F. Parameterization of Mesoscale Eddies

In addition to the diffusion of tracers along isopycnals (Redi, 1982; Cox, 1987), the parameterization of Gent & McWilliams (1990) includes an additional tracer advection. The new equation for potential temperature, θ, is

$$\theta_t + \mathcal{L}^+(\theta) = R(A_I, \theta) + (A_{\tau V}^{si}\theta_z)_z, \tag{F1}$$

with

$$\mathcal{L}^+(G) = \left(\frac{1}{acos\phi}\right)\left[[(u+u^*)G]_\lambda + [(v+v^*)Gcos\phi]_\phi\right] + [(w+w^*)G]_z. \tag{F2}$$

Here, u, v, and w are the longitudinal (λ), latitudinal (ϕ), and vertical (z) Eulerian-mean velocity components, respectively, t is time, a is the mean radius of the earth. Also, R is the diffusion operator along isopycnals (see below; see also Redi, 1982; Cox, 1987; Gent & McWilliams, 1990), A_I is the isopycnal tracer diffusion coefficient, and u^*, v^*, and w^* are the longitudinal, latitudinal, and vertical components of the non-divergent, eddy-induced transport velocity, respectively. This transport velocity is defined by

$$\begin{aligned} u^* &= \left[\frac{A_{ITD}}{acos\phi}\frac{\varrho_\lambda}{\varrho_z}\right]_z, \qquad v^* = \left[\frac{A_{ITD}}{a}\frac{\varrho_\phi}{\varrho_z}\right]_z, \\ w^* &= -\left(\frac{1}{acos\phi}\right)\left[\left(\frac{A_{ITD}}{acos\phi}\frac{\varrho_\lambda}{\varrho_z}\right)_\lambda + \left(\frac{A_{ITD}}{a}\frac{\varrho_\phi cos\phi}{\varrho_z}\right)_\phi\right], \end{aligned} \tag{F3}$$

where ϱ is local potential density, and A_{ITD} is the isopycnal thickness diffusivity. In (F1), $A_{\tau V}$ is the diapycnal diffusion coefficient, and the superscript si indicates that the small background values are greatly enhanced in regions of static instability. Finally, G represents a generic variable in (F2).

Note that in (F2) advection is by the effective transport velocity which is the sum of the Eulerian-mean and eddy-induced transport velocities. The same conservation equation, (F1), is used also for S. The discretization of the isopycnal transport parameterization formulae is now given.

Consistent with the GFDL ocean general circulation model (Bryan, 1969; Cox, 1984), all of the continuous equations of the isopycnal transport parameterization are discretized on the Arakawa B-grid. In addition, our implementation of the parameterization is a variant of that of Cox (1987) and Pacanowski et al. (1991, 1993). In this section, we use the generic shorthand notation for spatial differencing, i.e.,

$$\delta_\lambda = \frac{1}{\Delta\lambda_i}[G_{i+1/2} - G_{i-1/2}]. \tag{F4}$$

Here, the subscript i denotes the discrete spatial location at which the gradient is evaluated and 1/2 refers to one-half grid point offset with respect to the point i. $\Delta\lambda$, $\Delta\phi$, and Δz

are the grid spacings in the longitudinal, latitudinal, and vertical directions, respectively. Similarly, we apply the following generic shorthand notation for spatial averaging,

$$\overline{G}^{\lambda} = \frac{1}{2}[G_{i+1/2