral of the sea level equation (39.57) does not identically remove the effects on sea level from mass transport by ocean currents. Instead, the effects are only approximately removed, to the extent that the vertical mean density is horizontally constant.

3. **LOCAL STERIC:** Version II of the sea level equation (39.57) exposes an Eulerian time derivative acting on the vertically averaged density, rather than the vertical integral of the material time derivative present in Version I. In so doing, we see how sea level is directly affected through the local changes to the column mean density. As the column mean density decreases, the column expands and this in turn contributes a positive tendency to sea level evolution. We refer to this steric contribution as the *local steric effect* to contrast it with the non-Boussinesq steric term introduced in Section 39.2.1.

Although we focus in the remainder of this paper on Version I of the kinematic sea level equation (39.50), it is useful to here make a connection between the kinematic sea level equation (39.57) and the analogous sea level equation proposed by Gill and Niiler (1973) for a hydrostatic fluid. First, note that the sea level evolution in equation (39.57) can be partitioned into a contribution from mass changes and a contribution from local steric changes

$$\frac{\partial \eta}{\partial t} = \underbrace{\frac{Q_m - \nabla \cdot \mathbf{U}^\rho}{\bar{\rho}^z}}_{\text{mass}} - \underbrace{\frac{(H + \eta) \partial \bar{\rho}^z / \partial t}{\bar{\rho}^z}}_{\text{local steric}}. \quad (39.58)$$

It is just this decomposition that was promoted by Gill and Niiler (1973) for a hydrostatic fluid. Notably, equation (39.58) follows solely from the kinematics of a mass conserving fluid and so holds for both hydrostatic and non-hydrostatic fluids.

To further make the connection to Gill and Niiler (1973), introduce the hydrostatic approximation, in which the difference between hydrostatic pressure at the ocean bottom, p_b , and pressure applied to the ocean surface, p_a , is given by

$$p_b - p_a = g \int_{-H}^{\eta} \rho \, dz, \quad (39.59)$$

where g is the gravitational acceleration. Use of the column integrated mass balance (39.53) thus leads to the kinematic evolution equation

$$g^{-1} \partial_t (p_b - p_a) = Q_m - \nabla \cdot \mathbf{U}^\rho. \quad (39.60)$$

This equation says that mass per area within a column of seawater changes according to the surface mass flux and the convergence of the vertically integrated mass transport. Substituting this result into the sea level equation (39.57) leads to

$$\frac{\partial \eta}{\partial t} = \underbrace{\frac{\partial_t (p_b - p_a)}{g \bar{\rho}^z}}_{\text{mass}} - \underbrace{\frac{(H + \eta) \partial \bar{\rho}^z / \partial t}{\bar{\rho}^z}}_{\text{local steric}}. \quad (39.61)$$

As for the expression (39.58), we see that sea level experiences a positive tendency in those regions where mass locally increases and where the vertically averaged density decreases. As noted by Gill and Niiler (1973), and further supported by model analyses from Landerer et al. (2007), Yin et al. (2009), and Yin et al. (2010a), there are many useful physical insights concerning regional sea level garnered when performing this kinematic partitioning of sea level evolution into mass and local steric changes, with local steric effects further partitioned into thermal (thermosteric) and haline (halosteric) effects. The particular form of this partition as proposed by Gill and Niiler (1973) arises from taking the time derivative of the hydrostatic balance (39.59), which leads after rearrangement to

$$\frac{\partial \eta}{\partial t} = \underbrace{\frac{\partial_t (p_b - p_a)}{g \rho(\eta)}}_{\text{mass}} - \underbrace{\frac{\int_{-H}^{\eta} (\partial \rho / \partial t) \, dz}{\rho(\eta)}}_{\text{steric}}. \quad (39.62)$$

The two expressions (39.61) and (39.62) have the same physical content, though the terms on the right hand side slightly differ quantitatively.

39.2.3 How we make use of the kinematic sea level equations

The kinematic sea level equations (39.50) and (39.57) arise just from mass balance and the kinematic boundary conditions, which lends great generality to their applicability; e.g., they are valid for both hydrostatic and non-hydrostatic fluids. At each point in the ocean, sea level tendencies must respect these fundamental kinematic results. Hence, equations (39.50) and (39.57) offer a means for partitioning the evolution of sea level into various physical processes. However, there are questions that kinematics cannot answer absent dynamical principles (e.g., [Lowe and Gregory, 2006](#)). That is, the kinematic approach provides a useful diagnostic framework, but it does not, alone, produce a complete predictive framework.

For example, a kinematic approach alone cannot uncover how sea level adjusts in the presence of forcing, or the time scales over which low frequency patterns emerge. For example, should we expect low frequency (order monthly or longer) sea level patterns to exhibit high values under regions of net precipitation and low values under regions of net evaporation? At any instant of time, net precipitation will contribute a positive tendency to the sea level, and net evaporation will contribute a negative tendency, as revealed by the kinematic sea level equations (39.50) and (39.57). However, the low frequency sea level patterns are not generally correlated with patterns of evaporation minus precipitation.

The reason for the decorrelation is that there is a dynamical adjustment occurring on a barotropic time scale (a few days) that causes perturbations from boundary mass fluxes, as well as from the non-Boussinesq steric effect, to be transmitted globally through gravity wave adjustments ([Greatbatch, 1994](#)). Consequently, these perturbations rapidly impact global mean sea level, but generally leave little imprint on regional low frequency sea level patterns. That is, the perturbations can be considered barotropic wavemakers. Additional impacts from the input of freshwater and buoyancy to the ocean alter regional sea level patterns through baroclinic adjustments captured by both volume conserving Boussinesq and mass conserving non-Boussinesq ocean models (see, e.g., [Hsieh and Bryan \(1996\)](#), [Bryan \(1996\)](#), [Landerer et al. \(2007\)](#), [Stammer \(2008\)](#), [Yin et al. \(2009\)](#), and [Yin et al. \(2010a\)](#), [Lorbacher et al. \(2012\)](#)).

39.3 The non-Boussinesq steric effect

The non-Boussinesq steric effect and the boundary mass flux are the only two means, within the kinematic sea level equation (39.50), to impact global mean sea level. Impacts from the convergence of vertically integrated ocean currents exactly vanish in the global mean and so are of no direct relevance to global sea level.

Certainly ocean currents are important for global mean sea level. In particular, they impact on turbulent mixing, which in turn modifies the way heat irreversibly penetrates into the ocean. So in that regard ocean dynamics is of fundamental importance for global sea level. But it is through the non-Boussinesq steric effect that mixing impacts global mean sea level. It is also through the non-Boussinesq steric effect that boundary fluxes of buoyancy impact global mean sea level. Consequently, to understand mechanisms for changes to global mean sea level requires a framework to explore the non-Boussinesq steric effect.

39.3.1 Defining the non-Boussinesq steric effect

Version I of the sea level equation (39.50), repeated here for completeness,

$$\frac{\partial \eta}{\partial t} = \frac{Q_m}{\rho(\eta)} - \nabla \cdot \mathbf{U} - \int_{-H}^{\eta} dz \frac{1}{\rho} \frac{d\rho}{dt}, \quad (39.63)$$

provides a means to kinematically partition various physical processes impacting sea level evolution. We are particularly interested in the time tendency

$$\left(\frac{\partial \eta}{\partial t} \right)_{\text{non-bouss steric}} \equiv - \left(\int_{-H}^{\eta} dz \frac{1}{\rho} \frac{d\rho}{dt} \right), \quad (39.64)$$

which we refer to as the *non-Boussinesq steric effect*, since it is absent in Boussinesq fluids (Greatbatch, 1994). The time derivative acting on *in situ* density in the non-Boussinesq steric effect (39.64) is a material derivative, rather than an Eulerian tendency. This detail is fundamental to the formulation given in the following, and we return to its implications throughout this paper.

39.3.2 Vertical motion across pressure surfaces

To initiate a discussion of the non-Boussinesq steric effect, we use the material evolution of *in situ* density in the form of equation (39.41) to write

$$-\frac{1}{\rho} \frac{d\rho}{dt} = \alpha \frac{d\Theta}{dt} - \beta \frac{dS}{dt} - \frac{1}{\rho c_{\text{sound}}^2} \frac{dp}{dt}. \quad (39.65)$$

The material evolution of *in situ* density is thus affected by the material evolution of buoyancy, through material changes in temperature and salinity, and by material evolution of pressure, with pressure evolution the focus of this subsection.

To garner some exposure to the physics of dp/dt as it appears in equation (39.65), consider the special case of a hydrostatic fluid, where the volume per time per horizontal area of fluid crossing a surface of constant hydrostatic pressure is given by (see Section 6.7 of Griffies (2004))

$$\begin{aligned} w^{(p)} &= \frac{\partial z}{\partial p} \frac{dp}{dt} \\ &= -(\rho g)^{-1} \frac{dp}{dt}. \end{aligned} \quad (39.66)$$

The transport measured by $w^{(p)}$ is the pressure-coordinate analog of the vertical velocity component $w = dz/dt$ in a geopotential coordinate representation of the vertical direction. To the extent that pressure surfaces are well approximated by depth surfaces, $w^{(p)} \approx w$. Fluid moving into regions of increasing hydrostatic pressure ($dp/dt > 0$) represents downward movement of fluid, with $w^{(p)} < 0$ in this case. Conversely, motion into decreasing hydrostatic pressure represents upward motion, with $w^{(p)} > 0$.

We may identify the following contribution to the non-Boussinesq steric effect associated with vertical motion across hydrostatic pressure surfaces

$$\begin{aligned} \left(\frac{\partial \eta}{\partial t} \right)_{dp/dt} &\equiv - \int_{-H}^{\eta} dz \frac{1}{\rho c_{\text{sound}}^2} \frac{dp}{dt} \\ &= g \int_{-H}^{\eta} dz \frac{w^{(p)}}{c_{\text{sound}}^2} \\ &= g (H + \eta) \left(\frac{w^{(p)}}{c_{\text{sound}}^2} \right)^z \end{aligned} \quad (39.67)$$

where we introduced the vertical average operator for an ocean column. In columns where the vertically averaged vertical motion is upward, the column stretches, thus imparting a positive sea level tendency. The opposite occurs for vertically averaged downward fluid motion. Fluid generally moves across pressure surfaces under adiabatic and isohaline parcel motions, as in the presence of gravity or planetary waves. Additionally, diapycnal mixing generally gives rise to vertical transport, for example as seen in the case of a horizontally unstratified fluid in the presence of vertical diffusion. Diagnosing the term (39.67) does not identify the cause of the dia-pressure motion; it merely quantifies the effects of this motion on global mean sea level.

To gain a sense for the scale of the term (39.67), approximate the shallow water gravity wave speed as

$$c_{\text{grav}}^2 \approx g (H + \eta), \quad (39.68)$$

thus leading to

$$\left(\frac{\partial \eta}{\partial t}\right)_{dp/dt} \approx \overline{w^{(p)} (c_{\text{grav}}/c_{\text{sound}})^2}^z, \quad (39.69)$$

where $(c_{\text{grav}}/c_{\text{sound}})^2 \leq 10^{-2}$.

The contribution (equation (39.67)) to the non-Boussinesq steric effect possesses a higher wave number structure and larger magnitude locally than contributions from surface water fluxes and from the convergence of column integrated mass transport. We expect the high wave number power to increase in simulations with refined resolutions and realistic diurnal and astronomical (tidal) forcing (absent in this simulation), such that internal gravity wave activity is further promoted. In this way, dynamically active regions, such as the Southern Ocean, act as a barotropic wavemaker through vertical motion. As explained in Section 39.2.3 and [Greatbatch \(1994\)](#), contributions to the non-Boussinesq steric effect act dynamically in a manner analogous to mass forcing, leaving a regional low frequency field that is smoother than forcing from the non-Boussinesq steric effect. We thus do not generally see an imprint of the non-Boussinesq steric effect on low frequency (i.e., longer than monthly) regional sea level.

39.3.3 An expanded form of the kinematic sea level equation

We now focus on how material changes in temperature and salinity contribute to the non-Boussinesq steric effect. To do so, assume that the material evolution of conservative temperature and salinity is given by the convergence of a subgrid scale flux plus a source term

$$\rho \frac{d\Theta}{dt} = -\nabla \cdot \mathbf{J}^{(\Theta)} + \rho \mathcal{S}^\theta \quad (39.70)$$

$$\rho \frac{dS}{dt} = -\nabla \cdot \mathbf{J}^{(S)} + \rho \mathcal{S}^S. \quad (39.71)$$

The conservative temperature flux $\mathbf{J}^{(\Theta)}$ and the salt flux $\mathbf{J}^{(S)}$ include their respective diffusive and skew fluxes, as formulated by equation (39.33). Note that the temperature flux $\mathbf{J}^{(\Theta)}$ may also include the effects from penetrative shortwave radiation (e.g., [Iudicone et al., 2008](#)). The forms for the conservative temperature and salinity equations (39.70) and (39.71) lead to (see equation (39.65))

$$\begin{aligned} -\alpha \frac{d\Theta}{dt} + \beta \frac{dS}{dt} &= \left(\frac{\partial v}{\partial \Theta}\right) \nabla \cdot \mathbf{J}^{(\Theta)} + \left(\frac{\partial v}{\partial S}\right) \nabla \cdot \mathbf{J}^{(S)} - \alpha \mathcal{S}^\theta + \beta \mathcal{S}^S \\ &= \nabla \cdot \left(\frac{\partial v}{\partial \Theta} \mathbf{J}^{(\Theta)} + \frac{\partial v}{\partial S} \mathbf{J}^{(S)}\right) - \mathbf{J}^{(\Theta)} \cdot \nabla \left(\frac{\partial v}{\partial \Theta}\right) - \mathbf{J}^{(S)} \cdot \nabla \left(\frac{\partial v}{\partial S}\right) - \alpha \mathcal{S}^\theta + \beta \mathcal{S}^S \end{aligned} \quad (39.72)$$

where

$$v = \rho^{-1} \quad (39.73)$$

is the specific volume, and its derivatives are given by

$$\frac{\partial v}{\partial \Theta} = \frac{\alpha}{\rho}, \quad (39.74)$$

$$\frac{\partial v}{\partial S} = -\frac{\beta}{\rho}, \quad (39.75)$$

where $\alpha = -\rho^{-1} \partial \rho / \partial \Theta$ and $\beta = \rho^{-1} \partial \rho / \partial S$ are the thermal expansion and haline contraction coefficients introduced in Section 39.1.4. Bringing these results together leads to

$$\begin{aligned} \left(\frac{\partial \eta}{\partial t} \right)_{\text{non-bouss steric}} &= - \int_{-H}^{\eta} dz \nabla \cdot [(\alpha/\rho) \mathbf{J}^{(\Theta)} - (\beta/\rho) \mathbf{J}^{(S)}] \\ &\quad + \int_{-H}^{\eta} dz (\alpha S^{\Theta} - \beta S^S) \\ &\quad + \int_{-H}^{\eta} dz \left(\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) - \frac{1}{\rho c_{\text{sound}}^2} \frac{dp}{dt} \right). \end{aligned} \quad (39.76)$$

39.3.3.1 Exposing the boundary fluxes of temperature and salinity

We expose the surface and bottom boundary fluxes through use of the following identity

$$\begin{aligned} \int_{-H}^{\eta} dz \nabla \cdot [(\alpha/\rho) \mathbf{J}^{(\Theta)} - (\beta/\rho) \mathbf{J}^{(S)}] &= \nabla_z \cdot \left(\int_{-H}^{\eta} dz [(\alpha/\rho) \mathbf{J}^{(\Theta)} - (\beta/\rho) \mathbf{J}^{(S)}] \right) \\ &\quad + \nabla(z - \eta) \cdot [(\alpha/\rho) \mathbf{J}^{(\Theta)} - (\beta/\rho) \mathbf{J}^{(S)}]_{z=\eta} \\ &\quad - \nabla(z + H) \cdot [(\alpha/\rho) \mathbf{J}^{(\Theta)} - (\beta/\rho) \mathbf{J}^{(S)}]_{z=-H}. \end{aligned} \quad (39.77)$$

Tracer fluxes through the ocean bottom $z = -H(x, y)$ are written as

$$\begin{aligned} Q^{(C)} &= \mathbf{J}^{(C)} \cdot \hat{\mathbf{n}} \left(\frac{dA_{\hat{\mathbf{n}}}}{dA} \right) \\ &= \mathbf{J}^{(C)} \cdot \nabla(z + H), \end{aligned} \quad (39.78)$$

where $\hat{\mathbf{n}} = -\nabla(z + H)/|\nabla(z + H)|$ is the bottom normal vector, and we used the relation (see Section 20.13.2 of Griffies (2004))

$$dA_{\hat{\mathbf{n}}} = |\nabla(z + H)| dA \quad \text{at } z = -H(x, y) \quad (39.79)$$

between the area element $dA_{\hat{\mathbf{n}}}$ on the ocean bottom, and its horizontal projection dA . Hence, the conservative temperature and salt fluxes through the ocean bottom are given by

$$Q^{(\Theta)} = \mathbf{J}^{(\Theta)} \cdot \nabla(z + H) \quad \text{at } z = -H \quad (39.80)$$

$$Q^{(S)} = \mathbf{J}^{(S)} \cdot \nabla(z + H) \quad \text{at } z = -H. \quad (39.81)$$

The flux

$$\text{enthalpy flux} = c_p Q^{(\Theta)} \quad (39.82)$$

is the heat, or more precisely the enthalpy, per time per horizontal area entering the ocean through the bottom, with

$$c_p \approx 3992.1 \text{ J kg}^{-1} \text{ K}^{-1} \quad (39.83)$$

the heat capacity for seawater at constant pressure, with the value given by IOC et al. (2010) the most precise. Geothermal heating is quite small relative to surface heat fluxes. However, the work of Adcroft et al. (2001) and Emile-Geay and Madec (2009) motivate retaining this contribution to the temperature equation, as there are some systematic impacts from geothermal heating over long periods. For climatological purposes, the introduction of salt to the ocean through the sea floor is negligible, in which case

$$Q^{(S)} = 0 \quad \text{at } z = -H. \quad (39.84)$$

Likewise, water may enter through the sea floor, but we know of no recommended dataset for doing so. Thus, we set $Q_m = 0$ at the bottom.

As for the ocean bottom, tracer fluxes through the ocean surface $z = \eta(x, y, t)$ are written as

$$\begin{aligned} Q^{(C)} &= -\mathbf{J}^{(C)} \cdot \hat{\mathbf{n}} \left(\frac{dA_{\hat{\mathbf{n}}}}{dA} \right) \\ &= -\mathbf{J}^{(C)} \cdot \nabla(z - \eta), \end{aligned} \quad (39.85)$$

where $\hat{\mathbf{n}} = \nabla(z - \eta)/|\nabla(z - \eta)|$ is the surface outward normal vector, and we used the relation (see Section 20.13.2 of Griffies (2004))

$$dA_{\hat{\mathbf{n}}} = |\nabla(z - \eta)| dA \quad \text{at } z = \eta \quad (39.86)$$

between the area element $dA_{\hat{\mathbf{n}}}$ on the ocean surface, and its horizontal projection dA . The sign convention is chosen so that $Q^{(C)} > 0$ signals the addition of tracer to the ocean. Hence, the conservative temperature and salt fluxes through the ocean surface take the form

$$Q^{(\Theta)} = -\mathbf{J}^{(\Theta)} \cdot \nabla(z - \eta) \quad \text{at } z = \eta \quad (39.87)$$

$$Q^{(S)} = -\mathbf{J}^{(S)} \cdot \nabla(z - \eta) \quad \text{at } z = \eta. \quad (39.88)$$

Again, the sign convention is chosen so that positive flux $Q^{(\Theta)}$ adds heat to the ocean, and positive flux $Q^{(S)}$ adds salt to the ocean. There is near zero salt flux through the ocean surface, except for the small exchange during the formation and melt of sea ice. We further discuss surface fluxes in Section 39.3.4.

39.3.3.2 Expanded form of the kinematic sea level equation

Bringing these results together leads to an expanded form of the kinematic sea level equation

$$\begin{aligned} \frac{\partial \eta}{\partial t} &= -\nabla_z \cdot \left(\int_{-H}^{\eta} dz \frac{(\alpha \mathbf{J}^{(\Theta)} - \beta \mathbf{J}^{(S)} + \rho \mathbf{u})}{\rho} \right) \\ &\quad + \int_{-H}^{\eta} dz (\alpha \mathcal{S}^{\theta} - \beta \mathcal{S}^S) \\ &\quad + \int_{-H}^{\eta} dz \left(\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) - \frac{1}{\rho c_{\text{sound}}^2} \frac{dp}{dt} \right) \\ &\quad + \left(\frac{\alpha Q^{(\Theta)} - \beta Q^{(S)} + Q_m}{\rho} \right)_{z=\eta} + \left(\frac{\alpha Q^{(\Theta)}}{\rho} \right)_{z=-H}. \end{aligned} \quad (39.89)$$

This expression provides a precise measure of how physical processes and boundary fluxes determine the kinematic evolution of sea level. In particular, this equation suggests the following physical interpretations, with reference made to Figure 39.2 for a schematic.

39.3.3.3 Buoyancy fluxes and mass fluxes

The first integral in the kinematic sea level equation (39.89) is the convergence of vertically integrated lateral subgrid scale flux of buoyancy (arising from the non-Boussinesq steric effect), along with the lateral advective flux of mass. When integrated over the global ocean, this convergence vanishes exactly. Hence, this term moves mass and buoyancy around in the ocean without altering the global mean sea level.

The combination $\alpha Q^{(\Theta)} - \beta Q^{(S)}$ represents the buoyancy flux crossing the ocean surface at $z = \eta(x, y, t)$. Boundary fluxes that increase net buoyancy of the global ocean increase global mean sea level. In particular, it is precisely through this effect that surface ocean warming acts to increase global mean sea level. These boundary buoyancy fluxes act in concert with the boundary mass flux Q_m to impact sea level through interactions at the ocean surface.

39.3.3.4 Interior sources

The second integral in equation (39.89) arises from the interior source of temperature and salinity, such as may arise from such parameterized processes as non-local mixing in the KPP boundary layer scheme (Large et al., 1994). The third integral in equation (39.89) is another source, which arises from the non-Boussinesq steric effect

$$\text{source} = \mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) - \frac{1}{\rho c_{\text{sound}}^2} \frac{dp}{dt}, \quad (39.90)$$

with the source having units of inverse time. There are three terms contributing to the source, the third of which we discussed in Section 39.3.2 relates to vertical motion across pressure surfaces. The first and second terms relate to the orientation of temperature and salinity subgrid scale fluxes in relation to thermodynamic properties of the fluid and geometric properties of the density surface. We consider special cases of the source term in Section 39.5 for diapycnal diffusion, approximated as vertical diffusion; Section 39.6 for neutral diffusion, where it is shown how cabbeling and thermobaricity, arising from parameterized mixing effects from mesoscale eddies along neutral directions, impact global mean sea level; and Section 39.7 for quasi-Stokes transport from mesoscale eddies as parameterized according to Gent and McWilliams (1990).

39.3.3.5 Concerning the absence of advective temperature and salt fluxes

It is notable that the expanded kinematic sea level equation (39.89) only involves the subgrid scale fluxes of temperature and salt, as well as their non-advective boundary fluxes. There are no interior nor boundary advective fluxes of buoyancy that directly impact sea level. This absence relates to our choice to focus on Version I of the sea level equation (39.50), in which the time derivative acting on *in situ* density is a material or Lagrangian derivative, not an Eulerian derivative. This point is of fundamental importance to how we identify physical processes impacting global mean sea level in Section 39.4 and further considered in Sections 39.5, 39.6, and 39.7. Hence, buoyancy changes impact global mean sea level only through processes that render a nonzero material change to *in situ* density, with such changes occurring through boundary fluxes, interior mixing, nonlinear equation of state effects, and subgrid scale eddy induced advection.

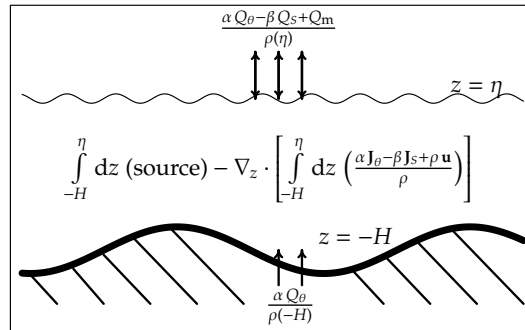


Figure 39.2: A schematic ocean basin as in Figure 39.1, but now illustrating the various boundary and internal processes that impact sea level according to the unpacked kinematic sea level equation (39.89). The surface mass flux, Q_m , is combined here with a surface buoyancy flux to yield the surface boundary forcing $\rho(\eta)^{-1} (\alpha Q^{(\Theta)} - \beta Q^{(S)} + Q_m)$. Likewise, the bottom boundary forcing arises from geothermal heating in the form $\alpha Q^{(\Theta)}/\rho(-H)$, which is a small, but always positive, contribution to global mean sea level evolution. In the ocean interior, the convergence of mass and buoyancy, along with a source term (equation (39.90)), yield the forcing $\int_{-H}^{\eta} dz (\text{source}) - \nabla_z \cdot \int_{-H}^{\eta} dz \rho^{-1} (\alpha \mathbf{J}^{(\Theta)} - \beta \mathbf{J}^{(S)} + \rho \mathbf{u})$.

39.3.4 Boundary fluxes of heat and salt

We now detail how boundary fluxes of heat, salt, and freshwater impact the non-Boussinesq steric effect appearing in equation (39.89). The global integral of these fluxes impact on the global mean sea level (Section 39.4). Start by noting that the contribution from the surface buoyancy flux,

$$\left(\frac{\partial \eta}{\partial t} \right)_{\text{surface buoyancy}} = \frac{\alpha Q^{(\Theta)} - \beta Q^{(S)}}{\rho(\eta)}, \quad (39.91)$$

is determined both by the boundary fluxes of heat and salt, as well as the boundary values for the thermal expansion coefficient, haline contraction coefficient, and density.

39.3.4.1 Boundary heat fluxes

The non-advective heat flux crossing the surface ocean boundary consists of the following contributions (in units of W m^{-2}),

$$Q_{\text{surface}} = Q_{\text{SW}} + Q_{\text{LW}} + Q_{\text{sens}} + Q_{\text{lat}} + Q_{\text{frazil}} \quad (39.92)$$

with a sign convention chosen so that positive fluxes add heat to the liquid seawater. What appears in the sea level equation is actually the surface flux of temperature, not heat. So it is necessary to convert between heat and temperature fluxes when considering the impacts on sea level. As noted by McDougall (2003), to convert from heat fluxes to fluxes of potential temperature requires the use of a non-constant specific heat capacity, which varies by roughly 5% over the globe. In contrast, converting between heat fluxes and conservative temperature fluxes is done with a constant specific heat capacity

$$Q_{\text{surface}} = c_p Q_{\text{surface}}^{(\Theta)}, \quad (39.93)$$

thus serving to further promote the use of conservative temperature.

We now summarize the various contributions to surface ocean heating.

- **SHORTWAVE:** The dominant heating occurs through the shortwave contribution $Q_{\text{SW}} > 0$. Shortwave radiation penetrates on the order of 10m to 100m into the ocean interior, with the distance depending on optical properties of seawater (see, e.g., Sweeney et al., 2005, and cited references).
- **LONGWAVE:** The longwave contribution $Q_{\text{LW}} < 0$ represents the net longwave energy that is re-radiated back to the atmosphere. Even though there are many occasions for backscattering, the net effect of longwave radiation is to cool the ocean.
- **SENSIBLE:** Sensible heating Q_{sens} arises from turbulent exchange with the atmosphere, and is generally parameterized by turbulent bulk formula. The sensible heat term typically cools the ocean surface.
- **LATENT:** Latent heating Q_{lat} cools the ocean, as it is the energy extracted from the ocean to vaporize liquid water that enters the atmosphere. Additionally, the latent heating term includes heat extracted from the ocean to melt solid runoff (i.e., calving land ice) or snow entering the liquid ocean. These latent heat terms are thus related to mass transport across the ocean surface according to

$$Q_{\text{lat}}^{\text{vapor}} = H^{\text{vapor}} Q_{\text{m}}^{\text{evap}} \quad (39.94)$$

$$Q_{\text{lat}}^{\text{melt}} = H^{\text{fusion}} (Q_{\text{m}}^{\text{calving}} + Q_{\text{m}}^{\text{snow}}), \quad (39.95)$$

where $H^{\text{vapor}} \approx 2.5 \times 10^6 \text{ J kg}^{-1}$ is the latent heat of vaporization, $Q_{\text{m}}^{\text{evap}}$ is the evaporative mass flux in units of $\text{kg m}^{-2} \text{ s}^{-1}$, $H^{\text{fusion}} \approx 3.34 \times 10^5 \text{ J kg}^{-1}$ is the latent heat of fusion, $Q_{\text{m}}^{\text{calving}}$ is the mass flux of calving land ice entering the ocean, and $Q_{\text{m}}^{\text{snow}}$ is the mass flux of frozen precipitation falling on the ocean surface.

- **FRAZIL:** As the temperature of seawater cools to the freezing point, sea ice is formed, initially through the production of frazil. Operationally in an ocean model, liquid water can be supercooled at any particular time step through surface fluxes and transport. An adjustment process heats the liquid water back to the freezing point, with this positive heat flux Q_{frazil} extracted from the ice model as frazil sea ice is formed.

In addition to the above listed heat fluxes, there is a heat flux associated with the heat content of water transferred across the ocean surface. However, this flux represents an advective (or mass transport) heat flux that is captured by the advective term in the material time derivative.

The total non-advective boundary flux of heat affecting ocean heat content is the sum of the surface flux detailed above, and the bottom flux arising from geothermal heating

$$Q_{\text{heat}} = Q_{\text{surface}} + Q_{\text{geothermal}} \quad (39.96)$$

The geothermal contribution to heating is quite small (less than 0.1 W m^{-2} locally), and so contributes only a very small amount to the global sea level evolution.

39.3.4.2 Boundary salt fluxes and sea ice

For most purposes of ocean climate modelling, the mass flux of salt into the liquid ocean is limited to exchanges associated with the formation and melt of sea ice

$$Q^{(S)} = Q_{\text{ice-ocean salt exchange}}, \quad (39.97)$$

which arises since the salinity of sea ice is nonzero (on the order of 5 parts per thousand). Hence, as sea ice forms, it extracts a small amount of salt from the liquid ocean, and a larger amount of liquid freshwater. The converse happens upon melting sea ice.

So how does the formation and melt of sea ice impact global mean sea level? First, there is a transfer of mass between the solid and liquid phases. In particular, salt and freshwater transfer affect the mass of the liquid ocean through the mass flux Q_m appearing in equation (39.89) in the form

$$\left(\frac{\partial \eta}{\partial t} \right)_{\text{sea ice mass exchange}} = \frac{Q^{(S)} + Q_w}{\rho(\eta)}, \quad (39.98)$$

with Q_w the mass flux from freshwater. As salt and freshwater are added to the ocean through sea ice melt, they raise the global mean liquid sea level according to this expression. However, the transfer of mass from the solid to liquid phase leads to near zero net change in the effective sea level, since the liquid ocean response to sea ice loading is consistent with an inverse barometer. This mass transfer includes the transfer of both freshwater and salt. However, most ocean and sea ice models used for climate studies ignore the mass of salt when computing the liquid ocean mass and solid sea ice mass. So in practice, the mass transfer referred to here occurs only through the transfer of freshwater.

In addition to exchanging mass, there is an exchange of buoyancy as salt is moved between the solid and liquid phases. Changing buoyancy at the ocean surface impacts the non-Boussinesq steric effect according to

$$\left(\frac{\partial \eta}{\partial t} \right)_{\text{surface salt buoyancy}} \equiv - \left(\frac{\beta Q^{(S)}}{\rho} \right). \quad (39.99)$$

As salt is added to the upper ocean through ice melt, this process removes buoyancy.

39.4 Evolution of global mean sea level

Global mean sea level is of great interest for climate studies. Indeed, it is often the only field that climate models report when considering projections for sea level changes due to human-induced global warming. The purpose of this section is to formulate evolution equations for the global mean sea level which, in an ocean with constant horizontal area, are equivalent to equations for global ocean volume.

39.4.1 Preliminaries

We begin by establishing some notation and restating assumptions. For this purpose, write the volume of liquid seawater in the global ocean in the following equivalent manners

$$\begin{aligned}\mathcal{V} &= \int_{\text{globe}} dA \int_{-H}^{\eta} dz \\ &= \int_{\text{globe}} dA (\eta + H) \\ &= \mathcal{A} (\bar{H} + \bar{\eta}).\end{aligned}\tag{39.100}$$

In the final step, we introduced the global ocean surface area

$$\mathcal{A} = \int_{\text{globe}} dA = \int_{\text{globe}} dx dy,\tag{39.101}$$

the global mean sea level

$$\bar{\eta} = \mathcal{A}^{-1} \int_{\text{globe}} \eta dA,\tag{39.102}$$

and the global mean ocean depth beneath a resting sea surface at $z = 0$

$$\bar{H} = \mathcal{A}^{-1} \int_{\text{globe}} H dA.\tag{39.103}$$

We assume throughout this paper that the ocean surface area remains constant in time. That is, we do not consider changes in sea level associated with shoreline changes. We also assume that the ocean depth $z = -H(x, y)$ remains constant in time. An evolution equation for the global ocean volume is therefore equivalent to an evolution equation for global mean sea level

$$\partial_t \mathcal{V} = \mathcal{A} \partial_t \bar{\eta}.\tag{39.104}$$

39.4.2 Global mean sea level and the non-Boussinesq steric effect

The first version of the global mean sea level equation is formulated starting from Version I of the kinematic sea level equation (39.50), rewritten here for completeness

$$\frac{\partial \eta}{\partial t} = \frac{Q_m}{\rho(\eta)} - \nabla \cdot \mathbf{U} - \int_{-H}^{\eta} dz \frac{1}{\rho} \frac{d\rho}{dt}.\tag{39.105}$$

A global integration, with no-flux side walls or periodic domains, leads to

$$\partial_t \bar{\eta} = \left(\overline{\frac{Q_m}{\rho(\eta)}} \right) - \frac{\mathcal{V}}{\mathcal{A}} \left\langle \frac{1}{\rho} \frac{d\rho}{dt} \right\rangle,\tag{39.106}$$

where

$$\langle \psi \rangle = \mathcal{V}^{-1} \int_{\text{globe}} dA \int_{-H}^{\eta} \psi dz\tag{39.107}$$

is the global volume mean of a field. The relation (39.106) says that the global mean sea level changes in time according to (A) the area mean of the mass flux multiplied by the specific volume of seawater at the ocean surface, and (B) the global mean of the non-Boussinesq steric effect.

To uncover some of the physical processes that impact global sea level, consider the unpacked version of equation (39.106), obtained by taking a global area integral of the sea level equation (39.89)

$$\begin{aligned} \mathcal{A} \partial_t \bar{\eta} = & \int_{\text{globe}(z=\eta)} dA \left(\frac{Q_m + \alpha Q^{(\Theta)} - \beta Q^{(S)}}{\rho} \right) + \int_{\text{globe}(z=-H)} dA \left(\frac{\alpha Q^{(\Theta)} - \beta Q^{(S)}}{\rho} \right) \\ & + \int_{\text{globe}} \left(\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) - \frac{1}{\rho c_{\text{sound}}^2} \frac{dp}{dt} \right) dV \\ & + \int_{\text{globe}} (\alpha \mathcal{S}^\theta - \beta \mathcal{S}^S) dV. \end{aligned} \quad (39.108)$$

To reach this result required the following steps:

1. The direct impacts of ocean currents drop out from a global area integral of equation (39.89), due to the no-normal flow solid earth boundary condition, or periodic boundary condition.
2. We assume zero water fluxes entering the ocean through the solid earth boundary.
3. There may be nonzero boundary heat or salt fluxes, though as noted in Section 39.3.4.2, salt exchanged through ocean boundaries is generally limited to a small exchange associated with sea ice formation and melt. Correspondingly, we drop the bottom salt flux.

39.4.3 Global mean sea level and the global steric effect

Equation (39.106), and its unpacked version (39.108), help us to understand how physical processes and boundary effects impact the global mean sea level. However, if one is only interested in the net effect on sea level, then an alternative formulation is appropriate. This second version of the evolution equation for global mean sea level is developed by introducing the following relation between the total mass of liquid seawater, total volume of seawater, and global mean seawater density,

$$\mathcal{M} = \mathcal{V} \langle \rho \rangle. \quad (39.109)$$

In this relation,

$$\mathcal{M} = \int_{\text{globe}} dA \int_{-H}^{\eta} \rho dz \quad (39.110)$$

is the global liquid seawater mass, and \mathcal{V} is the global volume of seawater (equation (39.100)). It follows that

$$\partial_t \mathcal{M} = \mathcal{V} \partial_t \langle \rho \rangle + \langle \rho \rangle \partial_t \mathcal{V}. \quad (39.111)$$

An area integration of the mass budget (39.53) indicates that total seawater mass changes if there is a nonzero mass flux across the ocean surface

$$\partial_t \mathcal{M} = \mathcal{A} \overline{Q_m} \quad (39.112)$$

where

$$\overline{Q_m} = \mathcal{A}^{-1} \int_{\text{globe}} Q_m dA \quad (39.113)$$

is the global mean mass per horizontal area per time crossing the ocean boundaries. Bringing these results together, and recalling that $\partial_t \mathcal{V} = \mathcal{A} \partial_t \bar{\eta}$ (equation (39.104)) leads to the second evolution equation for the global mean sea level

$$\partial_t \bar{\eta} = \frac{\overline{Q_m}}{\langle \rho \rangle} - \left(\frac{\mathcal{V}}{\mathcal{A}} \right) \frac{\partial_t \langle \rho \rangle}{\langle \rho \rangle}. \quad (39.114)$$

As in the formulation given in Section 39.4.2, there is no direct contribution from ocean currents, since they act to redistribute volume without changing the total volume. The first term on the right hand side

of equation (39.114) alters sea level by adding or subtracting volume from the ocean through the surface boundary, and the second term arises from temporal changes in the global mean density. We refer to the second term as the *global steric effect* to make a distinction from the global mean of the non-Boussinesq steric effect discussed in Section 39.4.2. Both terms on the right hand side of equation (39.114) are readily diagnosed from a model simulation, thus providing a means to partition the change in global mean sea level into a contribution from boundary fluxes and one from global steric changes.

We comment here on the dominance of changes in global mean temperature on the global steric effect in equation (39.114). For this purpose, write the time tendency of global mean density in the form

$$\partial_t \ln \langle \rho \rangle = -\alpha_{\text{bulk}} \partial_t \langle \Theta \rangle + \beta_{\text{bulk}} \partial_t \langle S \rangle + \frac{1}{(\rho c_{\text{sound}}^2)_{\text{bulk}}} \partial_t \langle p \rangle. \quad (39.115)$$

This equation defines α_{bulk} as a bulk, or *effective*, thermal expansion coefficient, β_{bulk} as a bulk haline contraction coefficient, and $(\rho c_{\text{sound}}^2)_{\text{bulk}}$ as a bulk density times the squared sound speed. These coefficients can be thought of as best fit parameters for the linear relation (39.115) connecting changes in global mean sea level to global mean temperature, salinity, and pressure. For the case that the global mean salinity remains constant, which is roughly the case for most climate purposes (see Section 39.3.4.2), and when changes in pressure are sub-dominant, as they generally are, then global mean density changes are dominated by global mean temperature changes. Hence, the global mean sea level equation (39.114) takes on the approximate form

$$\partial_t \bar{\eta} \approx \frac{\overline{Q_m}}{\langle \rho \rangle} + \left(\frac{\nu \alpha_{\text{bulk}}}{\mathcal{A}} \right) \partial_t \langle \Theta \rangle. \quad (39.116)$$

This approximate result provides a very good indicator of the 20th century global mean sea level rise associated with ocean warming (Church and Gregory, 2001; Nicholls and Cazenave, 2010).

39.4.4 Relating global steric to non-Boussinesq steric

Equating the two equations (39.106) and (39.114) for global mean sea level yields the identity

$$\left(\frac{\overline{Q_m}}{\rho(\eta)} \right) - \frac{\nu}{\mathcal{A}} \left\langle \frac{1}{\rho} \frac{d\rho}{dt} \right\rangle = \frac{\overline{Q_m}}{\langle \rho \rangle} - \left(\frac{\nu}{\mathcal{A}} \right) \frac{\partial_t \langle \rho \rangle}{\langle \rho \rangle}. \quad (39.117)$$

In the special case of zero surface boundary fluxes of mass, the global mean of the non-Boussinesq steric effect is equal to the global steric effect

$$\left\langle \frac{1}{\rho} \frac{d\rho}{dt} \right\rangle = \frac{\partial_t \langle \rho \rangle}{\langle \rho \rangle} \quad \text{if } Q_m = 0. \quad (39.118)$$

This result motivates a commonly used method to adjust the sea level in Boussinesq models, even in the presence of nonzero boundary fluxes of mass.

39.5 Vertical diffusion and global mean sea level

This section is the first of three to further explore the non-Boussinesq steric effect discussed in Sections 39.3 and 39.4. The focus here is on how subgrid scale fluxes of conservative temperature and salinity arising from vertical diffusion in the ocean interior determine patterns of the non-Boussinesq steric effect. Vertical diffusion is the method used by global ocean climate models to parameterize the impacts from diapycnal mixing processes. Section 39.6 considers the same questions for neutral diffusion, and Section 39.7 considers parameterized quasi-Stokes transport. By exposing patterns of the non-Boussinesq steric effect associated with these parameterizations, we provide a framework to explore where the global mean sea level in ocean climate models is impacted by subgrid scale physical parameterizations.

39.5.1 Contributions from vertical diffusion

Recall that we have already discussed in Sections 39.3.3 and 39.3.4 those contributions from boundary fluxes of buoyancy. Hence, the impacts of concern here from vertical diffusion involve just contributions to the source term (equation (39.90))

$$\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) = -\rho \left[D^\Theta \partial_z \Theta \partial_z(\alpha/\rho) - D^S \partial_z S \partial_z(\beta/\rho) \right]. \quad (39.119)$$

Here, we consider the tracer flux to equal that arising from downgradient vertical diffusion

$$\mathbf{J}^{(\Theta)} = -\rho D^\Theta \partial_z \Theta \hat{\mathbf{z}} \quad (39.120)$$

$$\mathbf{J}^{(S)} = -\rho D^S \partial_z S \hat{\mathbf{z}}, \quad (39.121)$$

with eddy diffusivities for conservative temperature and salinity, $D^\Theta > 0$ and $D^S > 0$. This flux is meant to parameterize mixing arising from diapycnal processes (see Section 7.4.3 of Griffies (2004) for discussion). In the ocean interior, the diffusivities have values $D \approx 10^{-6} \text{ m}^2 \text{ s}^{-1}$ at the equator, beneath the regions of strong vertical shears associated with the equatorial undercurrent (Gregg et al., 2003), and $D \approx 10^{-5} \text{ m}^2 \text{ s}^{-1}$ in the middle latitudes (Ledwell et al., 1993, 2011), with far larger values near rough topography and other boundary regions (Polzin et al., 1997; Naveira-Garabato et al., 2004). Additionally, D can be set to a very large value in gravitationally unstable regions. Finally, differences in the temperature and salinity diffusivities arise from double diffusive processes (Schmitt, 1994).

39.6 Neutral diffusion and global mean sea level

Mesoscale eddy motions impact the large-scale ocean temperature and salinity distributions in important and nontrivial ways, with their parameterization in coarsely resolved climate models a longstanding focus of theoretical physical oceanography. Amongst the common means for parameterizing these effects is to prescribe a diffusion of temperature and salinity oriented according to the planes tangent to locally referenced potential density surfaces (Solomon, 1971; Redi, 1982; Olbers et al., 1985; McDougall and Church, 1986; Cox, 1987; Gent and McWilliams, 1990; Griffies et al., 1998). The purpose of this section is to detail how such neutral diffusion of temperature and salinity affects the non-Boussinesq steric effect.

To expose how neutral diffusion impacts sea level first requires the establishment of salient properties of neutral diffusion. Thereafter, we describe how the physical processes of cabbeling and thermobaricity impact global mean sea level. Note that following McDougall (1987b) (see also Klocker and McDougall (2010a), Klocker and McDougall (2010b), and IOC et al. (2010)), we are concerned with cabbeling and thermobaricity as diapycnal transport processes that arise from a coarse-grained perspective of the mixing of fluid parcels by mesoscale eddies along neutral directions. We are not concerned with how molecular diffusion or vertical diffusion can be formulated in terms of micro-scale cabbeling and thermobaricity processes.

39.6.1 Neutral directions and neutral tangent plane

Following McDougall (1987a), we introduce the notion of a neutral direction by considering an infinitesimal displacement $d\mathbf{x}$, in which the *in situ* density changes according to

$$d\rho = \rho d\mathbf{x} \cdot \left(-\alpha \nabla \Theta + \beta \nabla S + \frac{1}{\rho c_s^2} \nabla p \right). \quad (39.122)$$

Under adiabatic and isohaline motions, the density change is associated just with pressure changes

$$(d\rho)_{\text{adiabatic/isohaline}} = \rho d\mathbf{x} \cdot \left(\frac{1}{\rho c_s^2} \nabla p \right). \quad (39.123)$$

Therefore, if we consider an adiabatic and isohaline displacement, the difference in density between the parcel and the surrounding environment is given by

$$\begin{aligned} d\rho - (d\rho)_{\text{adiabatic/isohaline}} &= \rho \, d\mathbf{x} \cdot (-\alpha \nabla \Theta + \beta \nabla S) \\ &= \rho \, d\mathbf{x} \cdot \hat{\gamma} |-\alpha \nabla \Theta + \beta \nabla S|, \end{aligned} \quad (39.124)$$

where the *dianeutral unit vector* is defined by

$$\hat{\gamma} = \frac{-\alpha \nabla \Theta + \beta \nabla S}{|-\alpha \nabla \Theta + \beta \nabla S|}. \quad (39.125)$$

At each point in the fluid, displacements orthogonal to $\hat{\gamma}$ define neutral directions, and the accumulation of such displacements define a *neutral tangent plane*.

39.6.2 Fluxes computed with locally orthogonal coordinates

There are two means to write the neutral diffusion fluxes for a tracer. The first way follows from [Redi \(1982\)](#), who considers the three dimensional fluxes computed parallel to the local neutral direction. This is an approach that arises naturally if working in local orthogonal coordinates ([Griffies et al., 1998](#); [Griffies, 2004](#))

$$\mathbf{J}^{(\Theta)} = -\rho A_n [\nabla \Theta - \hat{\gamma} (\hat{\gamma} \cdot \nabla \Theta)] \quad (39.126)$$

$$\mathbf{J}^{(S)} = -\rho A_n [\nabla S - \hat{\gamma} (\hat{\gamma} \cdot \nabla S)]. \quad (39.127)$$

In this equation, $A_n > 0$ is a diffusivity (units of squared length per time) setting the magnitude of the neutral diffusive fluxes. Geometrically, the neutral diffusive fluxes of conservative temperature, $\mathbf{J}^{(\Theta)}$, and salinity, $\mathbf{J}^{(S)}$, are proportional to that portion of the tracer gradient parallel to the neutral directions. Hence, by construction,

$$\mathbf{J}^{(\Theta)} \cdot \hat{\gamma} = 0 \quad (39.128)$$

$$\mathbf{J}^{(S)} \cdot \hat{\gamma} = 0. \quad (39.129)$$

39.6.3 Fluxes computed with projected neutral coordinates

The second means for computing neutral diffusive fluxes follows from [Gent and McWilliams \(1990\)](#), whereby we employ the projected coordinates used in generalized vertical coordinate models (see [Starr \(1945\)](#), [Bleck \(1978\)](#), [McDougall \(1995\)](#), or Chapter 6 of [Griffies \(2004\)](#)), here applied to the locally defined neutral tangent plane as in [McDougall \(1987a\)](#). To use this framework requires a vertically stable stratification. Here, lateral gradients of a tracer are computed along the neutral direction, but the lateral distance used to compute the gradient is taken as the distance on the horizontal plane resulting from projecting the neutral slope onto the constant depth surfaces (see, for example, Figure 6.4 in [Griffies, 2004](#)). For this case, the horizontal and vertical components of the neutral diffusion tracer fluxes take the form

$$\mathbf{J}^h = -A_n \rho \nabla_n C \quad (39.130)$$

$$J^z = -A_n \rho \mathbf{S} \cdot \nabla_n C, \quad (39.131)$$

where the projected lateral gradient operator is given by

$$\nabla_n = \nabla_z + \mathbf{S} \partial_z. \quad (39.132)$$

Just as for the fluxes associated with orthogonal coordinates, we have $\mathbf{J} \cdot \hat{\gamma} = 0$, which ensures that the fluxes are indeed aligned with neutral directions. Furthermore, these flux components are equal to the [Redi \(1982\)](#) flux components in the limit that the magnitude of the neutral slope

$$\mathbf{S} = - \left(\frac{-\alpha \nabla_z \Theta + \beta \nabla_z S}{-\alpha \partial_z \Theta + \beta \partial_z S} \right) \quad (39.133)$$

becomes small. Because of this connection, the neutral diffusion fluxes associated with projected neutral coordinates are known as the *small slope* fluxes. However, there is no approximation involved with using the small slope fluxes, so long as the vertical stratification is stable. Rather, their use represents a choice associated with how we measure lateral distances when computing gradients. Since the small slope fluxes are directly related to those along-isopycnal fluxes used in isopycnal models, they are more commonly used in level models than the fluxes implied by Redi (1982). The small slope fluxes are also simpler to compute, as they involve fewer terms.

39.6.4 Compensating neutral diffusive fluxes of temperature and salinity

It is straightforward to show that either of the neutral diffusive fluxes defined in Sections 39.6.2 or 39.6.3 satisfy the identity

$$\alpha \mathbf{J}^{(\Theta)} - \beta \mathbf{J}^{(S)} = 0. \quad (39.134)$$

That is, each component of the neutral diffusive flux of buoyancy vanishes, by definition. This is a key property of neutral diffusion, which in particular means that neutral diffusion directly contributes to the non-Boussinesq steric effect only through the source term defined by equation (39.90).

39.6.5 The cabbeling and thermobaricity parameters

To help further elucidate how neutral diffusion impacts the non-Boussinesq steric effect, and hence global mean sea level, we massage the source term (39.90) for the case of neutral diffusive fluxes, with the final result being

$$\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) = \rho^{-1} \mathbf{J}^{(\Theta)} \cdot (\mathcal{T} \nabla p + \mathcal{C} \nabla \Theta). \quad (39.135)$$

The cabbeling parameter \mathcal{C} and thermobaricity parameter \mathcal{T} are given by equations (39.136) and (39.141) discussed in the following. We now consider some of the physical implications of this result.

39.6.6 Physical aspects of cabbeling

Consider the mixing of two water parcels along a neutral direction through the action of mesoscale eddies. Let the parcels separately have distinct conservative temperature and/or salinity, but equal locally referenced potential density. If the equation of state were linear (Section 39.1.5), then the resulting mixed parcel would have the same density as the unmixed separate parcels. Due to the more general equilibrium thermodynamics in the ocean, in which there is a dependence of density on temperature, salinity, and pressure, the mixed parcel actually has a different density. Furthermore, the density of the mixed parcel is greater than the unmixed parcels. This densification upon mixing is a physical process known as *cabbeling* (McDougall, 1987b).

The sign definite nature of cabbeling (i.e., cabbeling always results in denser parcels after mixing) is a direct result of the geometry of the locally referenced potential density surface when viewed in conservative temperature and salinity space. This property in turn manifests with the following inequality for the cabbeling parameter

$$\mathcal{C} = \frac{\partial \alpha}{\partial \Theta} + 2 \frac{\alpha}{\beta} \frac{\partial \alpha}{\partial S} - \left(\frac{\alpha}{\beta} \right)^2 \frac{\partial \beta}{\partial S} \geq 0, \quad (39.136)$$

which is an empirical property of the ocean's equilibrium thermodynamics (IOC et al., 2010).

Downgradient neutral diffusion is meant to parameterize the mesoscale eddy induced mixing of tracers along neutral directions. We verify that the neutral diffusive flux considered thus far is indeed downgradient by considering the small slope neutral diffusive flux (equations (39.130) and (39.131)) to render

$$\nabla \Theta \cdot \mathbf{J}^{(\Theta)} = -A_n \rho (\nabla_n \Theta)^2 \leq 0. \quad (39.137)$$

Likewise, the neutral diffusive flux (39.126) from Redi (1982) yields

$$\nabla \Theta \cdot \mathbf{J}^{(\Theta)} = -A_n \rho [(\nabla \Theta)^2 - (\hat{\gamma} \cdot \nabla \Theta)^2] \leq 0. \quad (39.138)$$

Given this downgradient nature of the neutral diffusive fluxes, we have

$$\rho^{-1} \mathcal{C} \mathbf{J}^{(\Theta)} \cdot \nabla \Theta \leq 0, \quad (39.139)$$

thus providing a mathematical expression for the cabbeling process. That is, cabbeling results in a positive material evolution of density; i.e., density increases due to cabbeling, and this process can be interpreted as a downward or negative diapycnal transport (McDougall, 1987b).

An increase in density through cabbeling results in the reduction of sea level due to the compression of the fluid column as manifest in the non-Boussinesq steric effect

$$\left(\frac{\partial \eta}{\partial t} \right)_{\text{cab}} = - \int_{-H}^{\eta} dz \mathcal{C} A_n (\nabla_n \Theta)^2 < 0. \quad (39.140)$$

39.6.7 Physical aspects of thermobaricity

The thermobaricity parameter

$$\mathcal{T} = \beta \partial_p \left(\frac{\alpha}{\beta} \right) \quad (39.141)$$

is nonzero due to pressure dependence of the ratio of thermal expansion coefficient to haline contraction coefficient. As both thermal and haline effects are present, the parameter \mathcal{T} is more precisely split into two terms

$$\mathcal{T} = \frac{\partial \alpha}{\partial p} - \frac{\alpha}{\beta} \frac{\partial \beta}{\partial p}. \quad (39.142)$$

Thermobaricity is the common name for the sum, since pressure variations in the thermal expansion coefficient dominate those of the haline contraction coefficient. Since the thermal expansion coefficient generally increases as pressure increases, the thermobaricity parameter is typically positive. Since neutral diffusive fluxes need not be oriented in a special manner relative to the pressure gradient, there is no sign-definite nature to the thermobaricity source term

$$\begin{aligned} \rho^{-1} \mathbf{J}^{(\Theta)} \cdot \mathcal{T} \nabla p &= -A_n \mathcal{T} (\nabla_n \Theta + \hat{\mathbf{z}} \mathbf{S} \cdot \nabla_n \Theta) \cdot \nabla p \\ &= -A_n \mathcal{T} \nabla_n \Theta \cdot \nabla_n p. \end{aligned} \quad (39.143)$$

Thus, thermobaricity can either increase or decrease density. However, for the bulk of the ocean, thermobaricity tends to increase density, just as cabbeling.

The tendency for the non-Boussinesq steric effect from thermobaricity in an ocean climate model is determined according to

$$\left(\frac{\partial \eta}{\partial t} \right)_{\text{thermob}} = - \int_{-H}^{\eta} dz \mathcal{T} A_n \nabla_n p \cdot \nabla_n \Theta. \quad (39.144)$$

39.7 Parameterized quasi-Stokes transport and global mean sea level

We now consider how the parameterization of quasi-Stokes transport impacts on the non-Boussinesq steric effect with a focus on its contribution to global mean sea level.

39.7.1 Formulation with buoyancy impacted by quasi-Stokes transport

To start, we make use of the non-divergent condition $\nabla \cdot (\rho \mathbf{v}^*) = 0$ (see equation (39.30)) satisfied by the parameterized eddy-induced velocity \mathbf{v}^* to write the contribution to the non-Boussinesq steric term from

quasi-Stokes advection

$$\begin{aligned}
 -\left(\frac{1}{\rho} \frac{d\rho}{dt}\right)_{\text{quasi-Stokes}} &= \left(\alpha \frac{d\Theta}{dt} - \beta \frac{dS}{dt}\right)_{\text{quasi-Stokes}} \\
 &= \frac{1}{\rho} (-\alpha \nabla \cdot (\rho \mathbf{v}^* \Theta) + \beta \nabla \cdot (\rho \mathbf{v}^* S)) \\
 &= \mathbf{v}^* \cdot (-\alpha \nabla \Theta + \beta \nabla S),
 \end{aligned} \tag{39.145}$$

with the final expression in the form of an advection of buoyancy by the quasi-Stokes velocity. We now introduce the vector streamfunction for the quasi-Stokes mass transport

$$\begin{aligned}
 \rho \mathbf{v}^* &= \nabla \wedge \rho \Psi \\
 &= \partial_z (\rho \Upsilon) - \hat{\mathbf{z}} \nabla_z \cdot (\rho \Upsilon),
 \end{aligned} \tag{39.146}$$

where $\Psi = \Upsilon \wedge \hat{\mathbf{z}}$ is the quasi-Stokes vector streamfunction, and Υ is the eddy induced transport (see Section 39.1.3.2), which then leads to

$$-\left(\frac{1}{\rho} \frac{d\rho}{dt}\right)_{\text{quasi-Stokes}} = \partial_z (\rho \Upsilon) \cdot \left(\frac{-\alpha \nabla_z \Theta + \beta \nabla_z S}{\rho} \right) - \nabla_z \cdot (\rho \Upsilon) \left(\frac{-\alpha \partial_z \Theta + \beta \partial_z S}{\rho} \right). \tag{39.147}$$

We specialize this result to the case of a stable vertical stratification (i.e., the buoyancy frequency $N^2 > 0$), in which case

$$-\left(\frac{1}{\rho} \frac{d\rho}{dt}\right)_{\text{quasi-Stokes}} = \left(\frac{N^2}{\rho g} \right) \nabla_n \cdot (\rho \Upsilon), \tag{39.148}$$

where $\nabla_n = \nabla_z + \mathbf{S} \partial_z$ is the projected neutral gradient operator introduced by equation (39.132), N^2 is the squared buoyancy frequency given by equation, and \mathbf{S} is the neutral slope given by equation (39.133). The special case of the Gent et al. (1995) parameterization is revealing and most pertinent given its ubiquitous use in ocean climate modelling to parameterize quasi-Stokes transport from mesoscale eddies. Here, we set the quasi-Stokes transport to

$$\Upsilon^{\text{gm}} = -A_{\text{gm}} \mathbf{S} = -A_{\text{gm}} \nabla_n z, \tag{39.149}$$

where $A_{\text{gm}} > 0$ is a diffusivity, and

$$\mathbf{S} = \nabla_n z \tag{39.150}$$

is an alternative expression for the neutral slope, written here in terms of the projected lateral gradient of the depth of the neutral tangent plane (e.g., equation (6.6) of Griffies, 2004). These expressions for the parameterized quasi-Stokes transport yield

$$-\left(\frac{1}{\rho} \frac{d\rho}{dt}\right)_{\text{gm}} = -\left(\frac{N^2}{\rho g} \right) \nabla_n \cdot (\rho A_{\text{gm}} \nabla_n z). \tag{39.151}$$

The operator on the right hand side represents a neutral diffusion of the depth of a neutral tangent plane. We can make this correspondence precise through introducing the inverse infinitesimal thickness between two neutral tangent planes¹

$$h = dz = \left(\frac{\rho N^2}{g} \right)^{-1} d\gamma, \tag{39.152}$$

where $d\gamma$ is the density increment between the two tangent planes. This substitution leads to

$$-\left(\frac{1}{\rho} \frac{d\rho}{dt}\right)_{\text{gm}} = -\left(\frac{1}{\rho^2 h g} \right) \nabla_n \cdot (N^2 A_{\text{gm}} \rho^2 h \nabla_n z). \tag{39.153}$$

When surfaces of constant buoyancy bow downwards, as in a warm core eddy (see Figure 39.3), this configuration represents a local minimum in the height of the buoyancy surface (or maximum in the depth).

¹For example, see equation (9.70) in Griffies (2004) for the isopycnal diffusion operator in isopycnal coordinates.

The curvature of this surface is negative, so that the diffusion operator (39.151) is positive. The Gent et al. (1995) scheme acts to dissipate the negative curvature by transporting light water away from the anomalously light region, thus raising the depth maxima. That is, heat is transported away from a warm core eddy. This physical interpretation of Gent et al. (1995) accords with its implementation in isopycnal coordinate ocean models, where it appears as an isopycnal layer depth diffusion rather than an isopycnal layer thickness diffusion (see Section 11.3.2 of Griffies et al. (2000a)). The common interpretation as a thickness diffusion is *only* valid when the diffusivity is depth independent (see Section 9.5.4 of Griffies, 2004), and the ocean bottom is flat (see Section 11.3.2 of Griffies et al. (2000a)).

39.7.2 Effects on global mean sea level

So how does diffusion of the depth of a neutral tangent plane contribute to the global mean sea level through the non-Boussinesq steric effect? This question is answered by taking the vertical integral of the GM effect in equation (39.151) to find

$$\left(\frac{\partial \eta}{\partial t}\right)_{\text{gm}} = - \int_{-H}^{\eta} \left(\frac{N^2}{\rho g}\right) \nabla_n \cdot (\rho A_{\text{gm}} \nabla_n z) dz. \quad (39.154)$$

Hence, the parameterized quasi-Stokes transport of buoyancy acts to erode depth maxima by raising buoyancy surfaces, which in turn renders a negative tendency to global mean sea level through the non-Boussinesq steric effect (see Figure 39.3). Conversely, the global mean sea level is raised on the flanks of sea level maxima.

The behaviour exhibited in Figure 39.3 reflects the compensation between downward bowing buoyancy surfaces and upward bowing sea level in a baroclinic ocean respecting hydrostatic balance (e.g., Tomczak and Godfrey, 1994). Figure 39.3 depicts the effect of Gent et al. (1995) for a 1.5 layer ocean.

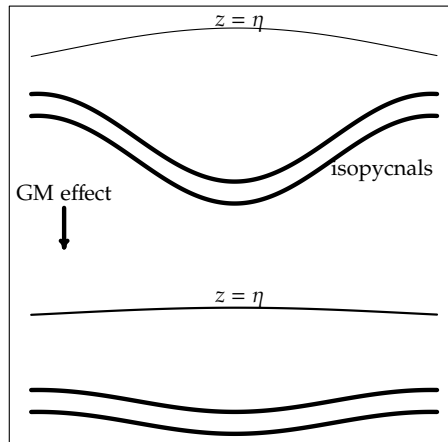


Figure 39.3: Schematic to illustrate the impact of the Gent et al. (1995) parameterization on isopycnals (dark curves) and sea level (through the non-Boussinesq steric effect), assuming the idealized case of a 1.5 layer ocean. Note that the slope of the pycnocline is about 100-300 times larger than the sea level (Rule 1a of Tomczak and Godfrey, 1994). As the Gent et al. (1995) scheme shoals the depth of the depressed pycnocline (moving from the top portion of the panel to the lower portion), it also lowers sea level. The opposite follows for the case of a raised pycnocline, in which the depressed sea level is raised as the pycnocline is lowered. In particular, as warm water is transported away from the core of a warm eddy, sea level is lowered through the non-Boussinesq steric effect.

39.8 MOM sea level diagnostics: Version I

Here we summarize diagnostics related to Version I of the kinematic sea level equation as detailed in Section 39.2.1. As discussed in Griffies and Greatbatch (2012), these diagnostics are most useful in assessing how various terms contribute to the evolution of global mean sea level.

39.8.1 Surface buoyancy fluxes

Each of the following diagnostics are computed inside the module

ocean_core/ocean_sbc.F90

Each term has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- eta_tend_sw: This diagnostic measures the contribution due to the net surface shortwave radiation

$$\text{eta_tend_sw} = \left(\frac{\alpha Q_{\text{sw}}}{c_p \rho} \right)_{k=1}, \quad (39.155)$$

where Q_{sw} is the shortwave heat flux crossing through the ocean surface. The global area mean of eta_tend_sw is saved in eta_tend_sw_glob.

- eta_tend_lw: This diagnostic measures the contribution due to the net surface longwave radiation

$$\text{eta_tend_lw} = \left(\frac{\alpha Q_{\text{lw}}}{c_p \rho} \right)_{k=1}, \quad (39.156)$$

where Q_{lw} is the longwave heat flux crossing through the ocean surface. The global area mean of eta_tend_lw is saved in eta_tend_lw_glob.

- eta_tend_sens: This diagnostic measures the contribution due to the net surface sensible heating

$$\text{eta_tend_sens} = \left(\frac{\alpha Q_{\text{sens}}}{c_p \rho} \right)_{k=1}, \quad (39.157)$$

where Q_{sens} is the sensible heat flux crossing through the ocean surface. The global area mean of eta_tend_sens is saved in eta_tend_sens_glob.

- eta_tend_evap_heat: This diagnostic measures the contribution due to the net surface latent heat of vaporization acting to cool the upper ocean

$$\text{eta_tend_evap_heat} = \left(\frac{\alpha Q_{\text{vaporization}}}{c_p \rho} \right)_{k=1}, \quad (39.158)$$

where $Q_{\text{vaporization}}$ is the latent heat of vaporization impacting the surface ocean. The global area mean of eta_tend_evap_heat is saved in eta_tend_evap_heat_glob.

- eta_tend_fprec_melt: This diagnostic measures the contribution due to the net latent heat of fusion from frozen precipitation acting to cool the upper ocean

$$\text{eta_tend_fprec_melt} = \left(\frac{\alpha Q_{\text{fprec fusion}}}{c_p \rho} \right)_{k=1}, \quad (39.159)$$

where $Q_{\text{fprec fusion}}$ is the latent heat of fusion from frozen precipitation impacting the surface ocean. The global area mean of eta_tend_fprec_melt is saved in eta_tend_fprec_melt_glob.

- `eta_tend_iceberg_melt`: This diagnostic measures the contribution due to the net latent heat of fusion from solid land ice or icebergs that act to cool the upper ocean

$$\text{eta_tend_iceberg_melt} = \left(\frac{\alpha Q_{\text{iceberg fusion}}}{c_p \rho} \right)_{k=1}, \quad (39.160)$$

where $Q_{\text{iceberg fusion}}$ is the latent heat of fusion from icebergs that impacting the surface ocean. The global area mean of `eta_tend_iceberg_melt` is saved in `eta_tend_fprec_iceberg_melt_glob`.

- `eta_tend_heat_coupler`: This diagnostic measures the contribution due to the total surface heating on the ocean surface

$$\text{eta_tend_heat_coupler} = \left(\frac{\alpha Q_{\text{coupler heat}}}{c_p \rho} \right)_{k=1}, \quad (39.161)$$

where $Q_{\text{coupler heat}}$ is the surface ocean heat that enters through the coupler. This heat includes short-wave, longwave, sensible, and latent heating, so that we have the identity

$$\begin{aligned} \text{eta_tend_heat_coupler} = & \text{eta_tend_sw} + \text{eta_tend_lw} + \text{eta_tend_sens} \\ & + \text{eta_tend_evap_heat} + \text{eta_tend_fprec_melt} + \text{eta_tend_iceberg_melt}. \end{aligned} \quad (39.162)$$

The global area mean of `eta_tend_heat_coupler` is saved in `eta_tend_fprec_heat_coupler_glob`.

- `eta_tend_heat_restore`: This diagnostic measures the contribution due to the heat input to the ocean through surface restoring, as may occur in idealized simulations

$$\text{eta_tend_heat_restore} = \left(\frac{\alpha Q_{\text{heat restoring}}}{c_p \rho} \right)_{k=1}, \quad (39.163)$$

where $Q_{\text{heat restoring}}$ is the heat input through surface restoring. The global area mean of `eta_tend_heat_restore` is saved in `eta_tend_fprec_heat_restore_glob`.

- `eta_tend_salt_coupler`: This diagnostic measures the contribution due to the salt input to the ocean through the coupler, with such salt generally limited to regions of sea ice (see Section 39.3.4.2)

$$\text{eta_tend_salt_coupler} = - \left(\frac{\beta Q_{\text{salt coupler}}}{c_p \rho} \right)_{k=1}, \quad (39.164)$$

where $Q_{\text{salt coupler}}$ is the salt flux entering through the surface ocean. The global area mean of `eta_tend_salt_coupler` is saved in `eta_tend_fprec_salt_coupler_glob`.

- `eta_tend_salt_restore`: This diagnostic measures the contribution due to the salt input to the ocean through surface restoring, as may occur in idealized simulations

$$\text{eta_tend_salt_restore} = - \left(\frac{\beta Q_{\text{salt restoring}}}{c_p \rho} \right)_{k=1}, \quad (39.165)$$

where $Q_{\text{salt restoring}}$ is the salt input through surface restoring. The global area mean of `eta_tend_salt_restore` is saved in `eta_tend_fprec_salt_restore_glob`.

39.8.2 Surface mass fluxes

Each of the following diagnostics are computed inside the module

`ocean_core/ocean_sbc.F90`

Each term has units of m s^{-1} .

- `eta_tend_evap`: This diagnostic measures the contribution due to the mass of water exchanged due to evaporation or condensation at the ocean surface

$$\text{eta_tend_evap} = \left(\frac{Q_{\text{evaporation}}}{\rho} \right)_{k=1}, \quad (39.166)$$

where $Q_{\text{evaporation}}$ is the mass exchanged due to evaporation or condensation. The global area mean of `eta_tend_evap` is saved in `eta_tend_evap_glob`.

- `eta_tend_lprec`: This diagnostic measures the contribution due to the mass of water exchanged due to liquid precipitation at the ocean surface

$$\text{eta_tend_lprec} = \left(\frac{Q_{\text{liquid precip}}}{\rho} \right)_{k=1}, \quad (39.167)$$

where $Q_{\text{liquid precip}}$ is the mass exchanged due to liquid precipitation. The global area mean of `eta_tend_lprec` is saved in `eta_tend_lprec_glob`.

- `eta_tend_fprec`: This diagnostic measures the contribution due to the mass of water exchanged due to frozen precipitation at the ocean surface

$$\text{eta_tend_fprec} = \left(\frac{Q_{\text{frozen precip}}}{\rho} \right)_{k=1}, \quad (39.168)$$

where $Q_{\text{frozen precip}}$ is the mass exchanged due to frozen precipitation (i.e., snow). The global area mean of `eta_tend_fprec` is saved in `eta_tend_fprec_glob`.

- `eta_tend_runoff`: This diagnostic measures the contribution due to the mass of water exchanged due to liquid river runoff at the ocean surface

$$\text{eta_tend_runoff} = \left(\frac{Q_{\text{liquid runoff}}}{\rho} \right)_{k=1}, \quad (39.169)$$

where $Q_{\text{liquid runoff}}$ is the mass exchanged due to liquid river runoff. The global area mean of `eta_tend_runoff` is saved in `eta_tend_runoff_glob`.

- `eta_tend_iceberg`: This diagnostic measures the contribution due to the mass of water exchanged due to solid river runoff (i.e., icebergs) at the ocean surface

$$\text{eta_tend_iceberg} = \left(\frac{Q_{\text{iceberg}}}{\rho} \right)_{k=1}, \quad (39.170)$$

where Q_{iceberg} is the mass exchanged due to solid river runoff. The global area mean of `eta_tend_iceberg` is saved in `eta_tend_iceberg_glob`.

- `eta_tend_water_coupler`: This diagnostic measures the net contribution due to the mass of water exchanged through the coupler at the ocean surface

$$\text{eta_tend_water_coupler} = \left(\frac{Q_{\text{water coupler}}}{\rho} \right)_{k=1}, \quad (39.171)$$

where $Q_{\text{water coupler}}$ is the mass exchanged due to all water crossing the ocean surface as mediated by the coupler. This tendency is equal to

$$\begin{aligned} \text{eta_tend_water_coupler} = & \text{eta_tend_evap} + \text{eta_tend_lprec} + \text{eta_tend_fprec} \\ & + \text{eta_tend_runoff} + \text{eta_tend_iceberg}. \end{aligned} \quad (39.172)$$

The global area mean of `eta_tend_water_coupler` is saved in `eta_tend_water_coupler_glob`.

39.8.3 Bottom heat flux

The following diagnostic is computed inside the module

ocean_core/ocean_bbc.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_geoheat`: This diagnostic measures the contribution due to the net bottom geothermal heating

$$\text{eta_tend_geoheat} = \left(\frac{\alpha Q_{\text{geoheat}}}{c_p \rho} \right)_{k=\text{kmt}}, \quad (39.173)$$

where Q_{geoheat} is the geothermal heat flux crossing through the ocean bottom, and $k=\text{kmt}$ is the bottom grid cell. The global area mean of `eta_tend_geoheat` is saved in `eta_tend_geoheat_glob`.

39.8.4 River insertion of liquid and solid water

The following diagnostic is computed inside the module

ocean_param/sources/ocean_rivermix.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_rivermix`: This diagnostic measures the contribution due to the insertion of river water (solid plus liquid) into the ocean (see Chapter 28)

$$\text{eta_tend_rivermix} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_rivermix} \right], \quad (39.174)$$

where `neut_rho_rivermix` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.10.7. The global area mean of `eta_tend_rivermix` is saved in `eta_tend_rivermix_glob`.

39.8.5 River insertion of liquid water

The following diagnostic is computed inside the module

ocean_param/sources/ocean_rivermix.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_runoffmix`: This diagnostic measures the contribution due to the insertion of river water (solid plus liquid) into the ocean (see Chapter 28)

$$\text{eta_tend_runoffmix} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_runoffmix} \right], \quad (39.175)$$

where `neut_rho_runoffmix` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.10.9. The global area mean of `eta_tend_runoffmix` is saved in `eta_tend_runoffmix_glob`.

39.8.6 River insertion of solid water

The following diagnostic is computed inside the module

ocean_param/sources/ocean_rivermix.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_calvingmix`: This diagnostic measures the contribution due to the insertion of river water (solid plus liquid) into the ocean (see Chapter 28)

$$\text{eta_tend_calvingmix} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_calvingmix} \right], \quad (39.176)$$

where `neut_rho_calvingmix` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.10.11. The global area mean of `eta_tend_calvingmix` is saved in `eta_tend_calvingmix_glob`.

39.8.7 Heating of liquid ocean due to frazil formation

The following diagnostic is computed inside the module

ocean_tracers/ocean_tracer.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_frazil`: This diagnostic measures the contribution due to the heating of liquid water due to frazil formation (see Section 39.3.4.1)

$$\text{eta_tend_frazil} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_frazil} \right], \quad (39.177)$$

where `neut_rho_frazil` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.11.7.1. The global area mean of `eta_tend_frazil` is saved in `eta_tend_frazil_glob`.

39.8.8 Motion across pressure surfaces

The following diagnostic is computed inside the module

ocean_core/ocean_advection_velocity.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_press`: This diagnostic measures the contribution due to the motion across pressure surfaces (see Section 39.3.2 and equation (39.67))

$$\text{eta_tend_press} = g(H + \eta) \left(\overline{\left(\frac{w^{(p)}}{c_{\text{sound}}^2} \right)^2} \right)^{1/2}. \quad (39.178)$$

The global area mean of `eta_tend_press` is saved in `eta_tend_press_glob`. Note that even if not using pressure as the vertical coordinate, the vertical velocity component across pressure surfaces, $w^{(p)}$, is very well approximated by the vertical velocity component across geopotential, z^* , or p^* surfaces.

39.8.9 Mixing associated with vertical diffusion

The following diagnostic is computed inside the module

ocean_param/mixing/ocean_vert_mix.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

There are two forms computed for this diagnostic. The first is based on the integration by parts manipulations considered in Section 39.3.3. Here, we expose the product between flux components and gradients of α/ρ and β/ρ , which then leads to the interior source term given by the third term in equation (39.89). The second approach is based directly on equation (39.72) prior to any integration by parts manipulations. Both approaches are equivalent in the continuum for processes those that have zero boundary fluxes. The second approach is simpler to diagnose, since it works directly on tendencies rather than requiring the product of flux components with the gradients of α/ρ and β/ρ . However, the first form is more directly related to physical interpretations, as it exposes flux components.

- `eta_tend_diff_cbt_flux`: This diagnostic measures the contribution due to vertical diffusion (see Section 39.5 and equation (39.119)) just associated with the dianeutral diffusivities `diff_cbt_t` and `diff_cbt_s`

$$\text{eta_tend_diff_cbt_flux} = - \int_{-H}^{\eta} dz \rho \left[D^{\Theta} \partial_z \Theta \partial_z (\alpha/\rho) - D^S \partial_z S \partial_z (\beta/\rho) \right]. \quad (39.179)$$

The global area mean of `eta_tend_diff_cbt_flux` is saved in `eta_tend_diff_cbt_flux_glob`.

- `eta_tend_diff_cbt_tend`: This diagnostic measures the contribution due to vertical diffusion just arising from the dianeutral diffusivity `diff_cbt_t` and `diff_cbt_s` (see Section 39.5 prior to exposing the flux components)

$$\text{eta_tend_diff_cbt_tend} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_diff_cbt} \right], \quad (39.180)$$

where `neut_rho_diff_cbt` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.11.3.2. The global area mean of `eta_tend_diff_cbt_tend` is saved in `eta_tend_diff_cbt_tend_glob`.

39.8.10 Mixing associated with neutral diffusion

The following diagnostic is computed inside the module

ocean_param/neutral/ocean_nphysics_util.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

There are two general approaches to this diagnostic:

- **NUMERICAL METHOD**: This approach computes the contribution to sea level tendency associated with the full neutral diffusion operator, including the portion within the upper ocean boundary layer where neutral diffusion exponentially transforms into horizontal diffusion as per the recommendations from Treguier et al. (1997), Ferrari et al. (2008), and Ferrari et al. (2010).

Within this method there are also two approaches:

- The first is based on the integration by parts manipulations considered in Section 39.3.3. Here, we expose the product between flux components and gradients of α/ρ and β/ρ , which then leads to the interior source term given by the third term in equation (39.89).

- The second approach is based directly on equation (39.72) prior to any integration by parts manipulations.

Both approaches are equivalent in the continuum for processes such as neutral diffusion that have zero boundary fluxes. The second approach is simpler to diagnose, since it works directly on tendencies rather than requiring the product of flux components with the gradients of α/ρ and β/ρ . However, the first form is more directly related to physical interpretations, as it exposes flux components.

- **ANALYTICAL METHOD:** This approach computes the contribution to cabbeling and thermobaricity according to the analytical manipulations detailed in Sections 39.6.6 and 39.6.7.

An approximate means to isolate the effects of horizontal diffusion is found by subtracting the cabbeling and thermobaricity pieces from the full neutral diffusion operator.

- **eta_tend_ndiff_flg:** This diagnostic measures the contribution due to the neutral diffusion operator (see Section 39.6), including all contributions to the horizontal and vertical flux components

$$\text{eta_tend_ndiff_flg} = \int_{-H}^{\eta} dz \left[\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) \right], \quad (39.181)$$

where the three-dimensional temperature and salinity fluxes are computed according to the neutral diffusion operator. Note that we incorporate all pieces of the flux components, including the implicit-in-time piece of the vertical flux. The global area mean of `eta_tend_ndiff_flg` is saved in `eta_tend_ndiff_flg_glob`. This diagnostic has been implemented only for the

`ocean_param/nphysics/ocean_nphysicsC.F90`

module.

- **eta_tend_ndiff_tend:** This diagnostic measures the contribution due to neutral diffusion prior to exposing the flux components, and without including the time-implicit portion of the K33 term

$$\text{eta_tend_ndiff_tend} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_ndiff} \right], \quad (39.182)$$

where `neut_rho_ndiff` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.11.4.1. The global area mean of `eta_tend_ndiff_tend` is saved in `eta_tend_ndiff_tend_glob`.

- **eta_tend_k33_tend:** This diagnostic measures the contribution due to time-implicit portion of the K33 term from neutral diffusion

$$\text{eta_tend_k33_tend} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_k33} \right], \quad (39.183)$$

where `neut_rho_k33` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.11.3.7. The global area mean of `eta_tend_k33_tend` is saved in `eta_tend_k33_tend_glob`. Combining with the diagnostic `eta_tend_ndiff_tend` yields an approximation to the full neutral diffusion operator. Hence, when globally integrated, this result should approximate that from the diagnostic `eta_tend_ndiff_flg_glob`:

$$\text{eta_tend_ndiff_tend_glob} + \text{eta_tend_k33_tend_glob} \approx \text{eta_tend_ndiff_flg_glob}. \quad (39.184)$$

- **cabbeling_tend_intz:** This diagnostic measures the contribution due to cabbeling (see Section 39.6.6 and equation (39.140))

$$\text{cabbeling_tend_intz} = \int_{-H}^{\eta} dz \mathcal{C} A_n (\nabla_n \Theta)^2. \quad (39.185)$$

The global area mean of `cabbeling_tend_intz` is saved in `cabbeling_tend_intz_glob`. To compute the contribution to sea level requires a minus sign

$$\text{sea level tendency from cabbeling} = -\text{cabbeling_tend_intz}. \quad (39.186)$$

- `thermobaric_tend_intz`: This diagnostic measures the contribution due to thermobaricity (see Section 39.6.7 and equation (39.144))

$$\text{thermobaric_tend_intz} = \int_{-H}^{\eta} dz \, T \, A_n \nabla_n p \cdot \nabla_n \Theta. \quad (39.187)$$

The global area mean of `thermobaric_tend_intz` is saved in `thermobaric_tend_intz_glob`. To compute the contribution to sea level requires a minus sign

$$\text{sea level tendency from thermobaricity} = -\text{thermobaric_tend_intz}. \quad (39.188)$$

39.8.11 Parameterized eddy advection from GM

The following diagnostic is computed inside the module

`ocean_param/neutral/ocean_nphysics_util.F90`

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43). There are two general approaches to this diagnostic:

- **NUMERICAL METHOD:** This approach computes the contribution to sea level tendency associated with the full mesoscale eddy induced quasi-Stokes transport operator. Within this method there are also two approaches:
 - The first is based on the integration by parts manipulations considered in Section 39.3.3. Here, we expose the product between flux components and gradients of α/ρ and β/ρ , which then leads to the interior source term given by the third term in equation (39.89).
 - The second approach is based directly on equation (39.72) prior to any integration by parts manipulations.

Both approaches are equivalent in the continuum for processes those that have zero boundary fluxes. The second approach is simpler to diagnose, since it works directly on tendencies rather than requiring the product of flux components with the gradients of α/ρ and β/ρ . However, the first form is more directly related to physical interpretations, as it exposes flux components.

- **ANALYTICAL METHOD:** This approach computes the contribution to cabbeling and thermobaricity according to the analytical manipulations detailed in Section 39.7.
- `eta_tend_gm_flux`: This diagnostic measures the contribution due to the eddy-induced transport from either Gent et al. (1995) or Ferrari et al. (2010) (see Section 39.7)

$$\text{eta_tend_gm_flux} = \int_{-H}^{\eta} dz \left[\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) \right], \quad (39.189)$$

where the three-dimensional temperature and salinity fluxes are computed according to the quasi-Stokes transport operator. The global area mean of `eta_tend_gm_flux` is saved in `eta_tend_gm_flux_glob`. This diagnostic has been implemented only for the

`ocean_param/nphysics/ocean_nphysicsC.F90`

module.

- `eta_tend_gm_tend`: This diagnostic measures the contribution due to parameterized quasi-Stokes transport (see Section 39.7) prior to exposing the flux components

$$\text{eta_tend_gm_tend} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_gm} \right], \quad (39.190)$$

where `neut_rho_gm` is the time tendency for locally reference potential density, with this diagnostic detailed in Section 36.10.3. The global area mean of `eta_tend_gm_tend` is saved in `eta_tend_gm_tend_glob`.

- `eta_tend_gm90`: This diagnostic measures the contribution due to the Gent and McWilliams (1990) parameterization of eddy advection (see Section 39.7 and equation (39.154)), which is written in the following analytic form

$$\text{eta_tend_gm90} = - \int_{-H}^{\eta} \left(\frac{N^2}{\rho g} \right) \nabla_n \cdot (\rho A_{gm} \nabla_n z) dz. \quad (39.191)$$

The global area mean of `eta_tend_gm90` is saved in `eta_tend_gm90_glob`. As this diagnostic computes the contribution based on an analytic form, it is generally not as precise as the diagnostic computed from the actual numerical implementation. For this reason, we recommend `eta_tend_gm_flux` for more precise calculations required for sea level budgets.

39.8.12 Parameterized eddy advection from submesoscale parameterization

The following diagnostic is computed inside the module

`ocean_param/mixing/ocean_submesoscale.F90`

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

There are two forms computed for this diagnostic. The first is based on the integration by parts manipulations considered in Section 39.3.3. Here, we expose the product between flux components and gradients of α/ρ and β/ρ , which then leads to the interior source term given by the third term in equation (39.89). The second approach is based directly on equation (39.72) prior to any integration by parts manipulations. Both approaches are equivalent in the continuum for processes those that have zero boundary fluxes. The second approach is simpler to diagnose, since it works directly on tendencies rather than requiring the product of flux components with the gradients of α/ρ and β/ρ . However, the first form is more directly related to physical interpretations, as it exposes flux components.

- `eta_tend_submeso_flux`: This diagnostic measures the contribution due to the Fox-Kemper et al. (2008b) submesoscale eddy parameterization

$$\text{eta_tend_submeso_flux} = \int_{-H}^{\eta} dz \left[\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) \right], \quad (39.192)$$

where the three-dimensional temperature and salinity fluxes are computed according to the quasi-Stokes transport operator according to the Fox-Kemper et al. (2008b) scheme. The global area mean of `eta_tend_submeso_flux` is saved in `eta_tend_submeso_flux_glob`.

- `eta_tend_submeso_tend`: This diagnostic measures the contribution due to parameterized quasi-Stokes transport from submesoscale processes according to Fox-Kemper et al. (2008b), with this diagnostic computed prior to exposing the flux components

$$\text{eta_tend_submeso_tend} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_submeso} \right], \quad (39.193)$$

where `neut_rho_submeso` is the time tendency for locally referenced potential density, with this diagnostic detailed in Section 36.10.4. The global area mean of `eta_tend_submeso_tend` is saved in `eta_tend_submeso_tend_glob`.

39.8.13 Parameterized horizontal diffusion from submesoscale parameterization

The following diagnostic is computed inside the module

ocean_param/mixing/ocean_submesoscale.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

There are two forms computed for this diagnostic. The first is based on the integration by parts manipulations considered in Section 39.3.3. Here, we expose the product between flux components and gradients of α/ρ and β/ρ , which then leads to the interior source term given by the third term in equation (39.89). The second approach is based directly on equation (39.72) prior to any integration by parts manipulations. Both approaches are equivalent in the continuum for processes those that have zero boundary fluxes. The second approach is simpler to diagnose, since it works directly on tendencies rather than requiring the product of flux components with the gradients of α/ρ and β/ρ . However, the first form is more directly related to physical interpretations, as it exposes flux components.

- `eta_tend_subdiff_flg`: This diagnostic measures the contribution due to the horizontal diffusive aspects motivated by the [Fox-Kemper et al. \(2008b\)](#) submesoscale eddy parameterization (see Section 24.6)

$$\text{eta_tend_subdiff_flg} = \int_{-H}^{\eta} dz \left[\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) \right], \quad (39.194)$$

where the two-dimensional horizontal temperature and salinity diffusive fluxes are computed according to Section 24.6. The global area mean of `eta_tend_subdiff_flg` is saved in `eta_tend_subdiff_flg_glob`.

- `eta_tend_subdiff_tend`: This diagnostic measures the contribution due to the horizontal diffusive aspects motivated by the [Fox-Kemper et al. \(2008b\)](#) submesoscale eddy parameterization (see Section 24.6), with this diagnostic computed prior to exposing the flux components

$$\text{eta_tend_subdiff_tend} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_subdiff} \right], \quad (39.195)$$

where `neut_rho_subdiff` is the time tendency for locally referenced potential density. The global area mean of `eta_tend_submeso_tend` is saved in `eta_tend_submeso_tend_glob`.

39.8.14 Penetrative shortwave radiation

The following diagnostic is computed inside the module

ocean_param/sources/ocean_shortwave.F90

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_sw_pen`: This diagnostic measures the contribution due to the penetrative shortwave radiation (see Chapter 17).

$$\text{eta_tend_sw_pen} = \int_{-H}^{\eta} dz \left[\mathbf{J}^{(\Theta)} \cdot \nabla(\alpha/\rho) - \mathbf{J}^{(S)} \cdot \nabla(\beta/\rho) \right], \quad (39.196)$$

where the three-dimensional temperature and salinity fluxes are computed according to the shortwave penetration scheme. All components of the salt flux are zero, as are the horizontal temperature flux components. The vertical temperature flux component is diagnosed given the temperature

tendency from the shortwave scheme, and assuming the surface shortwave flux component is computed from the surface flux module and thus contained in `eta_tend_sw` (see Section 39.8.1). That is, we compute

$$J^{(\theta)}(k) = J^{(\theta)}(k-1) + \text{th_tendency}(k), \quad (39.197)$$

where $J^{(\theta)}(k=0) = 0$, and `th_tendency(k)` is the thickness and density weighted temperature tendency from the penetrative shortwave scheme. The global area mean of `eta_tend_sw_pen` is saved in `eta_tend_sw_pen_glob`.

39.8.15 Sigma transport

The following diagnostic is computed inside the module

`ocean_param/mixing/ocean_sigma_transport.F90`

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_sigma`: This diagnostic measures the contribution from sigma transport scheme. Since this scheme is not implemented as a traditional flux-form, we diagnose its contribution just based on its tendency

$$\text{eta_tend_sigma} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_sigma} \right], \quad (39.198)$$

where `neut_rho_sigma` is the time tendency for locally referenced potential density, with this diagnostic detailed in Section 36.11.6.5. The global area mean of `eta_tend_sigma` is saved in `eta_tend_sigma_glob`.

39.8.16 Mixdownslope

The following diagnostic is computed inside the module

`ocean_param/mixing/ocean_mixdownslope.F90`

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_mixdown`: This diagnostic measures the contribution from mixdown transport scheme. Since this scheme is not implemented as a traditional flux-form, we diagnose its contribution just based on its tendency

$$\text{eta_tend_mixdown} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_mixdown} \right], \quad (39.199)$$

where `neut_rho_mixdown` is the time tendency for locally referenced potential density, with this diagnostic detailed in Section 36.11.6.4. The global area mean of `eta_tend_mixdown` is saved in `eta_tend_mixdown_glob`.

39.8.17 KPP nonlocal mixing

The following diagnostic is computed inside the module

`ocean_param/mixing/ocean_vert_kpp.F90`

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations (39.42) and (39.43).

- `eta_tend_kpp_nloc`: This diagnostic measures the contribution from nonlocal KPP transport scheme [Large et al. \(1994\)](#). Since this scheme is not implemented as a traditional flux-form, we diagnose its contribution just based on its tendency

$$\text{eta_tend_kpp_nloc} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_kpp_nloc} \right], \quad (39.200)$$

where `neut_rho_kpp_nloc` is the time tendency for locally referenced potential density, with this diagnostic detailed in Section [36.11.3.10](#). The global area mean of `eta_tend_kpp_nloc` is saved in `eta_tend_kpp_nloc_glob`.

39.8.18 Cross land mixing

The following diagnostic is computed inside the module

`ocean_param/sources/ocean_xlandmix.F90`

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations [\(39.42\)](#) and [\(39.43\)](#).

- `eta_tend_xmix`: This diagnostic measures the contribution from cross land mixing scheme. Since this scheme is not implemented as a traditional flux-form, we diagnose its contribution just based on its tendency

$$\text{eta_tend_xmix} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_xmix} \right], \quad (39.201)$$

where `neut_rho_xmix` is the time tendency for locally referenced potential density, with this diagnostic detailed in Section [36.11.6.6](#). The global area mean of `eta_tend_xmix` is saved in `eta_tend_xmix_glob`.

39.8.19 Cross land insertion

The following diagnostic is computed inside the module

`ocean_param/sources/ocean_xlandinsert.F90`

It has units of m s^{-1} . Additionally, note that the thermal expansion and haline contraction coefficients are evaluated in MOM according to equations [\(39.42\)](#) and [\(39.43\)](#).

- `eta_tend_xinsert`: This diagnostic measures the contribution from cross land mixing scheme. Since this scheme is not implemented as a traditional flux-form, we diagnose its contribution just based on its tendency

$$\text{eta_tend_xinsert} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_xinsert} \right], \quad (39.202)$$

where `neut_rho_xinsert` is the time tendency for locally referenced potential density, with this diagnostic detailed in Section [36.11.6.7](#). The global area mean of `eta_tend_xinsert` is saved in `eta_tend_xinsert_glob`.

39.8.20 Smoothing of free surface or bottom pressure

- `eta_tend_smooth`: This diagnostic measures the contribution from the smoothing of either the surface height or bottom pressure. Since this scheme is not implemented as a traditional flux-form, we diagnose its contribution just based on its tendency

$$\text{eta_tend_smooth} = - \sum_k \left[\left(\frac{dz}{\rho} \right) \text{neut_rho_smooth} \right], \quad (39.203)$$

where `neut_rho_smooth` is the time tendency for locally referenced potential density, with this diagnostic detailed in Section [36.11.7.2](#). The global area mean of `eta_tend_smooth` is saved in `eta_tend_smooth_glob`.

39.9 MOM sea level diagnostics: Version II

Here we summarize diagnostics related to Version II of the kinematic sea level equation as detailed in Section 39.2.2 (see equation (39.58)). As discussed in Griffies and Greatbatch (2012), these diagnostics are often used in assessing how various terms contribute to the evolution of regional sea level. The diagnostics are approximate when using the volume conserving Boussinesq form of MOM, and more accurate when using the mass conserving non-Boussinesq form. Each diagnostic field is computed inside the module

ocean_core/ocean_barotropic.F90

The tendencies are in units of m s^{-1} and the sealevel has units of meter.

- `eta_nonbouss_tend`: This diagnostic measures the contribution due to the evolution of the mass conserving non-Boussinesq sea level according to equation (39.57)

$$\text{eta_nonbouss_tend} = \frac{Q_m - \nabla \cdot \mathbf{U}^\rho - (H + \eta) \partial \bar{\rho}^z / \partial t}{\bar{\rho}^z} + \text{sources.} \quad (39.204)$$

Contributions from sources are associated with such non-local processes as cross-land mixing and insertion (Chapters 29 and 30). Sources are also accounting for numerically motivated smoothing the free surface, which optionally occurs to suppress the checker-board null mode present for gravity waves on a B-grid (Section 31.1). The global area mean of `eta_nonbouss_tend` is saved in `eta_nonbouss_tend_global`.

- `eta_nonbouss`: This diagnostic time steps the sea level as forced by `eta_nonbouss_tend`. Accordingly, for a Boussinesq model it approximates the non-Boussinesq sea level. For a non-Boussinesq model, `eta_nonbouss` should agree quite closely to the actual `eta_t` field.

$$\text{eta_nonbouss}(\tau + 1) = \text{eta_nonbouss}(\tau) + \Delta\tau \text{eta_nonbouss_tend.} \quad (39.205)$$

The global area mean of `eta_nonbouss` is saved in `eta_nonbouss_global`.

- `eta_dynamic_tend`: This diagnostic measures the contribution from the convergence of the depth integrated mass transport in equation (39.57)

$$\text{eta_dynamic_tend} = - \left(\frac{\nabla \cdot \mathbf{U}^\rho}{\bar{\rho}^z} \right). \quad (39.206)$$

When using the Boussinesq version of MOM, we approximate this term using

$$\text{eta_dynamic_tend} \approx -\nabla \cdot \mathbf{U} \quad \text{Boussinesq.} \quad (39.207)$$

The global area mean of `eta_dynamic_tend` is saved in `eta_dynamic_tend_global`.

- `eta_water_tend`: This diagnostic measures the contribution from the mass fluxes crossing the ocean surface (see equation 39.57))

$$\text{eta_water_tend} = \frac{Q_m}{\bar{\rho}^z}. \quad (39.208)$$

When using the Boussinesq version of MOM, we approximate this diagnostic using

$$\text{eta_water_tend} \approx \frac{Q_m}{\rho_o} \quad \text{Boussinesq.} \quad (39.209)$$

The global area mean of `eta_water_tend` is saved in `eta_water_tend_global`.

- `eta_nonsteric_tend`: This diagnostic measures the contribution due to the terms not associated with the steric effect (see equation (39.57))

$$\text{eta_nonsteric_tend} = \frac{Q_m - \nabla \cdot \mathbf{U} \bar{\rho}}{\bar{\rho}^z} + \text{sources.} \quad (39.210)$$

When using the Boussinesq version of MOM, we approximate this diagnostic using

$$\text{eta_nonsteric_tend} \approx \frac{Q_m}{\rho_o} - \nabla \cdot \mathbf{U} + \text{sources.} \quad \text{Boussinesq.} \quad (39.211)$$

The global area mean of `eta_nonsteric_tend` is saved in `eta_nonsteric_tend_global`.

- `eta_source_tend`: This diagnostic measures the contribution due to the source terms associated with such non-local processes as cross-land mixing and insertion (Chapters 29 and 30). Sources are also accounting for numerically motivated smoothing the free surface, which optionally occurs to suppress the checker-board null mode present for gravity waves on a B-grid (Section 31.1).

$$\text{eta_source_tend} = \text{sources.} \quad (39.212)$$

The global area mean of `eta_source_tend` is saved in `eta_source_tend_global`.

- `eta_steric_tend`: This diagnostic measures the contribution due to the term associated with the local steric effect (see equation (39.57))

$$\text{eta_steric_tend} = - \left(\frac{(H + \eta) \partial \bar{\rho}^z / \partial t}{\bar{\rho}^z} \right). \quad (39.213)$$

The global area mean of `eta_steric_tend` is saved in `eta_steric_tend_global`.

GYRE AND OVERTURNING CONTRIBUTIONS TO TRACER TRANSPORT

Contents

40.1	Formulation	595
40.2	Enabling the diagnostic	596

The purpose of this chapter is to discuss the diagnostic in MOM that computes the contribution to advective tracer transport in the j -direction (generalized meridional) associated with gyre and overturning components of the flow. [Bryan and Lewis \(1979\)](#) were amongst the first to make use of this diagnostic, and [Peixoto and Oort \(1992\)](#) (page 342) describe the decomposition in more detail. The gyre component has traditionally been associated with wind driven processes, and overturning associated with thermohaline processes. However, this physical association is not clean, as it is generally not possible to split the transport into two distinct physical processes. For example, removing wind forcing does not necessarily mean the gyre contribution is zero. Nonetheless, this decomposition is commonly performed, and such has provided some physical insight into the flow characteristics.

Note that in the present implementation, only the resolved advective transport is included in the gyre and overturning diagnostic. A more complete implementation should include eddy induced transport associated with mesoscale closure of [Gent et al. \(1995\)](#) (Chapter 23) and the submesoscale scheme of [Fox-Kemper et al. \(2008b\)](#) (Chapter 24).

The following MOM module is directly connected to the material in this chapter:

ocean_tracers/ocean_tracer_advect.F90

40.1 Formulation

The meridional advective transport of a tracer within a particular ocean basin is given by the integral

$$\mathcal{H}(y, t) = \int_{x_1}^{x_2} dx \int_{-H}^{\eta} dz \rho C v, \quad (40.1)$$

where v is the meridional velocity component, ρ is the *in situ* density, C is the tracer concentration, $z = -H(x, y)$ is the ocean bottom, $z = \eta(x, y, t)$ is the ocean surface, and x_1 and x_2 are the longitudinal boundaries of the basin or global ocean. With the generalized horizontal coordinates in MOM, the coordinate y is assumed to be aligned with the j grid lines. For the purposes of this diagnostic, we do *not* perform

a rotation of the flow to the geographical longitude and latitude. For the tripolar grid commonly used for global modeling with MOM, generalized zonal integrals along i -lines yield j -transport, and this is sufficient for most diagnostic purposes. Note the term *north* will continue to be used, with this generally denoting the j -direction, which corresponds to the geographical north when j -lines are parallel to longitudes.

In MOM, the density and thickness weighted advective velocity component

$$\mathcal{V} = v \rho \, dz \quad (40.2)$$

is computed and held in the array `vhrho.nt`, where ρ reduces to the constant Boussinesq density ρ_0 for the Boussinesq version of MOM, but it remains the *in situ* density for the nonBoussinesq version. The product $dx \mathcal{V}$ measures the mass per time of seawater leaving the north face of the grid cell, with $dx \mathcal{V} C$ thus measuring the mass per time of tracer leaving the north face. We now consider the following decomposition of this transport by defining the zonal average transport and zonal average tracer concentration as follows

$$[\mathcal{V}] = \frac{\sum_i dx \mathcal{V}}{\sum_i dx} \quad (40.3)$$

$$[C] = \frac{\sum_i dx C}{\sum_i dx}, \quad (40.4)$$

along with the deviations from zonal average

$$\mathcal{V} = [\mathcal{V}] + \mathcal{V}^* \quad (40.5)$$

$$C = [C] + C^*. \quad (40.6)$$

The discrete i -sum extends over the basin or global domain of interest, so that $\sum_i dx \mathcal{V}$ is the total meridional transport of seawater at this zonal band at a particular k -level. The resulting meridional tracer transport becomes

$$\mathcal{H}(y, t) = \sum_i \sum_k dx \mathcal{V} C \quad (40.7)$$

$$= \sum_i \sum_k dx ([\mathcal{V}][C] + \mathcal{V}^* C^*), \quad (40.8)$$

where the k sum extends over the depth of a column.

We identify three components in the code:

$$\text{merid_flux_advect} = \sum_i \sum_k dx \mathcal{V} C \quad (40.9)$$

$$\text{merid_flux_over} = \sum_i \sum_k dx [\mathcal{V}][C] \quad (40.10)$$

$$\text{merid_flux_gyre} = \sum_i \sum_k dx \mathcal{V}^* C^*. \quad (40.11)$$

Note that

$$\text{merid_flux_gyre} = \text{merid_flux_advect} - \text{merid_flux_over}. \quad (40.12)$$

This identity follows very simply when the advective flux takes on the simple form of either first order upwind or second order centered differences. It becomes more involved when considering higher order, or flux limited, advection schemes. Nonetheless, we use this as a definition of the gyre component, so that the advective flux is built from the advection scheme used in the model.

40.2 Enabling the diagnostic

The following is necessary to enable this diagnostic.

- Set any of the following within the diagnostic table:

```
"tracer"_merid_flux_advect_"basin"  
"tracer"_merid_flux_over_"basin"  
"tracer"_merid_flux_gyre_"basin"
```

where "tracer" is the MOM shortname for the tracer (e.g., "temp" or "salt"), and "basin" is one of the names "global", "southern", "atlantic", "pacific", "arctic", or "indian".

- To get the transport diagnostic partitioned into basins in the World Ocean, it is necessary to read in a netCDF basin mask. A sample mask is provided with the CORE test case based on the global one-degree ocean used for CM2.1 ([Griffies et al. \(2005\)](#), [Gnanadesikan et al. \(2006\)](#), [Delworth et al. \(2006\)](#), [Wittenberg et al. \(2006\)](#), and [Stouffer et al. \(2006a\)](#)). The logical to read in this mask is

```
read_basin_mask = .true.
```

The convention is that the basins are defined by the following values in the mask:

```
Southern = 1.0  
Atlantic = 2.0  
Pacific = 3.0  
Arctic = 4.0  
Indian = 5.0.
```


BALANCING THE HYDROLOGICAL CYCLE IN OCEAN-ICE MODELS

Contents

41.1	Transfer of water between sea ice and ocean	599
41.2	Balancing the hydrological cycle	599
41.3	Water mass flux from salt mass flux	600

The purpose of this chapter is to discuss issues of how to balance the hydrological cycle in ocean-ice simulations using MOM and the FMS coupler. In particular, we detail how the liquid water flux transferred between sea ice and liquid ocean is diagnosed in MOM, given the salt mass flux transferred between the media.

The following MOM module is directly connected to the material in this chapter:

ocean_core/ocean_sbc.F90

41.1 Transfer of water between sea ice and ocean

As sea ice forms, water is extracted from the liquid ocean and passed to the solid sea ice. Additionally, there is generally a nonzero salt mass transferred from ocean to ice, thus producing a nonzero salinity for sea ice. Conversely, as sea ice melts, liquid water and salt are added to the ocean.

The GFDL Sea Ice Simulator (SIS) generally simulates sea ice with a nonzero salt content. For example, in the coupled climate model simulations discussed by [Delworth et al. \(2006\)](#), sea ice is assumed to have a salinity

$$S_{ice} = \frac{\text{kg salt in sea ice}}{\text{kg sea ice}} \quad (41.1)$$

of $S_{ice} = 0.005$ (five parts per thousand).

41.2 Balancing the hydrological cycle

When running ocean-ice simulations, we are not explicitly representing land processes, such as rivers, catchment areas, snow accumulation, etc. However, to reduce model drift, it is important to balance the hydrological cycle in ocean-ice models. We thus need to prescribe some form of global normalization to the precipitation minus evaporation plus river runoff. The result of the normalization should be a global integrated zero net water input to the ocean-ice system over a chosen time scale.

How often the normalization is done is a matter of choice. In MOM, we choose to do so at each model time step, so that there is always a zero net input of water to the ocean-ice system. Others choose to normalize over an annual cycle, in which case the net imbalance over an annual cycle is used to alter the subsequent year's water budget in an attempt to damp the annual water imbalance. Note that the annual budget approach may be inappropriate with interannually varying precipitation forcing.

When running ocean-ice coupled models, it is incorrect to include the water transport between the ocean and ice models when aiming to balance the hydrological cycle. The reason is that it is the sum of the water in the ocean plus ice that should be balanced when running ocean-ice models, not the water in any one subcomponent. As an extreme example to illustrate the issue, consider an ocean-ice model with zero initial sea ice. As the ocean-ice model spins up, there *should* be a net accumulation of water in the growing sea ice, and thus a net loss of water from the ocean. The total water contained in the ocean plus ice system is constant, but there is an exchange of water between the subcomponents. This exchange should not be part of the normalization used to balance the hydrological cycle in ocean-ice models.

41.3 Water mass flux from salt mass flux

Balancing the hydrological cycle in an ocean-ice model run with the FMS coupler requires an indirect method. The reason is that melt water from the ice model is added to the precipitation field prior to being added to the ocean model. So prior to balancing the water fluxes, we must remove the ice melt from the precipitation. The namelist option for performing this normalization is

`zero_net_water_coupler`

found in the module

`ocean_core/ocean_sbc_mod.`

An alternative namelist option is

`zero_net_water_couple_restore`

which balances the water due to precipitation minus evaporation plus runoff plus restoring flux water. Note that in MOM4p0, ice melt was *not* removed from precipitation prior to computing the water normalization using `zero_net_water_coupler`. This bug compromised the integrity of the ocean-ice models run with MOM4p0 which used the option `zero_net_water_coupler`.

To obtain the water mass flux between sea ice and ocean, we could consider two approaches. First, we could alter the FMS sea ice code so that it carries explicit information about the water flux in question. This approach requires modifying code that lives outside of MOM, and so is not desirable from a MOM perspective. Alternatively, we can make use of the salt mass flux passed between the ocean and sea ice. This approach is limited to cases where the ice is assumed to have a uniform bulk salinity, as presently assumed in the GFDL sea ice model.

In MOM, we have access to the following salt flux

$$\text{salt_flux_to_ice} = S_{\text{ice}} \rho_{\text{sea ice}} \frac{T_{\text{ice melt}}}{\Delta t}. \quad (41.2)$$

In this equation, $\rho_{\text{sea ice}}$ is the density of sea ice, taken as $\rho_{\text{sea ice}} = 905 \text{ kg m}^{-3}$ in the GFDL ice model, and $T_{\text{ice melt}}/\Delta t$ is the thickness (in meters) of ice that melts over the time step (in seconds) of the ice model. By convention, this salt flux is positive when there is growth in sea ice, thus representing a passage of salt from the liquid ocean into the sea ice. Correspondingly, this flux is positive when there is a mass flux of water to the sea ice. Since we are here working from an ocean perspective, we prefer to measure the flux of salt entering the ocean from the melting sea ice

$$\text{salt_flux_to_ocean} = -\text{salt_flux_to_ice}. \quad (41.3)$$

Dividing the salt flux by the salinity of sea ice yields the mass flux of water that melts (i.e., mass flux of water transferred from sea ice to the ocean)

$$\begin{aligned}
 \text{mass_melt} &= \frac{\text{salt_flux_to_ocean}}{S_{\text{ice}}} \\
 &= \left(\frac{\text{kg ice melt}}{\text{area sea ice} \times T_{\text{ice melt}}} \right) \left(\frac{T_{\text{ice melt}}}{\Delta t} \right) \\
 &= \left(\frac{\text{kg ice melt}}{\text{area sea ice} \times \Delta t} \right).
 \end{aligned} \tag{41.4}$$

This mass of melting sea ice represents a gain of liquid water for the ocean.

Chapter

42

DIAGNOSING THE MOMENTUM BUDGET

Contents

42.1	A split-explicit algorithm to time step momentum	604
42.1.1	Splitting the velocity field	604
42.1.2	Details of the semi-discrete momentum equation	605
42.1.2.1	Split of the pressure gradient force with pressure-based vertical coordinates	607
42.1.2.2	Split of the pressure gradient force with depth-based vertical coordinates	607
42.2	Momentum budget diagnostics	608
42.2.1	Decomposing the momentum budget	608
42.2.2	MOM5 diagnostics for the momentum budget	609
42.2.2.1	Net change in grid cell momentum per area	609
42.2.2.2	Coriolis force per area	609
42.2.2.3	Momentum source/sink	609
42.2.2.4	Momentum advection	609
42.2.2.5	Slow/baroclinic pressure gradient force per area	610
42.2.2.6	Fast/barotropic pressure gradient force per area	610
42.2.2.7	Horizontal viscous friction from Laplacian and biharmonic operators	611
42.2.2.8	Vertical viscous friction, bottom drag, and surface stress	611
42.2.2.9	Summary of the momentum diagnostic terms	612

The purpose of this chapter is to detail the methods required to diagnose the momentum budget in

MOM5. The following MOM modules are directly connected to the material in this chapter:

```
ocean_core/ocean_barotropic.F90
ocean_core/ocean_bbc.F90
ocean_core/ocean_coriolis.F90
ocean_core/ocean_pressure.F90
ocean_core/ocean_sbc.F90
ocean_core/ocean_velocity.F90
ocean_core/ocean_velocity_advect.F90
ocean_param/lateral/ocean_bihcgrid_friction.F90
ocean_param/lateral/ocean_bihcst_friction.F90
ocean_param/lateral/ocean_bihgen_friction.F90
ocean_param/lateral/ocean_lapcgrid_friction.F90
ocean_param/lateral/ocean_lapcst_friction.F90
ocean_param/lateral/ocean_lapgen_friction.F90
ocean_param/vertical/ocean_vert_mix.F90
ocean_param/sources/ocean_momentum_source.F90
```

42.1 A split-explicit algorithm to time step momentum

In this section, we summarize elements of the split-explicit algorithm used to time step the momentum equations. Although essential for practical reasons, the split-explicit methods introduce a level of complexity to the model algorithms that can be frustrating, particularly when aiming to cleanly diagnose terms contributing to the momentum budget. This section may be readily skipped for those merely interested in the momentum budget diagnostics, which are summarized in Section 42.2.

As discussed in Section 11.1, an algorithm of practical utility for climate modeling must split the fast and slow dynamics so that the slow dynamics can be updated with a much longer time step than the fast dynamics. These algorithms are known as split-explicit methods. For a hydrostatic fluid, the fast motions can be approximated by the vertically integrated dynamics of Section 10.9 and the vertically integrated mass or volume budgets of Section 10.6. The remainder constitutes an approximation to the slower dynamics. Motions constituting the fast dynamics are embodied by the *barotropic* or *external* mode, and the slower motions are embodied by advection as well as *baroclinic* or *internal* waves.

42.1.1 Splitting the velocity field

We formulate the split between the fast and slow modes using density weighting within the mass-conserving non-Boussinesq framework. For Boussinesq equations, the density weighting reduces to a constant ρ_0 reference density factor that trivially cancels from the equations.

Following the discussion in Section 12.3.5 of Griffies (2004), we split of the horizontal velocity field according to

$$\mathbf{u} = \underbrace{\left(\mathbf{u} - \frac{\sum_k \mathbf{u} \rho dz}{\sum_k \rho dz} \right)}_{\text{slow}} + \underbrace{\left(\frac{\sum_k \mathbf{u} \rho dz}{\sum_k \rho dz} \right)}_{\text{fast}} \quad (42.1a)$$

$$\equiv \hat{\mathbf{u}} + \bar{\mathbf{u}}^z, \quad (42.1b)$$

where the vertical sum includes all grid cells within a column of fluid. The vertically averaged velocity is

given by

$$\bar{\mathbf{u}}^z = \frac{\sum_k \mathbf{u} \rho dz}{\sum_k \rho dz} \quad (42.2a)$$

$$= \frac{\mathbf{U}^\rho}{\sum_k \rho dz}. \quad (42.2b)$$

The vertically averaged velocity is updated according to the vertically integrated momentum equation of Section 10.9. The slower depth dependent velocity, $\hat{\mathbf{u}}$, has zero density weighted vertical sum. Therefore, its time tendency is independent of any depth independent forcing. A relevant depth independent forcing of concern arises from fast fluctuations in the surface height or bottom pressure associated with external gravity waves. We thus choose to update the slow dynamics using all pieces of the momentum forcing, *except* contributions from the rapid pressure and geopotential fluctuations contributing to the horizontal pressure gradient. This update produces a velocity \mathbf{u}' that is related to $\hat{\mathbf{u}}$ via

$$\hat{\mathbf{u}} = \mathbf{u}' - \left(\frac{\sum_k \mathbf{u}' \rho dz}{\sum_k \rho dz} \right). \quad (42.3)$$

A similar relation was discussed in Section 12.4.2 of Griffies (2004). Completing the updates of \mathbf{u}' and \mathbf{U}^ρ to a new time level, $\tau + 1$, allows for an update of the full horizontal velocity via the identity

$$\mathbf{u} = \hat{\mathbf{u}} + \bar{\mathbf{u}}^z \quad (42.4a)$$

$$= \left(\mathbf{u}' - \frac{\sum_k \mathbf{u}' \rho dz}{\sum_k \rho dz} \right) + \frac{\mathbf{U}^\rho}{\sum_k \rho dz} \quad (42.4b)$$

$$= \mathbf{u}' + \frac{\sum_k (\mathbf{u} - \mathbf{u}') \rho dz}{\sum_k \rho dz}. \quad (42.4c)$$

42.1.2 Details of the semi-discrete momentum equation

From Section 4.1.5, the horizontal linear momentum budgets for interior, bottom, and surface grid cells are given by

$$\begin{aligned} [\partial_t + (f + \mathcal{M})\hat{\mathbf{z}} \wedge](\mathbf{u} \rho dz) &= \rho dz S^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] - dz (\nabla_s p + \rho \nabla_s \Phi) + dz \rho \mathbf{F} \\ &\quad - [\rho (w^{(s)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_{k-1}} + [\rho (w^{(s)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_k} \end{aligned} \quad (42.5)$$

$$\begin{aligned} [\partial_t + (f + \mathcal{M})\hat{\mathbf{z}} \wedge](\mathbf{u} \rho dz) &= \rho dz S^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] - dz (\nabla_s p + \rho \nabla_s \Phi) + dz \rho \mathbf{F} \\ &\quad - [\rho (w^{(s)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_{kbot-1}} - \tau^{\text{bottom}} \end{aligned} \quad (42.6)$$

$$\begin{aligned} [\partial_t + (f + \mathcal{M})\hat{\mathbf{z}} \wedge](\mathbf{u} \rho dz) &= \rho dz S^{(\mathbf{u})} - \nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] - dz (\nabla_s p + \rho \nabla_s \Phi) + dz \rho \mathbf{F} \\ &\quad + [\tau^{\text{wind}} + Q_m \mathbf{u}_m] + [\rho (w^{(s)} \mathbf{u} - \kappa \mathbf{u}_{,z})]_{s=s_{k=1}}. \end{aligned} \quad (42.7)$$

See Section 4.1.5 for definitions of the various terms in these equations. Equations (42.5)-(42.7) provide prognostic expressions for the linear momentum per unit area of a grid cell

$$\mathbf{u} \rho dz = \text{linear momentum per horizontal area (kg m}^{-1}\text{s}^{-1}), \quad (42.8)$$

which is determined by the horizontal velocity, \mathbf{u} , multiplied by the mass per unit area of a grid cell

$$dz \rho = \text{mass per unit horizontal area (kg m}^{-2}\text{)}. \quad (42.9)$$

If the mass per unit area is time independent, the linear momentum budget reduces to a budget for the horizontal velocity. However, such grid cells are not the norm. More generally, the mass per unit area is a function of time, since the volume and mass of a grid cell changes. Hence, in this more general case, it is critical to formulate the semi-discrete momentum equations using finite volume formulations. Doing so ensures a conservation-based budget for the momentum in a grid cell.

Introducing a forward time step for the time tendency in equations (42.5)-(42.7) renders

$$\frac{(\mathbf{u}\rho dz)_{\tau+1} - (\mathbf{u}\rho dz)_{\tau}}{\Delta\tau} = -f\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) + \rho dz \mathcal{S}^{(\mathbf{u})} - \left(\nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] + \mathcal{M}\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) \right) \\ - dz(\nabla_s p + \rho \nabla_s \Phi) + dz\rho \mathbf{F} - [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{k-1}} + [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_k} \quad (42.10)$$

$$\frac{(\mathbf{u}\rho dz)_{\tau+1} - (\mathbf{u}\rho dz)_{\tau}}{\Delta\tau} = -f\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) + \rho dz \mathcal{S}^{(\mathbf{u})} - \left(\nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] + \mathcal{M}\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) \right) \\ - dz(\nabla_s p + \rho \nabla_s \Phi) + dz\rho \mathbf{F} - [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{kbot-1}} - \tau^{\text{bottom}} \quad (42.11)$$

$$\frac{(\mathbf{u}\rho dz)_{\tau+1} - (\mathbf{u}\rho dz)_{\tau}}{\Delta\tau} = -f\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) + \rho dz \mathcal{S}^{(\mathbf{u})} - \left(\nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] + \mathcal{M}\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) \right) \\ - dz(\nabla_s p + \rho \nabla_s \Phi) + dz\rho \mathbf{F} + [\tau^{\text{wind}} + Q_m \mathbf{u}_m] + [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{k=1}}. \quad (42.12)$$

We separated the term $(f + \mathcal{M})\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz)$ into two parts to correspond to the code implementation. That is, in MOM5, the Coriolis term

$$\text{Coriolis force per area} = -f\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) \quad (42.13)$$

is handled separately from the advection metric term

$$\text{advection metric} = -\mathcal{M}\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz). \quad (42.14)$$

The advection metric term is combined with the horizontal advection operator

$$\text{horizontal advection} = -\nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})], \quad (42.15)$$

whereas the Coriolis force is time stepped semi-implicitly when using the B-grid option (see Section 12.2), or with an Adams-Bashforth scheme when using the C-grid (see Section 12.3).

As already stated in Section 42.1.1, the full momentum equations (42.10)-(42.12) are stable only if taking a small time step sufficient to resolve fast barotropic wave motions. The split-explicit method is realized by splitting the pressure gradient body force into a fast and a slow piece

$$\nabla_s p + \rho \nabla_s \Phi = (\nabla_s p + \rho \nabla_s \Phi)_{\text{fast}} + (\nabla_s p + \rho \nabla_s \Phi)_{\text{slow}}. \quad (42.16)$$

Expressions for the fast and slow components of the pressure gradient depend on the choice for vertical coordinate, to which we return to in Sections 42.1.2.1 and 42.1.2.2. The slow portion of the split-explicit method time steps a primed velocity, \mathbf{u}' . Forcing for \mathbf{u}' is identical to the full momentum, sans the fast component of the pressure gradient in equations (42.10)-(42.12), so that

$$\frac{(\mathbf{u}\rho dz')_{\tau+1} - (\mathbf{u}\rho dz)_{\tau}}{\Delta\tau} = -f\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) + \rho dz \mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] - \mathcal{M}\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) \\ - dz(\nabla_s p + \rho \nabla_s \Phi)_{\text{slow}} + dz\rho \mathbf{F} - [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{k-1}} + [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_k} \quad (42.17)$$

$$\frac{(\mathbf{u}\rho dz')_{\tau+1} - (\mathbf{u}\rho dz)_{\tau}}{\Delta\tau} = -f\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) + \rho dz \mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] - \mathcal{M}\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) \\ - dz(\nabla_s p + \rho \nabla_s \Phi)_{\text{slow}} + dz\rho \mathbf{F} - [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{kbot-1}} - \tau^{\text{bottom}} \quad (42.18)$$

$$\frac{(\mathbf{u}\rho dz')_{\tau+1} - (\mathbf{u}\rho dz)_{\tau}}{\Delta\tau} = -f\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) + \rho dz \mathcal{S}^{(\mathbf{u})} - \nabla_s \cdot [dz\mathbf{u}(\rho\mathbf{u})] - \mathcal{M}\hat{\mathbf{z}} \wedge (\mathbf{u}\rho dz) \\ - dz(\nabla_s p + \rho \nabla_s \Phi)_{\text{slow}} + dz\rho \mathbf{F} + [\tau^{\text{wind}} + Q_m \mathbf{u}_m] + [\rho(w^{(s)}\mathbf{u} - \kappa\mathbf{u}_{,z})]_{s=s_{k=1}}. \quad (42.19)$$

All terms on the right hand side of equations (42.17)-(42.19) make use of the full velocity field \mathbf{u} , rather than the primed velocity field, \mathbf{u}' . Hence, the difference between the updated full velocity and the updated primed velocity is just due to the fast portion of the pressure gradient

$$\frac{(\mathbf{u}\rho dz)_{\tau+1} - (\mathbf{u}\rho dz')_{\tau+1}}{\Delta\tau} = -dz(\nabla_s p + \rho \nabla_s \Phi)_{\text{fast}}. \quad (42.20)$$

This identity holds only approximately in the model implementation, given that the fast pressure gradient is diagnosed as a time average over the small time steps of the barotropic cycle (see Section 11.3).

42.1.2.1 Split of the pressure gradient force with pressure-based vertical coordinates

When making use of pressure-based vertical coordinates in MOM5, equation (3.27) from Section 3.2 allows us to decompose the pressure gradient as

$$\nabla_s p + \rho \nabla_s \Phi = \underbrace{(\rho/\rho_o) \nabla (p_b + \rho_o \Phi_b)}_{\text{fast}} + \underbrace{\rho \nabla_s \Phi' - (\rho'/\rho_o) \nabla_s p}_{\text{slow}}. \quad (42.21)$$

Substitution of this expression in equation (42.20) yields

$$\mathbf{u} = \hat{\mathbf{u}} + \bar{\mathbf{u}}^z \quad (42.22a)$$

$$= \left(\mathbf{u}' - \frac{\sum_k \mathbf{u}' \rho dz}{\sum_k \rho dz} \right) + \frac{\mathbf{U}^\rho}{\sum_k \rho dz} \quad (42.22b)$$

$$= \mathbf{u}' + \frac{\sum_k (\mathbf{u} - \mathbf{u}') \rho dz}{\sum_k \rho dz} \quad (42.22c)$$

$$= \mathbf{u}' - \Delta \tau \left(\frac{\sum_k \rho dz \nabla (p_b + \rho_o \Phi_b)}{\rho_o \sum_k \rho dz} \right) \quad (42.22d)$$

$$= \mathbf{u}' - \Delta \tau \left(\frac{\nabla (p_b + \rho_o \Phi_b)}{\rho_o} \right). \quad (42.22e)$$

The final equality results from noting that

$$p_b + \rho_o \Phi_b = p_b - \rho_o g H \quad (42.23)$$

is depth independent, where

$$\Phi_b = -g H(x, y) \quad (42.24)$$

is the geopotential at the ocean bottom.

42.1.2.2 Split of the pressure gradient force with depth-based vertical coordinates

When making use of depth-based vertical coordinates in MOM5, equation (3.18) from Section 3.2 allows us to decompose the pressure gradient as

$$\nabla_s p + \rho \nabla_s \Phi = \underbrace{\nabla (p_a + g \rho_o \eta)}_{\text{fast}} + \underbrace{\nabla_s p' + \rho' \nabla_s \Phi}_{\text{slow}}. \quad (42.25)$$

Substitution of this expression in equation (42.20), along with setting $\rho dz = \rho_o dz$ for the Boussinesq fluid, yields

$$\mathbf{u} = \hat{\mathbf{u}} + \bar{\mathbf{u}}^z \quad (42.26a)$$

$$= \left(\mathbf{u}' - \frac{\sum_k \mathbf{u}' \rho dz}{\sum_k \rho dz} \right) + \frac{\mathbf{U}^\rho}{\sum_k \rho dz} \quad (42.26b)$$

$$= \mathbf{u}' + \frac{\sum_k (\mathbf{u} - \mathbf{u}') \rho dz}{\sum_k \rho dz} \quad (42.26c)$$

$$= \mathbf{u}' - \Delta \tau \left(\frac{\sum_k \rho dz \nabla (p_a + g \rho_o \eta)}{\rho_o \sum_k \rho dz} \right) \quad (42.26d)$$

$$= \mathbf{u}' - \Delta \tau \left(\frac{\nabla (p_a + g \rho_o \eta)}{\rho_o} \right). \quad (42.26e)$$

The final equality results from noting that $p_a + g \rho_o \eta$ is depth independent.

42.2 Momentum budget diagnostics

In this section, we develop the necessary diagnostic terms required to evaluate terms appearing in the momentum budget. Approximations are inherent in this diagnostic due to the split-explicit algorithm, whereby the fast dynamics is separately time stepped from the slow dynamics. We nonetheless detail the diagnostics assuming these time truncation errors do not damage the physical integrity of the diagnostics.

42.2.1 Decomposing the momentum budget

In Section 42.1.2, we wrote the semi-discrete momentum budget (42.10)-(42.12). We rewrite here that budget, decomposing forcing terms according to how they are evaluated in MOM5

$$\begin{aligned}
 \underbrace{\frac{(\mathbf{u} \rho \, dz)_{\tau+1} - (\mathbf{u} \rho \, dz)_{\tau}}{\Delta \tau}}_{\text{net time tendency}} &= \underbrace{-f \hat{\mathbf{z}} \wedge (\mathbf{u} \rho \, dz)}_{\text{Coriolis}} + \underbrace{\rho \, dz \mathcal{S}(\mathbf{u})}_{\text{source}} \\
 &\quad - \underbrace{\left(\nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] + \mathcal{M} \hat{\mathbf{z}} \wedge (\mathbf{u} \rho \, dz) + [\rho w^{(s)} \mathbf{u}]_{s=s_{k-1}} - [\rho w^{(s)} \mathbf{u}]_{s=s_k} \right)}_{\text{horizontal + vertical advection}} \\
 &\quad - \underbrace{dz (\nabla_s p + \rho \nabla_s \Phi)}_{\text{baroclinic press force}} - \underbrace{dz (\nabla_s p + \rho \nabla_s \Phi)}_{\text{time averaged fast press force}} \\
 &\quad + \underbrace{dz \rho \mathbf{F}}_{\text{horizontal friction}} - \underbrace{[(\rho \kappa \mathbf{u}_{,z})_{s=s_k} - (\rho \kappa \mathbf{u}_{,z})_{s=s_{k-1}}]}_{\text{vertical friction}}
 \end{aligned} \tag{42.27}$$

$$\begin{aligned}
 \underbrace{\frac{(\mathbf{u} \rho \, dz)_{\tau+1} - (\mathbf{u} \rho \, dz)_{\tau}}{\Delta \tau}}_{\text{net time tendency}} &= \underbrace{-f \hat{\mathbf{z}} \wedge (\mathbf{u} \rho \, dz)}_{\text{Coriolis}} + \underbrace{\rho \, dz \mathcal{S}(\mathbf{u})}_{\text{source}} \\
 &\quad - \underbrace{\left(\nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] + \mathcal{M} \hat{\mathbf{z}} \wedge (\mathbf{u} \rho \, dz) + [\rho w^{(s)} \mathbf{u}]_{s=s_{kbot-1}} \right)}_{\text{horizontal + vertical advection}} \\
 &\quad - \underbrace{dz (\nabla_s p + \rho \nabla_s \Phi)}_{\text{baroclinic press force}} - \underbrace{dz (\nabla_s p + \rho \nabla_s \Phi)}_{\text{time averaged fast press force}} \\
 &\quad + \underbrace{dz \rho \mathbf{F}}_{\text{horizontal friction}} - \underbrace{\left(\tau^{\text{bottom}} - (\rho \kappa \mathbf{u}_{,z})_{s=s_{kbot-1}} \right)}_{\text{vertical friction and bottom drag}}
 \end{aligned} \tag{42.28}$$

$$\begin{aligned}
 \underbrace{\frac{(\mathbf{u} \rho \, dz)_{\tau+1} - (\mathbf{u} \rho \, dz)_{\tau}}{\Delta \tau}}_{\text{net time tendency}} &= \underbrace{-f \hat{\mathbf{z}} \wedge (\mathbf{u} \rho \, dz)}_{\text{Coriolis}} + \underbrace{\rho \, dz \mathcal{S}(\mathbf{u})}_{\text{source}} \\
 &\quad - \underbrace{\left(\nabla_s \cdot [dz \mathbf{u} (\rho \mathbf{u})] + \mathcal{M} \hat{\mathbf{z}} \wedge (\mathbf{u} \rho \, dz) - [\rho w^{(s)} \mathbf{u}]_{s=s_{k=1}} - Q_m \mathbf{u}_m \right)}_{\text{horizontal + vertical advection}} \\
 &\quad - \underbrace{dz (\nabla_s p + \rho \nabla_s \Phi)}_{\text{baroclinic press force}} - \underbrace{dz (\nabla_s p + \rho \nabla_s \Phi)}_{\text{time averaged fast press force}} \\
 &\quad + \underbrace{dz \rho \mathbf{F}}_{\text{horizontal friction}} - \underbrace{\left((\rho \kappa \mathbf{u}_{,z})_{s=s_{k=1}} - \tau^{\text{wind}} \right)}_{\text{vertical friction and surf stress}} .
 \end{aligned} \tag{42.29}$$

Note how the bottom drag, τ^{bottom} , and surface stress, τ^{wind} , are both wrapped into the vertical friction terms. That is because bottom drag and surface stress are introduced to the model as boundary conditions for the vertical momentum friction equation. If desired, these terms can be separately diagnosed.

42.2.2 MOM5 diagnostics for the momentum budget

The following MOM5 diagnostics are available for the momentum budget.

42.2.2.1 Net change in grid cell momentum per area

The net change in grid cell momentum per horizontal cell area is diagnosed by

$$\text{momentum_change_u} = \frac{(u \rho \, dz)_{\tau+1} - (u \rho \, dz)_{\tau}}{\Delta \tau} \quad (42.30a)$$

$$\text{momentum_change_v} = \frac{(v \rho \, dz)_{\tau+1} - (v \rho \, dz)_{\tau}}{\Delta \tau}, \quad (42.30b)$$

with units of N m^{-2} . These diagnostics are evaluated at the end of a time step, once the full update for the new momentum per area, $(u \rho \, dz)_{\tau+1}$, is available. Again, for a Boussinesq model, the *in situ* density factor, ρ , is set to the constant Boussinesq reference density, ρ_0 .

This diagnostic has not yet been coded in MOM5. Code is needed inside ocean_core/ocean_velocity.F90.

42.2.2.2 Coriolis force per area

The Coriolis force per horizontal grid cell area is diagnosed by the two terms

$$\text{hrho_cor_u} + \text{ucori_impl} = f(v \rho \, dz) \quad (42.31a)$$

$$\text{hrho_cor_v} + \text{vcori_impl} = -f(u \rho \, dz), \quad (42.31b)$$

with units of N m^{-2} . As detailed in Chapter 12, the Coriolis force is discretized in time in one of the following manners:

- B-grid: via a semi-implicit method;
- C-grid: via a third order Adams-Bashforth method, in which case `ucori_impl` and `vcori_impl` are zero.

For a Boussinesq model, the *in situ* density factor, ρ , is set to the constant Boussinesq reference density, ρ_0 .

42.2.2.3 Momentum source/sink

The source/sink term for momentum is diagnosed by

$$\text{rayleigh_drag_u} = \rho \, dz S^{(u)} \quad (42.32a)$$

$$\text{rayleigh_drag_v} = \rho \, dz S^{(v)} \quad (42.32b)$$

with units of N m^{-2} . This particular example of a source/sink term concerns the effects from dissipation parameterized as Rayleigh drag.

42.2.2.4 Momentum advection

For interior grid cells, momentum advection is diagnosed by

$$\text{momentum_advect_u} = - \left(\nabla_s \cdot [u(u \rho \, dz)] - \mathcal{M}(v \rho \, dz) + [\rho w^{(s)} u]_{s=s_{k-1}} - [\rho w^{(s)} u]_{s=s_k} \right) \quad (42.33a)$$

$$\text{momentum_advect_v} = - \left(\nabla_s \cdot [v(u \rho \, dz)] + \mathcal{M}(u \rho \, dz) + [\rho w^{(s)} v]_{s=s_{k-1}} - [\rho w^{(s)} v]_{s=s_k} \right) \quad (42.33b)$$

with units of N m^{-2} . For bottom grid cells, momentum advection is diagnosed by

$$\text{momentum_advect_u} = -\left(\nabla_s \cdot [u(\mathbf{u} \rho dz)] - \mathcal{M}(v \rho dz) + [\rho w^{(s)} u]_{s=s_{kbot-1}}\right) \quad (42.34a)$$

$$\text{momentum_advect_v} = -\left(\nabla_s \cdot [v(\mathbf{u} \rho dz)] + \mathcal{M}(u \rho dz) + [\rho w^{(s)} v]_{s=s_{k-1}}\right) \quad (42.34b)$$

with units of N m^{-2} . For surface grid cells, momentum advection is diagnosed by

$$\text{momentum_advect_u} = -\left(\nabla_s \cdot [u(\mathbf{u} \rho dz)] - \mathcal{M}(v \rho dz) - [\rho w^{(s)} u]_{s=s_{k=1}} - Q_m u_m\right) \quad (42.35a)$$

$$\text{momentum_advect_v} = -\left(\nabla_s \cdot [v(\mathbf{u} \rho dz)] + \mathcal{M}(u \rho dz) - [\rho w^{(s)} v]_{s=s_{k=1}} - Q_m v_m\right), \quad (42.35b)$$

with units of N m^{-2} . As discussed in Section 11.2.3, the time discretization of momentum advection is generally handled via a third-order Adams-Bashforth method. The third order Adams-Bashforth method is an explicit scheme that requires the advection operator to be evaluated at time steps τ , $\tau - 1$, and $\tau - 2$.

This diagnostic has not yet been coded in MOM5. Code is needed inside ocean_core/ocean_velocity.F90.

42.2.2.5 Slow/baroclinic pressure gradient force per area

According to Section 42.1.2.1, the slow/baroclinic pressure gradient force for pressure-based vertical coordinates is diagnosed by

$$\text{press_baroclinic_force_u} = -dz \left[\rho \left(\frac{\partial \Phi'}{\partial x} \right)_s - (\rho'/\rho_o) \left(\frac{\partial p}{\partial x} \right)_s \right] \quad (42.36a)$$

$$\text{press_baroclinic_force_v} = -dz \left[\rho \left(\frac{\partial \Phi'}{\partial y} \right)_s - (\rho'/\rho_o) \left(\frac{\partial p}{\partial y} \right)_s \right] \quad (42.36b)$$

with units of $\text{N m}^{-2} = \text{Pa}$. For depth-based vertical coordinates, we follow the details in Section 42.1.2.2 to have the slow pressure gradient force diagnosed as

$$\text{press_baroclinic_force_u} = -dz \left[\left(\frac{\partial p'}{\partial x} \right)_s + \rho' \left(\frac{\partial \Phi}{\partial x} \right)_s \right] \quad (42.37a)$$

$$\text{press_baroclinic_force_v} = -dz \left[\left(\frac{\partial p'}{\partial y} \right)_s + \rho' \left(\frac{\partial \Phi}{\partial y} \right)_s \right]. \quad (42.37b)$$

This diagnostic has a different name in MOM5; the “baroclinic” portion of the name is missing. We need to make the name change to better align this diagnostic with that for the fast/barotropic pressure gradient. Code modification is needed inside ocean_core/ocean_pressure.F90.

42.2.2.6 Fast/barotropic pressure gradient force per area

According to Section 42.1.2.1, the fast/barotropic pressure gradient force for pressure-based vertical coordinates is diagnosed by

$$\text{press_barotropic_force_u} = -(\rho dz/\rho_o) \left[\frac{\partial p_b}{\partial x} + \rho_o \frac{\partial \Phi_b}{\partial x} \right] \quad (42.38a)$$

$$\text{press_barotropic_force_v} = -(\rho dz/\rho_o) \left[\frac{\partial p_b}{\partial y} + \rho_o \frac{\partial \Phi_b}{\partial y} \right] \quad (42.38b)$$

with units of $\text{N m}^{-2} = \text{Pa}$. For depth-based vertical coordinates, we follow the details in Section 42.1.2.2 to have the fast/barotropic pressure gradient force diagnosed as

$$\text{press_barotropic_force_u} = -dz \left[\frac{\partial p_a}{\partial x} + g \rho_o \frac{\partial \eta}{\partial x} \right] \quad (42.39a)$$

$$\text{press_barotropic_force_v} = -dz \left[\frac{\partial p_a}{\partial y} + g \rho_o \frac{\partial \eta}{\partial y} \right]. \quad (42.39b)$$

This diagnostic has not yet been coded in MOM5. Code is needed inside `ocean_core/ocean_barotropic.F90`.

42.2.2.7 Horizontal viscous friction from Laplacian and biharmonic operators

The horizontal viscous friction from a Laplacian and/or a biharmonic operator is diagnosed by

$$\text{bih_fric_u} + \text{lap_fric_u} = \underbrace{dz \rho F^{(u)}}_{\text{horizontal friction}} \quad (42.40a)$$

$$\text{bih_fric_v} + \text{lap_fric_v} = \underbrace{dz \rho F^{(v)}}_{\text{horizontal friction}} \quad (42.40b)$$

with units of N m^{-2} .

The Laplacian diagnostic code appears to have a bug; the diagnostic is spuriously called twice. Code modificatio is needed inside `ocean_param/lateral/ocean_lapgen_friction.F90`.

42.2.2.8 Vertical viscous friction, bottom drag, and surface stress

The vertical viscous friction in the ocean interior contains the following term

$$\text{vfrict_impl_u} = - \left[\left(\rho \kappa \frac{\partial u}{\partial z} \right)_{s=s_k} - \left(\rho \kappa \frac{\partial u}{\partial z} \right)_{s=s_{k-1}} \right] \quad (42.41a)$$

$$\text{vfrict_impl_v} = - \left[\left(\rho \kappa \frac{\partial v}{\partial z} \right)_{s=s_k} - \left(\rho \kappa \frac{\partial v}{\partial z} \right)_{s=s_{k-1}} \right] \quad (42.41b)$$

with units of N m^{-2} . This term is generally time discretized using implicit methods, hence the name for the diagnostic. For ocean surface cells, we absorb the surface stress into this diagnostic, thus yielding

$$\text{vfrict_impl_u} = - \left[\left(\rho \kappa \frac{\partial u}{\partial z} \right)_{s=s_{k=1}} - \tau_{(x)}^{\text{wind}} \right] \quad (42.42a)$$

$$\text{vfrict_impl_v} = - \left[\left(\rho \kappa \frac{\partial v}{\partial z} \right)_{s=s_{k=1}} - \tau_{(y)}^{\text{wind}} \right] \quad (42.42b)$$

For bottom ocean cells, we absorb the bottom drag into this diagnostic, thus yielding

$$\text{vfrict_impl_u} = - \left[\tau_{(x)}^{\text{drag}} - \left(\rho \kappa \frac{\partial u}{\partial z} \right)_{s=s_{kbot=1}} \right] \quad (42.43a)$$

$$\text{vfrict_impl_v} = - \left[\tau_{(y)}^{\text{drag}} - \left(\rho \kappa \frac{\partial v}{\partial z} \right)_{s=s_{kbot=1}} \right]. \quad (42.43b)$$

We can separately diagnose the surface and bottom stress via the following diagnostics

$$\text{bmf_u} = -\tau_{(x)}^{\text{drag}} \quad (42.44a)$$

$$\text{bmf_v} = -\tau_{(y)}^{\text{drag}} \quad (42.44b)$$

$$\text{tau_x} = \tau_{(x)}^{\text{wind}} \quad (42.44c)$$

$$\text{tau_y} = \tau_{(y)}^{\text{wind}}. \quad (42.44d)$$

42.2.2.9 Summary of the momentum diagnostic terms

In summary, the following terms contribute to the budget for the momentum per horizontal area of a grid cell

$$\begin{aligned} \text{momentum_change_u} = & \text{hrho_cor_u} + \text{ucori_impl} + \text{momentum_advect_u} \\ & + \text{press_baroclinic_force_u} + \text{press_barotropic_force_u} \\ & + \text{bih_fric_u} + \text{lap_fric_u} + \text{vfriact_impl_u} + \text{rayleigh_drag_u} \end{aligned} \quad (42.45)$$

$$\begin{aligned} \text{momentum_change_v} = & \text{hrho_cor_v} + \text{vcori_impl} + \text{momentum_advect_v} \\ & + \text{press_baroclinic_force_v} + \text{press_barotropic_force_v} \\ & + \text{bih_fric_v} + \text{lap_fric_v} + \text{vfriact_impl_v} + \text{rayleigh_drag_v}. \end{aligned} \quad (42.46)$$

Every term in these equations has units of N m^{-2} . Multiplication by the grid cell area yields the force acting on a grid cell due to the respective terms.

Bibliography

, ????

- Adcroft, A., Campin, J.-M., 2004. Rescaled height coordinates for accurate representation of free-surface flows in ocean circulation models. *Ocean Modelling* 7, 269–284.
- Adcroft, A., Hallberg, R., Harrison, M., 2008. A finite volume discretization of the pressure gradient force using analytic integration. *Ocean Modelling* 22, 106–113.
- Adcroft, A., Hallberg, R. W., 2006. On methods for solving the oceanic equations of motion in generalized vertical coordinates. *Ocean Modelling* 11, 224–233.
- Adcroft, A., Hill, C., Marshall, J., 1997. Representation of topography by shaved cells in a height coordinate ocean model. *Monthly Weather Review* 125, 2293–2315.
- Adcroft, A., Hill, C., Marshall, J., 1999. A new treatment of the coriolis terms in c-grid models at both high and low resolutions. *Monthly Weather Review* 127, 1928–1936.
- Adcroft, A., Scott, J. R., Marotzke, J., 2001. Impact of geothermal heating on the global ocean circulation. *Geophysical Research Letters* 28, 1735–1738.
- Aiki, H., Jacobson, T., Yamagata, T., 2004. Parameterizing ocean eddy transports from surface to bottom. *Journal of Geophysical Research* 31, L19 302, doi:10.1029/2004GL020703.
- Apel, J. R., 1987. *Principles of Ocean Physics*. Vol. 38 of International Geophysics Series. Academic Press, London.
- Arbic, B., Garner, S. T., Hallberg, R. W., Simmons, H. L., 2004. The accuracy of surface elevations in forward global barotropic and baroclinic tide models. *Deep Sea Research* 51, 3069–3101.
- Aris, R., 1962. *Vectors, Tensors and the Basic Equations of Fluid Mechanics*. Dover Publishing, New York.
- Asselin, R., 1972. Frequency filter for time integrations. *Monthly Weather Review* 100, 487–490.
- Bacon, S., Fofonoff, N. P., 1996. Oceanic heat flux calculation. *Journal of Atmospheric and Oceanic Technology* 13, 1327–1329.
- Batchelor, G. K., 1967. *An Introduction to Fluid Dynamics*. Cambridge University Press, Cambridge, England.
- Bates, M., 2011. Lagrangian blobs embedded in Eulerian models: a framework to parameterise vertical and downslope motion of gravitationally unstable water parcels. University of New South Wales, Sydney, Australia.
- Bates, M., Griffies, S. M., England, M., 2012a. A dynamic, embedded Lagrangian model for ocean climate models, Part I: Theory and implementation. *Ocean Modelling* 59–60, 41–59.

- Bates, M., Griffies, S. M., England, M., 2012b. A dynamic, embedded Lagrangian model for ocean climate models, Part II: Idealised overflow tests. *Ocean Modelling* 59-60, 60–76.
- Beckmann, A., Döscher, R., 1997. A method for improved representation of dense water spreading over topography in geopotential-coordinate models. *Journal of Physical Oceanography* 27, 581–591.
- Black, T. L., 1994. The new NMC mesoscale eta model: description and forecast examples. *Weather and Forecasting* 9, 265–278.
- Bleck, R., 1978. Finite difference equations in generalized vertical coordinates. Part I: Total energy conservation. *Contributions to Atmospheric Physics* 51, 360–372.
- Blumberg, A., Kantha, L., 1985. Open boundary condition for circulation models. *Journal of Hydraulic Engineering* 111, 237–255.
- Blumberg, A. F., Mellor, G. L., 1987. A description of a three-dimensional coastal ocean circulation model. In: Heaps, N. (Ed.), *Three-Dimensional Coastal Ocean Models*. Vol. 4 of Coastal and Estuarine Series. American Geophysical Union.
- Bryan, K., 1963. A numerical investigation of a nonlinear model of a wind-driven ocean. *Journal of Atmospheric Sciences* 20, 594–606.
- Bryan, K., 1969. A numerical method for the study of the circulation of the world ocean. *Journal of Computational Physics* 4, 347–376.
- Bryan, K., 1989. The design of numerical models of the ocean circulation. In: Anderson, D. L., Willebrand, J. (Eds.), *Oceanic Circulation Models: Combining Data and Dynamics*. Vol. 284 of NATO ASI Series. Series C. Kluwer Academic Publishers, pp. 465–511.
- Bryan, K., 1991. Michael cox (1941-1989): his pioneering contributions to ocean circulation modeling. *Journal of Physical Oceanography* 21, 1259–1270.
- Bryan, K., 1996. The steric component of sea level rise associated with enhanced greenhouse warming: a model study. *Climate Dynamics* 12, 545–555.
- Bryan, K., Cox, M. D., 1972. An approximate equation of state for numerical models of the ocean circulation. *Journal of Physical Oceanography* 4, 510–514.
- Bryan, K., Lewis, L. J., 1979. A water mass model of the world ocean. *Journal of Geophysical Research* 84, 2503–2517.
- Burchard, H., Rennau, H., 2008. Comparative quantification of physically and numerically induced mixing in ocean models. *Ocean Modelling* 20, 293–311.
- Callen, H. B., 1985. *Thermodynamics and an Introduction to Thermostatistics*. John Wiley and Sons, New York, 493 + xvi pp.
- Camerlengo, A., O'Brien, J., 1980. Open boundary condition in rotating fluids. *Journal of Computational Physics* 35, 12–35.
- Campin, J.-M., Adcroft, A., Hill, C., Marshall, J., 2004. Conservation of properties in a free-surface model. *Ocean Modelling* 6, 221–244.
- Campin, J.-M., Goosse, H., 1999. Parameterization of density-driven downsloping flow for a coarse-resolution ocean model in z-coordinate. *Tellus* 51A, 412–430.
- Chaikin, P. M., Lubensky, T. C., 1995. *Principles of Condensed Matter Physics*. Cambridge University Press, Cambridge, United Kingdom.

- Chang, Y. S., Xu, X., Özgökmen, T. M., Chassignet, E. P., Peters, H., Fischer, P. F., 2005. Comparison of gravity current mixing parameterizations and calibration using a high-resolution 3d nonhydrostatic spectral element model. *Ocean Modelling* 3-4, 342–368.
- Chapman, D., 1985. Numerical treatment of cross-shelf boundaries in a barotropic coastal ocean model. *Journal of Physical Oceanography* 103(C1), 1060–1075.
- Chassignet, E. P., Garraffo, Z., 2001. Viscosity parameterization and the Gulf Stream separation. In: Müller, P., Henderson, D. (Eds.), *From Stirring to Mixing in a Stratified Ocean*. Proceedings of the 12th 'Aha Huli' Hawaiian Winter Workshop. University of Hawaii at Manoa, pp. 37–41.
- Chassignet, E. P., Verron, J., 2005. *Ocean Weather Forecasting: an Integrated View of Oceanography*. Springer Publishers.
- Chen, D., Rothstein, L., Busalacchi, A., 1994. A hybrid vertical mixing scheme and its application to tropical ocean models. *Journal of Physical Oceanography* 24, 2156–2179.
- Church, J., Gregory, J., 2001. Changes in sea level. In: *Climate Change 2001: The Scientific Basis*. Cambridge University Press, Cambridge UK, pp. 639–693.
- Conkright, M., Antonov, J., Baranova, O., Boyer, T., Garcia, H., Gelfeld, F., Johnson, D., Locarnini, R., Murphy, P., O'Brien, T., Smolyar, I., Stephens, C., 2002. *World Ocean Database 2001, Volume 1: Introduction*. NOAA Atlas NESDIS 42, U.S. Government Printing Office 13, NOAA, Washington, D.C., 167 pp.
- Cox, M. D., 1984. *A Primitive Equation, 3-Dimensional Model of the Ocean*. NOAA/Geophysical Fluid Dynamics Laboratory, Princeton, USA.
- Cox, M. D., 1987. Isopycnal diffusion in a z-coordinate ocean model. *Ocean Modelling* 74, 1–5.
- Craig, P., Banner, M., 1994. Modeling wave-enhanced turbulence in the surface ocean layer. *Journal of Physical Oceanography* 24, 2546–2559.
- Cushman-Roisin, B., 1987. Subduction. In: *Dynamics of the oceanic surface mixed-layer*. Hawaii Institute of Geophysical Special Publications, pp. 181–196.
- Danabasoglu, G., Ferrari, R., McWilliams, J., 2008. Sensitivity of an ocean general circulation model to a parameterization of near-surface eddy fluxes. *Journal of Climate* 21, 1192–1208.
- Danabasoglu, G., Large, W., Briegleb, B., 2010. Climate impacts of parameterized nordic sea overflows. *Journal of Geophysical Research* 115, C11005, doi:10.1029/2010JC006243.
- Danabasoglu, G., Large, W. G., Tribbia, J. J., Gent, P. R., Briegleb, B. P., McWilliams, J. C., 2006. Diurnal coupling in the tropical oceans of CCSM3. *Journal of Climate* 19, 2347–2365.
- Danabasoglu, G., McWilliams, J. C., 1995. Sensitivity of the global ocean circulation to parameterizations of mesoscale tracer transports. *Journal of Climate* 8, 2967–2987.
- DeGroot, S. R., Mazur, P., 1984. *Non-Equilibrium Thermodynamics*. Dover Publications, New York, 510 pp.
- Delworth, T. L., Broccoli, A. J., Rosati, A., Stouffer, R. J., Balaji, V., Beesley, J. A., Cooke, W. F., Dixon, K. W., Dunne, J., Dunne, K. A., Durachta, J. W., Findell, K. L., Ginoux, P., Gnanadesikan, A., Gordon, C., Griffies, S. M., Gudgel, R., Harrison, M. J., Held, I. M., Hemler, R. S., Horowitz, L. W., Klein, S. A., Knutson, T. R., Kushner, P. J., Langenhorst, A. L., Lee, H.-C., Lin, S., Lu, L., Malyshev, S. L., Milly, P., Ramaswamy, V., Russell, J., Schwarzkopf, M. D., Shevliakova, E., Sirutis, J., Spelman, M., Stern, W. F., Winton, M., Wittenberg, A. T., Wyman, B., Zeng, F., Zhang, R., 2006. GFDL's CM2 global coupled climate models - Part 1: Formulation and simulation characteristics. *Journal of Climate* 19, 643–674.
- Delworth, T. L., Rosati, A., Anderson, W., Adcroft, A. J., Balaji, V., Benson, R., Dixon, K., Griffies, S. M., Lee, H.-C., Pacanowski, R. C., Vecchi, G. A., Wittenberg, A. T., Zeng, F., Zhang, R., 2012. Simulated climate and climate change in the GFDL CM2.5 high-resolution coupled climate model. *Journal of Climate* 25, 2755–2781.

- Denman, K., 1973. A time-dependend model of the upper ocean. *Journal of Physical Oceanography* 3, 173–184.
- Deremble, B., Hogg, A. M., Berloff, P., Dewar, W., 2012. On the application of no-slip lateral boundary conditions to ‘coarsely’ resolved ocean models. *Ocean Modelling*, doi:10.1016/j.ocemod.2011.05.002.
- DeSzoek, R. A., Bennett, A. F., 1993. Microstructure fluxes across density surfaces. *Journal of Physical Oceanography* 23, 2254–2264.
- DeSzoek, R. A., Samelson, R. M., 2002. The duality between the Boussinesq and non-Boussinesq hydrostatic equations of motion. *Journal of Physical Oceanography* 32, 2194–2203.
- Diansky, N., Bagno, A., Zalusky, V., 2002. Global ocean circulation sigma-model and its sensitivity to the wind stress forcing. *Izvestia, Atmospheric and Oceanic Physics* 38, 477–494.
- Donner, L. J., Wyman, B. L., Hemler, R. S., Horowitz, L. W., Ming, Y., Zhao, M., Golaz, J.-C., Ginoux, P., Lin, S.-J., Schwarzkopf, M. D., Austin, J., Alaka, G., Cooke, W. F., Delworth, T. L., Freidenreich, S. M., Gordon, C. T., Griffies, S. M., Held, I. M., Hurlin, W. J., Klein, S. A., Knutson, T. R., Langenhorst, A. R., Lee, H.-C., Lin, Y., Magi, B. I., Malyshev, S. L., Milly, P., Naik, V., Nath, M. J., Pincus, R., Ploshay, J. J., Ramaswamy, V., Seman, C. J., Shevliakova, E., Sirutis, J. J., Stern, W. F., Stouffer, R. J., Wilson, R. J., Winton, M., Wittenberg, A. T., Zeng, F., 2011. The dynamical core, physical parameterizations, and basic simulation characteristics of the atmospheric component of the GFDL global coupled model CM3. *Journal of Climate* 24, 3484–3519.
- Döscher, R., Beckmann, A., 2000. Effects of a bottom boundary layer parameterization in a coarse-resolution model of the North Atlantic Ocean. *Journal of Atmospheric and Oceanic Technology* 17, 698–707.
- Downes, S. M., Gnanadesikan, A., Griffies, S. M., Sarmiento, J., 2011. Water mass exchange in the Southern Ocean in coupled climate models. *Journal of Physical Oceanography* 41, 1756–1771.
- Dunne, J. P., John, J. G., Hallberg, R. W., Griffies, S. M., Shevliakova, E. N., Stouffer, R. J., Krasting, J. P., Sentman, L. A., Milly, P. C. D., Malyshev, S. L., Adcroft, A. J., Cooke, W., Dunne, K. A., Harrison, M. J., Levy, H., Samuels, B. L., Spelman, M., Winton, M., Wittenberg, A. T., Phillips, P. J., Zadeh, N., 2012. GFDLs ESM2 global coupled climate-carbon Earth System Models Part I: Physical formulation and baseline simulation characteristics. *Journal of Climate* 25, 6646–6665.
- Dunne, J. P., John, J. G., Hallberg, R. W., Griffies, S. M., Shevliakova, E. N., Stouffer, R. J., Krasting, J. P., Sentman, L. A., Milly, P. C. D., Malyshev, S. L., Adcroft, A. J., Cooke, W., Dunne, K. A., Harrison, M. J., Levy, H., Wittenberg, A., Phillips, P., Zadeh, N., 2013. GFDLs ESM2 global coupled climate-carbon Earth System Models Part II: Carbon system formulation and baseline simulation characteristics. *Journal of Climate* 26, 2247–2267.
- Durran, D. R., 1999. *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer Verlag, Berlin, 470 pp.
- Durski, S. M., Glenn, S. M., Haidvogel, D. B., 2004. Vertical mixing schemes in the coastal ocean: Comparison of the level 2.5 Mellor-Yamada scheme with an enhanced version of the K profile parameterization. *Journal of Geophysical Research* 109, doi:10.1029/2002JC001702.
- Eby, M., Holloway, G., 1994. Sensitivity of a large scale ocean model to a parameterization of topographic stress. *Journal of Physical Oceanography* 24, 2577–2588.
- Eden, C., Greatbatch, R., Olbers, D., 2007. Interpreting eddy fluxes. *Journal of Physical Oceanography* 37, 1282–1296.
- Eden, C., Jochum, M., Danabasoglu, G., 2009. Effects of different closures for thickness diffusivity. *Ocean Modelling* 26, 47–59.
- Emile-Geay, J., Madec, G., 2009. Geothermal heating, diapycnal mixing and the abyssal circulation. *Ocean Science* 5 (2), 203–217.

- England, M. H., 1995. The age of water and ventilation timescales in a global ocean model. *Journal of Physical Oceanography* 25, 2756–2777.
- Ezer, T., Arango, H., Shchepetkin, A. F., 2002. Developments in terrain-following ocean models: Intercomparisons of numerical aspects. *Ocean Modelling* 4, 249–267.
- Farneti, R., Delworth, T., Rosati, A., Griffies, S. M., Zeng, F., 2010. The role of mesoscale eddies in the rectification of the Southern Ocean response to climate change. *Journal of Physical Oceanography* 40, 1539–1557.
- Farrell, W., Clark, J., 1976. On postglacial sea level. *Geophysical Journal of the Royal Astronomical Society* 46, 646–667.
- Feistel, R., 1993. Equilibrium thermodynamics of seawater revisited. *Progress in Oceanography* 31, 101–179.
- Feistel, R., 2003. A new extended Gibbs thermodynamic potential of seawater. *Progress in Oceanography* 58, 43–114.
- Feistel, R., Hagen, E., 1995. On the Gibbs thermodynamic potential of seawater. *Progress in Oceanography* 36, 249–327.
- Ferrari, R., Griffies, S. M., Nurser, A. J. G., Vallis, G. K., 2010. A boundary-value problem for the parameterized mesoscale eddy transport. *Ocean Modelling* 32, 143–156.
- Ferrari, R., McWilliams, J. C., Canuto, V. M., Dubovikov, M., 2008. Parameterization of eddy fluxes near oceanic boundaries. *Journal of Climate* 21, 2770–2789.
- Ferreira, D., Marshall, J., 2006. Formulation and implementation of a residual-mean ocean circulation model. *Ocean Modelling* 13, 86–107.
- Fofonoff, N. P., 1962. Physical properties of seawater. In: Hill, M. N. (Ed.), *The Sea*. Vol. 1. Wiley-Interscience, pp. 3–30.
- Fox-Kemper, B., Danabasoglu, G., Ferrari, R., Griffies, S. M., Hallberg, R. W., Holland, M., Peacock, S., Samuels, B., 2011. Parameterization of mixed layer eddies. III: Global implementation and impact on ocean climate simulations. *Ocean Modelling* 39, 61–78.
- Fox-Kemper, B., Danabasoglu, G., Ferrari, R., Hallberg, R. W., 2008a. Parameterizing submesoscale physics in global models. *CLIVAR Exchanges* 13, 3–5.
- Fox-Kemper, B., Ferrari, R., Hallberg, R., 2008b. Parameterization of mixed layer eddies. I: Theory and diagnosis. *Journal of Physical Oceanography* 38, 1145–1165.
- Galbraith, E., Gnanadesikan, A., Griffies, S. M., Dunne, J., Wittenberg, A., Held, I., Kwon, E.-Y., Rodgers, K., Bianchi, D., Sarmiento, J., Slater, R., Simeon, J., 2011. The impact of climate variability on the distribution of radiocarbon in CM2Mc, a new earth system model. *Journal of Climate* 24, 4230–4254.
- Gargett, A. E., 1984. Vertical eddy diffusivity in the ocean interior. *Journal of Marine Research* 42, 359–393.
- Gent, P., Bryan, F., Danabasoglu, G., Doney, S., Holland, W., Large, W., McWilliams, J., 1998. The NCAR climate system model global ocean component. *Journal of Climate* 11, 1287–1306.
- Gent, P. R., McWilliams, J. C., 1990. Isopycnal mixing in ocean circulation models. *Journal of Physical Oceanography* 20, 150–155.
- Gent, P. R., Willebrand, J., McDougall, T. J., McWilliams, J. C., 1995. Parameterizing eddy-induced tracer transports in ocean circulation models. *Journal of Physical Oceanography* 25, 463–474.
- Gill, A., 1982. *Atmosphere-Ocean Dynamics*. Vol. 30 of International Geophysics Series. Academic Press, London, 662 + xv pp.

- Gill, A. E., Niiler, P., 1973. The theory of the seasonal variability in the ocean. *Deep-Sea Research* 20 (9), 141–177.
- Gnanadesikan, A., Dixon, K. W., Griffies, S. M., Balaji, V., Beesley, J. A., Cooke, W. F., Delworth, T. L., Gerdes, R., Harrison, M. J., Held, I. M., Hurlin, W. J., Lee, H.-C., Liang, Z., Nong, G., Pacanowski, R. C., Rosati, A., Russell, J., Samuels, B. L., Song, S. M., Spelman, M. J., Stouffer, R. J., Sweeney, C. O., Vecchi, G., Winton, M., Wittenberg, A. T., Zeng, F., Zhang, R., 2006. GFDL's CM2 global coupled climate models-Part 2: The baseline ocean simulation. *Journal of Climate* 19, 675–697.
- Goosens, M., Mittelbach, F., Samarin, A., 1994. *The L^AT_EX Companion*. Addison-Wesley, Reading, Massachusetts, 528 pp.
- Gordon, C., Cooper, C., Senior, C. A., Banks, H., J. M. Gregory, T. C. J., Mitchell, J. F. B., Wood, R. A., 2000. The simulation of SST, sea ice extents and ocean heat transports in a version of the Hadley Centre coupled model without flux adjustments. *Climate Dynamics* 16, 147–168.
- Greatbatch, R. J., 1994. A note on the representation of steric sea level in models that conserve volume rather than mass. *Journal of Geophysical Research* 99, 12767–12771.
- Greatbatch, R. J., Lamb, K. G., 1990. On parameterizing vertical mixing of momentum in non-eddy resolving ocean models. *Journal of Physical Oceanography* 20, 1634–1637.
- Greatbatch, R. J., Lu, Y., Cai, Y., 2001. Relaxing the Boussinesq approximation in ocean circulation models. *Journal of Atmospheric and Oceanic Technology* 18, 1911–1923.
- Greatbatch, R. J., McDougall, T. J., 2003. The non-Boussinesq temporal-residual-mean. *Journal of Physical Oceanography* 33, 1231–1239.
- Greatbatch, R. J., Mellor, G. L., 1999. An overview of coastal ocean models. In: Mooers, C. (Ed.), *Coastal Ocean Prediction*. Vol. 56 of *Coastal and Estuarine Studies*. American Geophysical Union, pp. 31–57.
- Gregg, M., Sanford, T., Winkel, D., 2003. Reduced mixing from the breaking of internal waves in equatorial waters. *Nature* 422, 513–515.
- Griffies, S. M., 1998. The Gent-McWilliams skew-flux. *Journal of Physical Oceanography* 28, 831–841.
- Griffies, S. M., 2004. *Fundamentals of Ocean Climate Models*. Princeton University Press, Princeton, USA, 518+xxxiv pages.
- Griffies, S. M., 2005. Some ocean model fundamentals. In: Chassignet, E. P., Verron, J. (Eds.), *GODAE Summer School*. Springer/Kluwer, pp. 19–74.
- Griffies, S. M., 2009. *Elements of MOM4p1: GFDL Ocean Group Technical Report No. 6*. NOAA/Geophysical Fluid Dynamics Laboratory, Princeton, USA, 444 pp.
- Griffies, S. M., Adcroft, A. J., 2008. Formulating the equations for ocean models. In: Hecht, M., Hasumi, H. (Eds.), *Ocean Modeling in an Eddying Regime*. Geophysical Monograph 177. American Geophysical Union, pp. 281–317.
- Griffies, S. M., Biastoch, A., Böning, C. W., Bryan, F., Danabasoglu, G., Chassignet, E., England, M. H., Gerdes, R., Haak, H., Hallberg, R. W., Hazeleger, W., Jungclaus, J., Large, W. G., Madec, G., Pirani, A., Samuels, B. L., Scheinert, M., Gupta, A. S., Severijns, C. A., Simmons, H. L., Treguier, A. M., Winton, M., Yeager, S., Yin, J., 2009. Coordinated Ocean-ice Reference Experiments (COREs). *Ocean Modelling* 26, 1–46.
- Griffies, S. M., Böning, C. W., Bryan, F. O., Chassignet, E. P., Gerdes, R., Hasumi, H., Hirst, A., Treguier, A.-M., Webb, D., 2000a. Developments in ocean climate modelling. *Ocean Modelling* 2, 123–192.
- Griffies, S. M., Gnanadesikan, A., Dixon, K. W., Dunne, J. P., Gerdes, R., Harrison, M. J., Rosati, A., Russell, J., Samuels, B. L., Spelman, M. J., Winton, M., Zhang, R., 2005. Formulation of an ocean model for global climate simulations. *Ocean Science* 1, 45–79.

- Griffies, S. M., Gnanadesikan, A., Pacanowski, R. C., Larichev, V., Dukowicz, J. K., Smith, R. D., 1998. Isoneutral diffusion in a z -coordinate ocean model. *Journal of Physical Oceanography* 28, 805–830.
- Griffies, S. M., Greatbatch, R. J., 2012. Physical processes that impact the evolution of global mean sea level in ocean climate models. *Ocean Modelling* 51, 37–72.
- Griffies, S. M., Hallberg, R. W., 2000. Biharmonic friction with a Smagorinsky viscosity for use in large-scale eddy-permitting ocean models. *Monthly Weather Review* 128, 2935–2946.
- Griffies, S. M., Harrison, M. J., Pacanowski, R. C., Rosati, A., 2004. A Technical Guide to MOM4. NOAA/Geophysical Fluid Dynamics Laboratory, Princeton, USA, 337 pp.
- Griffies, S. M., Pacanowski, R., Schmidt, M., Balaji, V., 2001. Tracer conservation with an explicit free surface method for z -coordinate ocean models. *Monthly Weather Review* 129, 1081–1098.
- Griffies, S. M., Pacanowski, R. C., Hallberg, R. W., 2000b. Spurious diapycnal mixing associated with advection in a z -coordinate ocean model. *Monthly Weather Review* 128, 538–564.
- Griffies, S. M., Winton, M., Donner, L. J., Downes, S. M., Farneti, R., Gnanadesikan, A., Horowitz, L. W., Hurlin, W. J., Lee, H.-C., Liang, Z., Palter, J. B., Samuels, B. L., Wittenberg, A. T., Wyman, B. L., Yin, J., Zadeh, N. T., 2011. GFDL's CM3 coupled climate model: Characteristics of the ocean and sea ice simulations. *Journal of Climate* 24, 3520–3544.
- Hallberg, R. W., 1997. Stable split time stepping schemes for large-scale ocean modeling. *Journal of Computational Physics* 135, 54–65.
- Hallberg, R. W., 2000. Time integration of diapycnal diffusion and Richardson number-dependent mixing in isopycnal coordinate ocean models. *Monthly Weather Review* 128, 1402–1419.
- Haltiner, G. T., Williams, R. T., 1980. *Numerical Prediction and Dynamic Meteorology*. John Wiley and Sons, New York, USA.
- Hecht, M., Peterson, M., Wingate, B., Hunke, E., Maltrud, M., 2008. Lateral mixing in the eddying regime and a new broad-ranging formulation. In: Hecht, M., Hasumi, H. (Eds.), *Ocean Modeling in an Eddying Regime*. Geophysical Monograph 177. American Geophysical Union, pp. 339–352.
- Held, I. M., Larichev, V. D., 1996. A scaling theory for horizontally homogeneous baroclinically unstable flow on a beta plane. *Journal of Atmospheric Sciences* 53, 946–952.
- Hendershott, M., 1972. The effects of solid earth deformation on global ocean tide. *Geophysical Journal of the Royal Astronomical Society* 29, 389–402.
- Herzfeld, M., Schmidt, M., Griffies, S. M., Liang, Z., 2011. Realistic test cases for limited area ocean modelling. *Ocean Modelling* 37, 1–34.
- Hesselberg, T., 1926. Die Gesetze der ausgeglichenen atmosphärischen Bewegungen. *Beiträger der Physik der freien Atmosphäre* 12, 141–160.
- Hirsch, C., 1988. *Numerical Computation of Internal and External Flows*. John Wiley and Sons.
- Holland, W., Chow, J., Bryan, F., 1998. Application of a third-order upwind scheme in the NCAR ocean model. *Journal of Climate* 11, 1487–1493.
- Holloway, G., 1992. Representing topographic stress for large-scale ocean models. *Journal of Physical Oceanography* 22, 1033–1046.
- Holloway, G., 1999. Moments of probable seas: statistical dynamics of Planet Ocean. *Physica D* 133, 199–214.
- Holloway, G., 2008. Observing global ocean topostrophy. *Journal of Geophysical Research* 113-C07054, doi:10.1029/2007JC004635.

- Holloway, G., Dupont, F., Golubeva, E., Hakkinen, S., Hunke, E., Jin, M., Karcher, M., Kauker, F., Maltrud, M., Morales-Maqueda, M. A., Maslowski, W., Platov, G., Stark, D., Steele, M., Suzuki, T., Wang, J., Zhang, J., 2007. Water properties and circulation in Arctic Ocean models. *Journal of Geophysical Research* 112-C04S03, doi:10.1029/2006JC003642.
- Hsieh, W., Bryan, K., 1996. Redistribution of sea level rise associated with enhanced greenhouse warming: a simple model study. *Climate Dynamics* 12, 535–544.
- Huang, R. X., 1993. Real freshwater flux as a natural boundary condition for the salinity balance and thermohaline circulation forced by evaporation and precipitation. *Journal of Physical Oceanography* 23, 2428–2446.
- Huang, R. X., Jin, X., Zhang, X., 2001. An oceanic general circulation model in pressure coordinates. *Advances in Atmospheric Physics* 18, 1–22.
- Hundsdoerfer, W., Trompert, R., 1994. Method of lines and direct discretization: a comparison for linear advection. *Applied Numerical Mathematics*, 469–490.
- Ilicak, M., Adcroft, A. J., Griffies, S. M., Hallberg, R. W., 2012. Spurious diapycnal mixing and the role of momentum dissipation. *Ocean Modelling* 45–46, 37–58.
- IOC, SCOR, IAPSO, 2010. The international thermodynamic equation of seawater-2010: calculation and use of thermodynamic properties. Intergovernmental Oceanographic Commission, Manuals and Guides No. 56, UNESCO, available from <http://www.TEOS-10.org>, 196pp.
- Iudicone, D., Madec, G., McDougall, T. J., 2008. Water-mass transformations in a neutral density framework and the key role of light penetration. *Journal of Physical Oceanography* 38, 1357–1376.
- Jackett, D. R., McDougall, T. J., 1997. A neutral density variable for the world's oceans. *Journal of Physical Oceanography* 27, 237–263.
- Jackett, D. R., McDougall, T. J., Feistel, R., Wright, D. G., Griffies, S. M., 2006. Algorithms for density, potential temperature, conservative temperature, and freezing temperature of seawater. *Journal of Atmospheric and Oceanic Technology* 23, 1709–1728.
- Jayne, S., 2009. The impact of abyssal mixing parameterizations in an ocean general circulation model. *Journal of Physical Oceanography* 39, 1756–1775.
- Jayne, S., St-Laurent, L. C., 2001. Parameterizing tidal dissipation over rough topography. *Geophysical Research Letters* 28, 811–814.
- Jerlov, N. G., 1968. *Optical Oceanography*. Elsevier.
- Kantha, L. H., Clayson, C. A., 2000. *Small Scale Processes in Geophysical Fluid Flows*. Academic Press, New York, USA, 883 pp.
- Killworth, P., 1989. On the parameterisation of deep convection in ocean models. In: Müller, P., Holloway, G. (Eds.), *Parameterizing small scale processes in the ocean. Proceeding of the 5th 'Aha Huliko'a Hawaiian Winter Workshop*. University of Hawaii at Manoa, pp. 59–74.
- Killworth, P. D., Edwards, N., 1999. A turbulent bottom boundary layer code for use in numerical ocean models. *Journal of Physical Oceanography* 29, 1221–1238.
- Killworth, P. D., Stainforth, D., Webb, D. J., Paterson, S. M., 1991. The development of a free-surface Bryan-Cox-Semtner ocean model. *Journal of Physical Oceanography* 21, 1333–1348.
- Klinger, B. A., Marshall, J., Send, U., 1996. Representation of convective plumes by vertical adjustment. *Journal of Geophysical Research* 101, 18175–18182.
- Klocker, A., McDougall, T. J., 2010a. Influence of the nonlinear equation of state on global estimates of diapycnal advection and diffusion. *Journal of Physical Oceanography* 40, 1690–1709.

- Klocker, A., McDougall, T. J., 2010b. Quantifying the consequences of the ill-defined nature of neutral surfaces. *Journal of Physical Oceanography* 40, 1866–1880.
- Klocker, A., McDougall, T. J., Jackett, D. R., 2009. A new method for forming approximately neutral surfaces. *Ocean Science* 5, 155–172.
- Klymak, J., Moum, J., Nash, J., Kunze, E., Girton, J., Carter, G., Lee, C., Sanford, T., Gregg, M., 2005. An estimate of tidal energy lost to turbulence at the hawaiian ridge. *Journal of Physical Oceanography* submitted.
- Kopp, R. E., Mitrovica, J. X., Griffies, S. M., Yin, J., Hay, C. C., Stouffer, R. J., 2010. The impact of Greenland melt on regional sea level: a preliminary comparison of dynamic and static equilibrium effects. *Climatic Change Letters* 103, 619–625.
- Kwon, E., Downes, S., Sarmiento, J., Farneti, R., Deutsch, C., 2013. Role of the seasonal cycle in the subduction rates of Upper-Southern Ocean mode waters. *Journal of Physical Oceanography* 43, 1096–1113.
- Lamport, L., 1994. \LaTeX : A Documentation Preparation System User's Guide and Reference Manual. Addison-Wesley, Reading, Massachusetts, 272 pp.
- Landau, L. D., Lifshitz, E. M., 1976. *Mechanics*. Pergamon Press, Oxford, UK, 170 pp.
- Landau, L. D., Lifshitz, E. M., 1987. *Fluid Mechanics*. Pergamon Press, Oxford, UK, 539 pp.
- Landerer, F., Jungclauss, J., Marotzke, J., 2007. Regional dynamic and steric sea level change in response to the IPCC-A1B Scenario. *Journal of Physical Oceanography* 37, 296–312.
- Large, W., 1998. Modeling the oceanic boundary layer. In: Chassignet, E. P., Verron, J. (Eds.), *Ocean Modeling and Parameterization*. Vol. 516 of NATO ASI Mathematical and Physical Sciences Series. Kluwer, pp. 81–120.
- Large, W., Gent, P., 1999. Validation of vertical mixing in an equatorial ocean model using large eddy simulations and observations. *Journal of Physical Oceanography* 29, 449–464.
- Large, W., McWilliams, J., Doney, S., 1994. Oceanic vertical mixing: a review and a model with a nonlocal boundary layer parameterization. *Reviews of Geophysics* 32, 363–403.
- Large, W. B., Nurser, A. G., 2001. Ocean surface water mass transformation. In: Seidler, G., Church, J., Gould, J. (Eds.), *Ocean Circulation and Climate*. Vol. 77 of International Geophysics Series. Academic Press, San Diego, pp. 317–336.
- Large, W. G., Danabasoglu, G., Doney, S. C., McWilliams, J. C., 1997. Sensitivity to surface forcing and boundary layer mixing in a global ocean model: annual-mean climatology. *Journal of Physical Oceanography* 27, 2418–2447.
- Large, W. G., Danabasoglu, G., McWilliams, J. C., Gent, P. R., Bryan, F. O., 2001. Equatorial circulation of a global ocean climate model with anisotropic horizontal viscosity. *Journal of Physical Oceanography* 31, 518–536.
- Leclair, M., Madec, G., 2009. A conservative leapfrog time-stepping method. *Ocean Modelling* 30, 88–94.
- Ledwell, J. R., St-Laurent, L., Girton, J., Toole, J., 2011. Diapycnal mixing in the Antarctic Circumpolar Current. *Journal of Physical Oceanography* 41, 241–246.
- Ledwell, J. R., Watson, A. J., Law, C. S., 1993. Evidence for slow mixing across the pycnocline from an open-ocean tracer-release experiment. *Nature* 364, 701–703.
- Lee, H.-C., Rosati, A., Spelman, M., 2006. Barotropic tidal mixing effects in a coupled climate model: Oceanic conditions in the northern Atlantic. *Ocean Modelling* 3-4, 464–477.

- Lemarié, F., Debreu, L., Shchepetkin, A. F., McWilliams, J. C., 2012a. On the stability and accuracy of the harmonic and biharmonic isoneutral mixing operators in ocean models. *Ocean Modelling* 52-53, 9–35.
- Lemarié, F., Kurian, J., Shchepetkin, A. F., Molemaker, M. J., Colas, F., McWilliams, J. C., 2012b. Are there inescapable issues prohibiting the use of terrain-following coordinates in climate models? *Ocean Modelling* 42, 57–79.
- Leonard, B. P., 1979. A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* 19, 59–98.
- Levitus, S., 1982. Climatological atlas of the world ocean. U.S. Government Printing Office 13, NOAA, Washington, D.C., 163 pp.
- Li, X., Chao, Y., McWilliams, J. C., Fu, L. L., 2001. A comparison of two vertical-mixing schemes in a Pacific ocean general circulation model. *Journal of Climate* 14, 1377–1398.
- Lin, S. J., 1997. A finite volume integration method for computing pressure gradient force in general vertical coordinates. *Quarterly Journal of the Royal Meteorological Society* 123, 1749–1762.
- Lorbacher, K., Marsland, S. J., Church, J. A., Griffies, S. M., Stammer, D., 2012. Rapid barotropic sea-level rise from ice-sheet melting scenarios. *Journal of Geophysical Research* 117, C06003.
- Losch, M., Adcroft, A., Campin, J.-M., 2004. How sensitive are coarse general circulation models to fundamental approximations in the equations of motion? *Journal of Physical Oceanography* 34, 306–319.
- Lowe, J. A., Gregory, J. M., 2006. Understanding projections of sea level rise in a hadley centre coupled climate model. *Journal of Geophysical Research: Oceans* 111 (C11), n/a–n/a.
URL <http://dx.doi.org/10.1029/2005JC003421>
- Machenhauer, B., Kaas, E., Lauritzen, P., 2009. Finite-volume methods in meteorology. In: Temam, R., Tribbia, J. (Eds.), *Computational Methods for the Atmosphere and the Oceans*. Elsevier, Amsterdam, p. 761.
- Madec, G., Imbard, M., 1996. A global ocean mesh to overcome the North Pole singularity. *CD 12*, 381–388.
- Maltrud, M., Holloway, G., 2008. Implementing biharmonic neptune in a global eddying ocean model. *Ocean Modelling* 21, 22–34.
- Manizza, M., Le Quere, C., Watson, A., Buitenhuis, E., 2005. Bio-optical feedbacks among phytoplankton, upper ocean physics and sea-ice in a global model. *Geophysical Research Letters* 32, doi:10.1029/2004GL020778.
- Marchesiello, J. M. P., Shchepetkin, A., 2001. Open boundary conditions for long-term integration of regional oceanic models. *Ocean Modelling* 3, 1–20.
- Marchuk, Kagan, 1989. *Dynamics of Ocean Tides*. Kluwer Academic.
- Marion, J. B., Thornton, S. T., 1988. *Classical Dynamics of Particles and Systems*. Harcourt Brace Jovanovich, San Diego, USA, 602 pp.
- Marotzke, J., 1991. Influence of convective adjustment on the stability of the thermohaline circulation. *Journal of Physical Oceanography* 21, 903–907.
- Marshall, J., Adcroft, A., Campin, J.-M., Hill, C., White, A., 2004. Atmosphere-ocean modeling exploiting fluid isomorphisms. *Monthly Weather Review* 132, 2882–2894.
- Marshall, J., Hill, C., Perelman, L., Adcroft, A., 1997. Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling. *Journal of Geophysical Research* 102, 5733–5752.
- Marshall, J., Jamous, D., Nilsson, J., 1999. Reconciling thermodynamic and dynamic methods of computation of water-mass transformation rates. *Deep-Sea Research I* 46, 545–572.

- Marshall, J., Schott, F., 1999. Open-ocean convection: observations, theory, and models. *Reviews of Geophysics* 37, 1–64.
- Martinsen, E., Engedahl, H., 1987. Implementation and testing of a lateral boundary scheme as an open boundary condition in a barotropic ocean model. *Coastal Engineering* 11, 603–627.
- Maze, G., Forget, G., Buckley, M., Marshall, J., Cerovecki, I., 2009. Using transformation and formation maps to study the role of air-sea heat fluxes in the North Atlantic eighteen degree water formation. *Journal of Physical Oceanography* 39, 1818–1835.
- McDougall, T. J., 1987a. Neutral surfaces. *Journal of Physical Oceanography* 17, 1950–1967.
- McDougall, T. J., 1987b. Thermobaricity, cabbeling, and water-mass conversion. *Journal of Geophysical Research* 92, 5448–5464.
- McDougall, T. J., 1995. The influence of ocean mixing on the absolute velocity vector. *Journal of Physical Oceanography* 25, 705–725.
- McDougall, T. J., 2003. Potential enthalpy: a conservative oceanic variable for evaluating heat content and heat fluxes. *Journal of Physical Oceanography* 33, 945–963.
- McDougall, T. J., Barker, P., Feistel, R., Jackett, D., 2012. A computationally efficient 48-term expression for the density of seawater in terms of conservative temperature, and related properties of seawater. *Ocean Science* in preparation.
- McDougall, T. J., Church, J. A., 1986. Pitfalls with numerical representations of isopycnal and diapycnal mixing. *Journal of Physical Oceanography* 16, 196–199.
- McDougall, T. J., Greatbatch, R., Lu, Y., 2002. On conservation equations in oceanography: How accurate are Boussinesq ocean models? *Journal of Physical Oceanography* 32, 1574–1584.
- McDougall, T. J., Jackett, D. R., 1988. On the helical nature of neutral trajectories in the ocean. *Progress in Oceanography* 20, 153–183.
- McDougall, T. J., Jackett, D. R., 2005. The material derivative of neutral density. *Journal of Marine Research* 63, 159–185.
- McDougall, T. J., Jackett, D. R., Wright, D. G., Feistel, R., 2003. Accurate and computationally efficient algorithms for potential temperature and density of seawater. *Journal of Atmospheric and Oceanic Technology* 20, 730–741.
- McDougall, T. J., McIntosh, P. C., 2001. The temporal-residual-mean velocity. Part II: isopycnal interpretation and the tracer and momentum equations. *Journal of Physical Oceanography* 31, 1222–1246.
- McWilliams, J. C., Sullivan, P., 2001. Vertical mixing by Langmuir circulations. *Spill Science and Technology Bulletin* 6, 225–237.
- Merryfield, W. J., Holloway, G., 2003. Application of an accurate advection algorithm to sea-ice modelling. *Ocean Modelling* 5, 1–15.
- Merryfield, W. J., Scott, R., 2007. Bathymetric influence on mean currents in two high resolution near-global ocean models. *Ocean Modelling* 16, 76–94.
- Mesinger, F., 1973. A method for construction of second-order accurate difference schemes permitting no false two-grid-interval waves in the height field. *Tellus* 25, 444–457.
- Mesinger, F., Arakawa, A., 1976. Numerical methods used in atmospheric models. In: *GARP Publication Series*. Vol. 1. p. 66 pages.
- Miller, M., Thorpe, A., 1981. Radiation conditions for the lateral boundaries of limited-area numerical models. *Quarterly Journal of the Royal Meteorological Society* 107, 615–628.

- Mitrovica, J. X., Tamisiea, M. E., Davis, J. L., Milne, G. A., 2001. Recent mass balance of polar ice sheets inferred from patterns of global sea-level change. *Nature* 409, 1026–1029.
- Morel, A., 1988. Optical modeling of the upper ocean in relation to its biogenous matter content (case-i waters). *JGR* 93, 10749–10768.
- Morel, A., Antoine, D., 1994. Heating rate within the upper ocean in relation to its bio-optical state. *Journal of Physical Oceanography* 24, 1652–1665.
- Morel, A., Berthon, J.-F., 1989. Surface pigments, algal biomass profiles, and potential production of the euphotic layer: relationship reinvestigated in view of remote-sensing applications. *Limnology and Oceanography* 34, 1542–1562.
- Morse, P. M., Feshbach, H., 1953. *Methods of Theoretical Physics Part I and II*. McGraw-Hill Book Company, New York.
- Moum, J., Caldwell, D., Nash, J., Gunderson, G., 2002. Observations of boundary mixing over the continental slope. *Journal of Physical Oceanography* 32, 2113–2130.
- Müller, P., 2006. *The Equations of Oceanic Motions*, 1st Edition. Cambridge University Press, Cambridge, 302pp.
- Munk, W., Anderson, E., 1948. Notes on a theory of the thermocline. *Journal of Marine Research* 3, 276–295.
- Murray, R., 1996. Explicit generation of orthogonal grids for ocean models. *Journal of Computational Physics* 126, 251–273.
- Naveira-Garabato, A., Polzin, K., King, B., Heywood, K., Visbeck, M., 2004. Widespread intense turbulent mixing in the Southern Ocean. *Science* 303, 210–213.
- Nicholls, R., Cazenave, A., 2010. Sea-level rise and its impact on coastal zones. *Science* 328, 1517–1520.
- Nikurashin, M., Ferrari, R., 2010. Radiation and dissipation of internal waves generated by geostrophic motions impinging on small-scale topography: Application to the Southern Ocean. *Journal of Physical Oceanography* 40, 2025–2042.
- O'Brien, J. J., 1986. *Advanced Physical Oceanographic Numerical Modelling*. D. Reidel Publishing Company.
- Ohlmann, J., 2003. Ocean radiant heating in climate models. *JOC* 16, 1337–1351.
- Ohlmann, J., Siegel, D., 2000. Ocean radiant heating. Part i: Optical influences. *JPO* 30, 1833–1848.
- Olbers, D. J., Wenzel, M., Willebrand, J., 1985. The inference of North Atlantic circulation patterns from climatological hydrographic data. *Reviews of Geophysics* 23, 313–356.
- Orlanski, I., 1976. A simple boundary condition for unbounded hyperbolic flows. *Journal of Computational Physics* 21, 251–269.
- Osborn, T. R., 1980. Estimates of the local rate of vertical diffusion from dissipation measurements. *Journal of Physical Oceanography* 10, 83–89.
- Pacanowski, R. C., 1987. Effect of equatorial currents on surface stress. *Journal of Physical Oceanography* 17, 833–838.
- Pacanowski, R. C., 1995. *MOM2 Documentation, User's Guide, and Reference Manual*. NOAA/Geophysical Fluid Dynamics Laboratory, Princeton, USA, 216 pp.
- Pacanowski, R. C., Dixon, K., Rosati, A., 1991. *The GFDL Modular Ocean Model User Guide*. NOAA/Geophysical Fluid Dynamics Laboratory, Princeton, USA, 16 pp.

- Pacanowski, R. C., Gnanadesikan, A., 1998. Transient response in a z-level ocean model that resolves topography with partial-cells. *Monthly Weather Review* 126, 3248–3270.
- Pacanowski, R. C., Griffies, S. M., 1999. The MOM3 Manual. NOAA/Geophysical Fluid Dynamics Laboratory, Princeton, USA, 680 pp.
- Pacanowski, R. C., Philander, G., 1981. Parameterization of vertical mixing in numerical models of the tropical ocean. *Journal of Physical Oceanography* 11, 1442–1451.
- Palma, E., Matano, R., 1998. On the implementation of passive open boundary conditions for a general circulation model: The barotropic mode. *Journal of Geophysical Research* 103(C1), 1319–1341.
- Palma, E., Matano, R., 2001. Dynamical impacts associated with radiation boundary conditions. *Journal of Sea Research* 46, 117–132.
- Paulson, E. A., Simpson, J. J., 1977. Irradiance measurements in the upper ocean. *Journal of Physical Oceanography* 7, 952–956.
- Peixoto, J. P., Oort, A. H., 1992. *Physics of Climate*. American Institute of Physics, 520 + xxxix pp.
- Phillips, N., 1973. Principles of large-scale numerical weather prediction. In: Morel, P. (Ed.), *Dynamic Meteorology*. Riedel, pp. 1–96.
- Pickard, G. L., Emery, W. J., 1990. *Descriptive Physical Oceanography*, 5th Edition. Pergamon Press, Oxford.
- Polzin, K. L., Toole, J. M., Ledwell, J. R., Schmitt, R. W., 1997. Spatial variability of turbulent mixing in the abyssal ocean. *Science* 276, 93–96.
- Prather, M., 1986. Numerical advection by conservation of second-order moments. *Journal of Geophysical Research* 91, 6671–6681.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P., 1992. *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd Edition. Cambridge University Press, Cambridge, 963 + xxvi pp.
- Rahmstorf, S., 1993. A fast and complete convection scheme for ocean models. *Ocean Modelling* 101, 9–11.
- Ray, R. D., 1998. Ocean self-attraction and loading in numerical tidal models. *Marine Geodesy* 21, 181–192.
- Raymond, W., Kuo, H., 1984. A radiation boundary condition for multidimensional flows. *Quarterly Journal of the Royal Meteorological Society* 110, 535–551.
- Redi, M. H., 1982. Oceanic isopycnal mixing by coordinate rotation. *Journal of Physical Oceanography* 12, 1154–1158.
- Robert, A., 1966. The integration of a low order spectral form of the primitive meteorological equations. *Journal of the Meteorological Society of Japan* 44, 237–244.
- Robinson, I., 2005. Satellite oceanography. In: Chassignet, E. P., Verron, J. (Eds.), *GODEA Summer School*. Springer/Kluwer.
- Roed, L., Cooper, C., 1987. A study of various open boundary conditions for wind-forced barotropic numerical ocean models. In: Nihoul, J., Jamart, B. (Eds.), *Three-dimensional Models of marine and Estuarine Dynamics*. Elsevier, pp. 305–335.
- Rosati, A., Miyakoda, K., 1988. A general circulation model for upper ocean simulation. *Journal of Physical Oceanography* 18, 1601–1626.
- Sadourny, R., 1975. The dynamics of finite-difference models of the shallow-water equations. *Journal of Atmospheric Sciences* 32, 680–689.

- Schmitt, R. W., 1994. Double diffusion in oceanography. *Annual Review of Fluid Mechanics* 26, 255–285.
- Shchepetkin, A., McWilliams, J., 2002. A method for computing horizontal pressure-gradient force in an ocean model with a non-aligned vertical coordinate. *Journal of Geophysical Research* 108, 35.1–35.34.
- Shchepetkin, A., McWilliams, J., 2005. The regional oceanic modeling system (ROMS): a split-explicit, free-surface, topography-following-coordinate oceanic model. *Ocean Modelling* 9, 347–404.
- Shchepetkin, A. F., 2005. If-less kpp. ROMS/TOMS Workshop: Adjoint Modeling and Applications.
- Simmons, H. L., Jayne, S. R., St.Laurent, L. C., Weaver, A. J., 2004. Tidally driven mixing in a numerical model of the ocean general circulation. *Ocean Modelling* 6, 245–263.
- Smagorinsky, J., 1963. General circulation experiments with the primitive equations: I. The basic experiment. *Monthly Weather Review* 91, 99–164.
- Smagorinsky, J., 1993. Some historical remarks on the use of nonlinear viscosities. In: Galperin, B., Orszag, S. A. (Eds.), *Large Eddy Simulation of Complex Engineering and Geophysical Flows*. Cambridge University Press, pp. 3–36.
- Smith, R. D., Kortas, S., Meltz, B., 1995. Curvilinear coordinates for global ocean models. Los Alamos preprint LA-UR-95-1146.
- Smith, R. D., McWilliams, J. C., 2003. Anisotropic horizontal viscosity for ocean models. *Ocean Modelling* 5, 129–156.
- Smyth, G. C. W. D., Skillingstad, E. D., Wijesekera, H., 2002. Nonlocal fluxes and Stokes drift effects in the K-profile parameterization. *Ocean Dynamics* 52, 104–115.
- Solomon, H., 1971. On the representation of isentropic mixing in ocean models. *Journal of Physical Oceanography* 1, 233–234.
- Sommerfeld, A., 1949. Partial differential equations. *Lectures in Theoretical Physics* 6. Academic Press, New York.
- Speer, K., Tziperman, E., 1992. Rates of water mass formation in the North Atlantic Ocean. *Journal of Physical Oceanography* 22, 2444–2457.
- Stacey, M. W., Pond, S., Nowak, Z. P., 1995. A numerical model of the circulation in Knight Inlet, British Columbia, Canada. *Journal of Physical Oceanography* 25, 1037–1062.
- Stammer, D., 2008. Response of the global ocean to Greenland and Antarctic ice melting. *Journal of Geophysical Research* 113, doi:10.1029/2006JC004079.
- Starr, V. P., 1945. A quasi-Lagrangian system of hydrodynamical equations. *Journal of Meteorology* 2, 227–237.
- Stevens, D. P., 1990. On open boundary conditions for three dimensional primitive equation ocean circulation models. *Geophysical and Astrophysical Fluid Dynamics* 51, 103–133.
- Stevens, D. P., 1991. The open boundary conditions in the united kingdom fine-resolution antarctic model. *Journal of Physical Oceanography* 21, 1494–1499.
- St.Laurent, L. C., Simmons, H., Jayne, S., 2002. Estimating tidally driven energy in the deep ocean. *Geophysical Research Letters* 29, 2106–2110.
- St.Laurent, L. C., Stringer, S., Garrett, C., Perrault-Joncas, D., 2003. The generation of internal tides at abrupt topography. *Deep-Sea Research* 50, 987–1003.
- St.Laurent, L. C., Toole, J., Schmitt, R., 2001. Buoyancy forcing by turbulence above rough topography in the abyssal brazil basin. *Journal of Physical Oceanography* 31, 3476–3495.

- Stouffer, R. J., Broccoli, A., Delworth, T., Dixon, K., Gudgel, R., Held, I., Hemler, R., Knutson, T., Lee, H.-C., Schwarzkopf, M., Soden, B., Spelman, M., Winton, M., Zeng, F., 2006a. GFDL's CM2 global coupled climate models: Part 4: idealized climate response. *Journal of Climate* 19, 723–740.
- Stouffer, R. J., Dixon, K., Spelman, M., Hurlin, W., Yin, J., Gregory, J., Weaver, A., Eby, M., Flato, G., Robitaille, D., Hasumi, H., Oka, A., Hu, A., Jungclauss, J., Kamenkovich, I., Levermann, A., Montoya, M., Murakami, S., Nawrath, S., Peltier, W., Vettoretti, G., Sokolov, A., Weber, S., 2006b. Investigating the causes of the response of the thermohaline circulation to past and future climate changes. *Journal of Climate* 19, 1365–1387.
- Sullivan, P. P., McWilliams, J. C., 2010. Dynamics of winds and currents coupled to surface waves. *Annual Review of Fluid Mechanics* 42, 19–42.
- Sweby, P., 1984. High-resolution schemes using flux limiters for hyperbolic conservation-laws. *SIAM Journal of Numerical Analysis* 21, 995–1011.
- Sweeney, C., Gnanadesikan, A., Griffies, S. M., Harrison, M., Rosati, A., Samuels, B., 2005. Impacts of shortwave penetration depth on large-scale ocean circulation and heat transport. *Journal of Physical Oceanography* 35, 1103–1119.
- Tang, Y., Grimshaw, R., 1996. Radiation boundary conditions in barotropic coastal ocean numerical models. *Journal of Computational Physics* 123, 96–110.
- Tang, Y., Roberts, M., 2005. The impact of a bottom boundary layer scheme on the North Atlantic Ocean in a global coupled climate model. *Journal of Physical Oceanography* 35, 202–217.
- Tennekes, H., 1973. The logarithmic wind profile. *Journal of Atmospheric Sciences* 30, 558–567.
- Terray, E., Donelan, M., Agrawal, Y., Drennan, W., Kahma, K., Williams, A., Hwang, P., Kitaigorodskii, S., 1996. Estimates of kinetic energy dissipation under breaking waves. *Journal of Physical Oceanography* 26, 792–807.
- Thiele, G., Sarmiento, J. L., 1990. Tracer dating and ocean ventilation. *Journal of Geophysical Research* 95, 9377–9391.
- Tomczak, M., Godfrey, J. S., 1994. *Regional Oceanography: An Introduction*. Pergamon Press, Oxford, England, 422 + vii pp.
- Treguier, A. M., Held, I. M., Larichev, V. D., 1997. On the parameterization of quasi-geostrophic eddies in primitive equation ocean models. *Journal of Physical Oceanography* 27, 567–580.
- Troen, I. B., Mahrt, L., 1986. A simple model of the atmospheric boundary layer: sensitivity to surface evaporation. *Boundary Layer Meteorology* 37, 129–148.
- Tziperman, E., 1986. On the role of interior mixing and air-sea fluxes in determining the stratification and circulation in the oceans. *Journal of Physical Oceanography* 16, 680–693.
- Umlauf, L., Burchard, H., Bolding, K., 2005. GOTM: source code and test case documentation: version 3.2. 231pp.
- Veronis, G., 1973. Large scale ocean circulation. *Advances in Applied Mechanics* 13, 2–92.
- Visbeck, M., Marshall, J. C., Haine, T., Spall, M., 1997. Specification of eddy transfer coefficients in coarse resolution ocean circulation models. *Journal of Physical Oceanography* 27, 381–402.
- Viúdez, A., 2000. Volume and mass transport across isosurfaces of a balanced fluid property. *Journal of Physical Oceanography* 30, 1478–1485.
- Wahr, J., 1998. Body tides on an elliptical, rotating, elastic and oceanless earth. *Geophysical Journal of the Royal Astronomical Society* 64, 677–703.

- Walín, G., 1982. On the relation between sea-surface heat flow and thermal circulation in the ocean. *Tellus* 34, 187–195.
- Warren, B., 2009. Note on the vertical velocity and diffusive salt flux induced by evaporation and precipitation. *Journal of Physical Oceanography* 39, 2680–2682.
- Webb, D. J., Coward, A. C., de Cuevas, B. A., Gwilliam, C. S., 1998. The first main run of the OCCAM global ocean model. Internal Document No. 34. Southampton Oceanography Centre, Southampton, England.
- Winters, K. B., D’Asaro, E. A., 1995. Diascalar flux and the rate of fluid mixing. *Journal of Fluid Mechanics* 317, 179–193.
- Winters, K. B., Lombard, P. N., Riley, J. J., D’Asaro, E. A., 1995. Available potential energy and mixing in density-stratified fluids. *Journal of Fluid Mechanics* 289, 115–128.
- Winton, M., Hallberg, R., Gnanadesikan, A., 1998. Simulation of density-driven frictional downslope flow in z-coordinate ocean models. *Journal of Physical Oceanography* 28, 2163–2174.
- Wittenberg, A., Rosati, A., Lau, G., Ploshay, J., 2006. GFDL’s CM2 global coupled climate models - Part 3: Tropical Pacific climate and ENSO. *Journal of Climate* 19, 698–722.
- Wright, D. G., 1997. An equation of state for use in ocean models: Eckart’s formula revisited. *Journal of Atmospheric and Oceanic Technology* 14, 735–740.
- Yin, J., Griffies, S. M., Stouffer, R., 2010a. Spatial variability of sea-level rise in 21st century projections. *Journal of Climate* 23, 4585–4607.
- Yin, J., Schlesinger, M., Stouffer, R., 2009. Model projections of rapid sea-level rise on the northeast coast of the United States. *Nature Geosciences* 2, 262–266.
- Yin, J., Stouffer, R., Spelman, M. J., Griffies, S. M., 2010b. Evaluating the uncertainty induced by the virtual salt flux assumption in climate simulations and future projections. *Journal of Climate* 23, 80–96.
- Zhao, R., Vallis, G. K., 2008. Parameterizing mesoscale eddies with residual and eulerian schemes, and a comparison with eddy-permitting models. *Ocean Modelling* 23, 1–12.
- Zika, J., McDougall, T., Sloyan, B., 2010. A tracer-contour inverse method for estimating ocean circulation and mixing. *Journal of Physical Oceanography* 40, 26–47.

Index

- Adcroft and Campin (2004), [8](#), [67](#), [89](#), [613](#)
Adcroft and Hallberg (2006), [136](#), [613](#)
Adcroft et al. (1997), [6](#), [44](#), [87](#), [88](#), [613](#)
Adcroft et al. (1999), [117](#), [179](#), [411](#), [613](#)
Adcroft et al. (2001), [566](#), [613](#)
Adcroft et al. (2008), [54](#), [57](#), [67](#), [613](#)
Aiki et al. (2004), [16](#), [297](#), [298](#), [613](#)
Apel (1987), [40](#), [613](#)
Arbic et al. (2004), [105](#), [613](#)
Aris (1962), [480](#), [613](#)
Asselin (1972), [163](#), [613](#)
Bacon and Fofonoff (1996), [40](#), [613](#)
Batchelor (1967), [480](#), [555](#), [613](#)
Bates et al. (2012a), [ii](#), [3](#), [11](#), [19](#), [21](#), [613](#)
Bates et al. (2012b), [ii](#), [3](#), [11](#), [19](#), [21](#), [613](#)
Bates (2011), [21](#), [613](#)
Beckmann and Döscher (1997), [380–383](#), [385](#), [389](#), [521](#), [614](#)
Black (1994), [8](#), [89](#), [614](#)
Bleck (1978), [575](#), [614](#)
Blumberg and Kantha (1985), [233](#), [614](#)
Blumberg and Mellor (1987), [9](#), [218](#), [614](#)
Bryan and Cox (1972), [98](#), [614](#)
Bryan and Lewis (1979), [16](#), [287](#), [595](#), [614](#)
Bryan (1963), [iii](#), [614](#)
Bryan (1969), [57](#), [58](#), [65](#), [115](#), [137](#), [161](#), [178](#), [215](#), [222](#), [283](#), [425](#), [614](#)
Bryan (1989), [179](#), [614](#)
Bryan (1991), [2](#), [614](#)
Bryan (1996), [563](#), [614](#)
Burchard and Rennau (2008), [449](#), [451](#), [453](#), [614](#)
Callen (1985), [34](#), [614](#)
Camerlengo and O'Brien (1980), [235](#), [614](#)
Campin and Goosse (1999), [381–383](#), [385–391](#), [614](#)
Campin et al. (2004), [86](#), [161](#), [614](#)
Chaikin and Lubensky (1995), [37](#), [553](#), [614](#)
Chang et al. (2005), [257](#), [614](#)
Chapman (1985), [233–235](#), [615](#)
Chassignet and Garraffo (2001), [356](#), [615](#)
Chassignet and Verron (2005), [26](#), [615](#)
Chen et al. (1994), [97](#), [615](#)
Church and Gregory (2001), [573](#), [615](#)
Conkright et al. (2002), [541](#), [615](#)
Cox (1984), [iii](#), [2](#), [163](#), [283](#), [284](#), [615](#)
Cox (1987), [574](#), [615](#)
Craig and Banner (1994), [269](#), [615](#)
Cushman-Roisin (1987), [543](#), [547](#), [615](#)
Danabasoglu and McWilliams (1995), [322](#), [615](#)
Danabasoglu et al. (2006), [16](#), [615](#)
Danabasoglu et al. (2008), [333](#), [615](#)
Danabasoglu et al. (2010), [520](#), [615](#)
DeGroot and Mazur (1984), [37](#), [553](#), [615](#)
DeSzoeke and Bennett (1993), [558](#), [616](#)
DeSzoeke and Samelson (2002), [9](#), [92](#), [616](#)
Delworth et al. (2006), [ii](#), [3](#), [4](#), [8](#), [13](#), [21](#), [143](#), [293](#), [539](#), [597](#), [599](#), [615](#)
Delworth et al. (2012), [ii](#), [21](#), [356](#), [615](#)
Denman (1973), [251](#), [615](#)
Deremble et al. (2012), [357](#), [371](#), [616](#)
Diansky et al. (2002), [66](#), [90](#), [616](#)
Donner et al. (2011), [8](#), [616](#)
Downes et al. (2011), [458](#), [616](#)
Dunne et al. (2012), [5](#), [8](#), [13](#), [21](#), [123](#), [143](#), [293](#), [356](#), [523](#), [533](#), [539](#), [616](#)
Dunne et al. (2013), [5](#), [8](#), [13](#), [14](#), [21](#), [356](#), [523](#), [533](#), [539](#), [616](#)
Durran (1999), [161–163](#), [179](#), [180](#), [182](#), [194](#), [213](#), [616](#)
Durski et al. (2004), [257](#), [259](#), [616](#)
Döscher and Beckmann (2000), [380–383](#), [616](#)
Eby and Holloway (1994), [374](#), [375](#), [616](#)
Eden et al. (2007), [558](#), [616](#)
Eden et al. (2009), [341](#), [616](#)
Emile-Geay and Madec (2009), [566](#), [616](#)
England (1995), [52](#), [616](#)
Ezer et al. (2002), [90](#), [617](#)
Farneti et al. (2010), [338](#), [356](#), [617](#)
Farrell and Clark (1976), [103](#), [105](#), [617](#)
Feistel and Hagen (1995), [98](#), [99](#), [617](#)
Feistel (1993), [98](#), [99](#), [617](#)
Feistel (2003), [98](#), [99](#), [617](#)
Ferrari et al. (2008), [79](#), [333](#), [353](#), [468](#), [493](#), [516](#), [585](#), [617](#)

- Ferrari et al. (2010), 79, 306, 323–326, 353, 468, 497, 516, 585, 587, 617
- Ferreira and Marshall (2006), 16, 297, 298, 617
- Fofonoff (1962), 40, 617
- Fox-Kemper et al. (2008a), 347, 421, 617
- Fox-Kemper et al. (2008b), 16, 347–349, 353, 421, 426, 460, 470, 471, 497, 517, 531, 532, 557, 588, 589, 595, 617
- Fox-Kemper et al. (2011), 16, 347, 352, 421, 422, 617
- Galbraith et al. (2011), 14, 617
- Gargett (1984), 296, 617
- Gent and McWilliams (1990), 79, 80, 83, 84, 297, 305, 306, 321, 324, 347, 420, 469, 483, 496, 556, 568, 574, 575, 588, 617
- Gent et al. (1995), xiii, 80, 81, 297, 324, 347, 349–351, 391, 420, 422, 426, 460, 470, 471, 496, 531, 532, 556, 557, 578, 579, 587, 595, 617
- Gent et al. (1998), 256, 617
- Gill and Niler (1973), 562, 617
- Gill (1982), 62, 67, 98, 103–105, 109, 264, 617
- Gnanadesikan et al. (2006), ii, 3, 4, 8, 13, 21, 143, 293, 597, 618
- Goosens et al. (1994), ii, 618
- Gordon et al. (2000), 340, 618
- Greatbatch and Lamb (1990), 16, 297, 391, 618
- Greatbatch and McDougall (2003), 85, 94, 558, 618
- Greatbatch and Mellor (1999), 90, 618
- Greatbatch et al. (2001), 156, 618
- Greatbatch (1994), 563–565, 618
- Gregg et al. (2003), 479, 574, 618
- Griffies and Adcroft (2008), 26, 105, 107, 465, 480, 481, 543, 546, 618
- Griffies and Greatbatch (2012), 552, 580, 592, 619
- Griffies and Hallberg (2000), 15, 77, 357, 619
- Griffies et al. (1998), 98, 305, 349, 478, 574, 575, 618
- Griffies et al. (2000a), 8, 58, 66, 90, 116, 161, 166, 193, 579, 618
- Griffies et al. (2000b), 5, 86, 162, 430, 433, 440, 493, 530, 619
- Griffies et al. (2001), 10, 52, 86, 150, 157, 160–164, 166, 172, 196, 222, 264, 411, 434, 523, 619
- Griffies et al. (2004), 2, 7, 21, 85, 87, 97, 115, 177, 183, 196, 215, 221, 388, 390, 397, 404, 412, 415, 505, 619
- Griffies et al. (2005), ii, 3, 4, 7, 8, 11, 13, 15, 16, 21, 87, 143, 160, 174, 293, 357, 480, 521, 522, 539, 597, 618
- Griffies et al. (2009), 262, 504, 618
- Griffies et al. (2011), 8, 21, 619
- Griffies (1998), 98, 297, 298, 303, 304, 306, 324, 349, 422, 470, 483, 497, 618
- Griffies (2004), 2, 10, 11, 21, 26, 27, 30, 33, 34, 36, 39–42, 46, 48, 59, 62, 75, 77, 79, 81, 83, 89, 92, 94, 106, 115, 117, 157, 160–162, 164, 167, 174, 180, 182, 191, 192, 194, 212, 215, 221, 222, 304, 305, 309, 319–322, 324, 328–332, 337–341, 356–361, 366, 367, 369, 370, 395, 411, 412, 420, 465, 467, 469, 478–481, 484, 527, 543, 544, 546, 547, 555, 558, 564, 566, 567, 574, 575, 578, 579, 604, 605, 618
- Griffies (2005), 26, 85, 618
- Griffies (2009), 21, 618
- Hallberg (1997), 67, 161, 165, 619
- Hallberg (2000), 76, 77, 619
- Haltiner and Williams (1980), 161, 619
- Hecht et al. (2008), 356, 619
- Held and Larichev (1996), 337, 619
- Hendershott (1972), 110, 619
- Herzfeld et al. (2011), 3, 4, 17, 21, 126, 230, 619
- Hesselberg (1926), 558, 619
- Hirsch (1988), 44, 54, 619
- Holland et al. (1998), 13, 162, 256, 619
- Holloway et al. (2007), 375, 619
- Holloway (1992), 77, 356, 357, 373, 374, 619
- Holloway (1999), 61, 78, 619
- Holloway (2008), 375, 619
- Hsieh and Bryan (1996), 563, 620
- Huang et al. (2001), 9, 92, 94, 620
- Huang (1993), 264, 620
- Hundsdoerfer and Trompert (1994), 13, 620
- Ilicak et al. (2012), 5, 493, 620
- Iudicone et al. (2008), 458–460, 473, 474, 480, 539, 565, 620
- Jackett and McDougall (1997), 459, 465, 620
- Jackett et al. (2006), 12, 86, 97–100, 620
- Jayne and St.Laurent (2001), 288, 620
- Jayne (2009), 291, 620
- Jerlov (1968), 250, 620
- Kantha and Clayson (2000), 262, 267, 268, 620
- Killworth and Edwards (1999), 380, 389, 620
- Killworth et al. (1991), 10, 157, 161, 164–166, 411, 523, 620
- Killworth (1989), 285, 620
- Klinger et al. (1996), 15, 283, 620
- Klocker and McDougall (2010a), 574, 620
- Klocker and McDougall (2010b), 574, 620
- Klocker et al. (2009), 465, 537, 621
- Klymak et al. (2005), 290, 621
- Kopp et al. (2010), 9, 105, 621
- Kwon et al. (2013), 543, 549, 621
- Lamport (1994), ii, 621
- Landau and Lifshitz (1976), 26, 621

- Landau and Lifshitz (1987), [553](#), [555](#), [621](#)
- Landerer et al. (2007), [562](#), [563](#), [621](#)
- Large and Gent (1999), [282](#), [621](#)
- Large and Nurser (2001), [458](#), [461](#), [489](#), [534](#), [621](#)
- Large et al. (1994), [xii](#), [16](#), [97](#), [256–258](#), [260](#), [261](#), [268](#), [269](#), [271–283](#), [289](#), [464](#), [477](#), [479](#), [480](#), [515](#), [568](#), [591](#), [621](#)
- Large et al. (1997), [256](#), [320](#), [621](#)
- Large et al. (2001), [15](#), [356](#), [357](#), [621](#)
- Large (1998), [256–258](#), [262](#), [269](#), [621](#)
- Leclair and Madec (2009), [10](#), [163](#), [621](#)
- Ledwell et al. (1993), [289](#), [479](#), [574](#), [621](#)
- Ledwell et al. (2011), [479](#), [574](#), [621](#)
- Lee et al. (2006), [xii](#), [16](#), [287](#), [288](#), [291–293](#), [512](#), [621](#)
- Lemarié et al. (2012a), [303](#), [329](#), [621](#)
- Lemarié et al. (2012b), [85](#), [90](#), [282](#), [303](#), [329](#), [622](#)
- Leonard (1979), [13](#), [622](#)
- Levitus (1982), [465](#), [622](#)
- Li et al. (2001), [257](#), [622](#)
- Lin (1997), [54](#), [57](#), [622](#)
- Lorbacher et al. (2012), [563](#), [622](#)
- Losch et al. (2004), [9](#), [77](#), [92](#), [139](#), [622](#)
- Lowe and Gregory (2006), [563](#), [622](#)
- Müller (2006), [37](#), [553](#), [624](#)
- Machenhauer et al. (2009), [44](#), [622](#)
- Madec and Imbard (1996), [123](#), [622](#)
- Maltrud and Holloway (2008), [357](#), [374](#), [375](#), [622](#)
- Manizza et al. (2005), [15](#), [249](#), [622](#)
- Marchesiello and Shchepetkin (2001), [233](#), [622](#)
- Marchuk and Kagan (1989), [109](#), [622](#)
- Marion and Thornton (1988), [26](#), [622](#)
- Marotzke (1991), [163](#), [285](#), [622](#)
- Marshall and Schott (1999), [27](#), [622](#)
- Marshall et al. (1997), [66](#), [161](#), [283](#), [622](#)
- Marshall et al. (1999), [458](#), [622](#)
- Marshall et al. (2004), [9](#), [92](#), [622](#)
- Martinsen and Engedahl (1987), [623](#)
- Maze et al. (2009), [458](#), [473](#), [623](#)
- McDougall and Church (1986), [574](#), [623](#)
- McDougall and Jackett (1988), [459](#), [460](#), [465](#), [623](#)
- McDougall and Jackett (2005), [465](#), [473](#), [537](#), [623](#)
- McDougall and McIntosh (2001), [350](#), [557](#), [558](#), [623](#)
- McDougall et al. (2002), [62](#), [156](#), [558](#), [623](#)
- McDougall et al. (2003), [97](#), [98](#), [623](#)
- McDougall et al. (2012), [99](#), [102](#), [623](#)
- McDougall (1987a), [280](#), [459](#), [574](#), [575](#), [623](#)
- McDougall (1987b), [98](#), [459](#), [464](#), [469](#), [478](#), [559](#), [574](#), [576](#), [577](#), [623](#)
- McDougall (1995), [32](#), [575](#), [623](#)
- McDougall (2003), [12](#), [40–42](#), [45](#), [102](#), [250](#), [553](#), [558](#), [569](#), [623](#)
- McWilliams and Sullivan (2001), [282](#), [623](#)
- Merryfield and Holloway (2003), [13](#), [623](#)
- Merryfield and Scott (2007), [375](#), [623](#)
- Mesinger and Arakawa (1976), [161](#), [623](#)
- Mesinger (1973), [411](#), [623](#)
- Miller and Thorpe (1981), [235](#), [623](#)
- Mitrovica et al. (2001), [103](#), [105](#), [108](#), [623](#)
- Morel and Antoine (1994), [15](#), [249](#), [251](#), [252](#), [624](#)
- Morel and Berthon (1989), [252](#), [624](#)
- Morel (1988), [251](#), [624](#)
- Morse and Feshbach (1953), [103](#), [624](#)
- Moum et al. (2002), [290](#), [624](#)
- Munk and Anderson (1948), [292](#), [624](#)
- Murray (1996), [6](#), [7](#), [121](#), [125](#), [126](#), [128](#), [624](#)
- Naveira-Garabato et al. (2004), [479](#), [574](#), [624](#)
- Nicholls and Cazenave (2010), [573](#), [624](#)
- Nikurashin and Ferrari (2010), [513](#), [624](#)
- O'Brien (1986), [179](#), [624](#)
- Ohlmann and Siegel (2000), [251](#), [624](#)
- Ohlmann (2003), [15](#), [624](#)
- Olbers et al. (1985), [574](#), [624](#)
- Orlanski (1976), [235](#), [624](#)
- Osborn (1980), [289](#), [624](#)
- Pacanowski and Gnanadesikan (1998), [6](#), [68](#), [72](#), [87–89](#), [196](#), [624](#)
- Pacanowski and Griffies (1999), [2](#), [3](#), [13](#), [21](#), [115](#), [121](#), [162](#), [216](#), [222](#), [284](#), [340](#), [411](#), [419](#), [420](#), [625](#)
- Pacanowski and Philander (1981), [16](#), [97](#), [283](#), [625](#)
- Pacanowski et al. (1991), [iii](#), [3](#), [366](#), [624](#)
- Pacanowski (1987), [162](#), [297](#), [624](#)
- Pacanowski (1995), [3](#), [624](#)
- Palma and Matano (1998), [233](#), [625](#)
- Palma and Matano (2001), [233](#), [625](#)
- Paulson and Simpson (1977), [250](#), [251](#), [625](#)
- Peixoto and Oort (1992), [595](#), [625](#)
- Phillips (1973), [103](#), [625](#)
- Pickard and Emery (1990), [433](#), [625](#)
- Polzin et al. (1997), [290](#), [479](#), [574](#), [625](#)
- Prather (1986), [13](#), [625](#)
- Press et al. (1992), [186](#), [188](#), [327](#), [625](#)
- Rahmstorf (1993), [15](#), [283–285](#), [515](#), [625](#)
- Raymond and Kuo (1984), [625](#)
- Ray (1998), [110](#), [625](#)
- Redi (1982), [83](#), [305](#), [321](#), [574–576](#), [625](#)
- Robert (1966), [162](#), [625](#)
- Robinson (2005), [486](#), [625](#)
- Roed and Cooper (1987), [233](#), [625](#)
- Rosati and Miyakoda (1988), [250](#), [625](#)
- Sadourny (1975), [178](#), [625](#)
- Schmitt (1994), [479](#), [574](#), [625](#)
- Shchepetkin and McWilliams (2002), [10](#), [11](#), [66](#), [89](#), [90](#), [626](#)
- Shchepetkin and McWilliams (2005), [159](#), [626](#)
- Shchepetkin (2005), [282](#), [626](#)
- Simmons et al. (2004), [16](#), [97](#), [287–293](#), [511](#), [626](#)

Smagorinsky (1963), [357](#), [626](#)
Smagorinsky (1993), [357](#), [626](#)
Smith and McWilliams (2003), [15](#), [357](#), [369](#), [626](#)
Smith et al. (1995), [123](#), [626](#)
Smyth et al. (2002), [257](#), [260](#), [278](#), [626](#)
Solomon (1971), [574](#), [626](#)
Sommerfeld (1949), [232](#), [626](#)
Speer and Tziperman (1992), [458](#), [626](#)
St.Laurent et al. (2001), [290](#), [626](#)
St.Laurent et al. (2002), [289](#), [626](#)
St.Laurent et al. (2003), [290](#), [626](#)
Stacey et al. (1995), [8](#), [89](#), [626](#)
Stammer (2008), [563](#), [626](#)
Starr (1945), [575](#), [626](#)
Stevens (1990), [232](#), [626](#)
Stevens (1991), [232](#), [626](#)
Stouffer et al. (2006a), [ii](#), [4](#), [21](#), [143](#), [597](#), [626](#)
Stouffer et al. (2006b), [9](#), [627](#)
Sullivan and McWilliams (2010), [282](#), [627](#)
Sve (????), [459](#), [613](#)
Sweby (1984), [13](#), [627](#)
Sweeney et al. (2005), [15](#), [569](#), [627](#)
Tang and Grimshaw (1996), [233](#), [627](#)
Tang and Roberts (2005), [383](#), [627](#)
Tennekes (1973), [269](#), [271](#), [627](#)
Terray et al. (1996), [269](#), [627](#)
Thiele and Sarmiento (1990), [52](#), [627](#)
Tomczak and Godfrey (1994), [579](#), [627](#)
Treguier et al. (1997), [79](#), [353](#), [468](#), [493](#), [516](#), [585](#),
[627](#)
Troen and Mahrt (1986), [273](#), [274](#), [277](#), [627](#)
Tziperman (1986), [458](#), [627](#)
Umlauf et al. (2005), [16](#), [256](#), [627](#)
Veronis (1973), [103](#), [104](#), [627](#)
Viúdez (2000), [458](#), [465](#), [543](#), [627](#)
Visbeck et al. (1997), [337](#), [340](#), [627](#)
Wahr (1998), [109](#), [627](#)
Walín (1982), [458](#), [459](#), [472](#), [480](#), [494](#), [503](#), [534](#),
[627](#)
Warren (2009), [37](#), [628](#)
Webb et al. (1998), [116](#), [628](#)
Winters and D'Asaro (1995), [430](#), [431](#), [433](#), [628](#)
Winters et al. (1995), [430](#), [431](#), [628](#)
Winton et al. (1998), [5](#), [380](#), [385](#), [628](#)
Wittenberg et al. (2006), [ii](#), [4](#), [21](#), [143](#), [597](#), [628](#)
Wright (1997), [99](#), [628](#)
Yin et al. (2009), [562](#), [563](#), [628](#)
Yin et al. (2010a), [562](#), [563](#), [628](#)
Yin et al. (2010b), [9](#), [628](#)
Zhao and Vallis (2008), [298](#), [628](#)
Zika et al. (2010), [459](#), [628](#)
IOC et al. (2010), [11](#), [12](#), [42](#), [97–99](#), [102](#), [263](#), [464](#),
[558](#), [559](#), [566](#), [574](#), [576](#), [620](#)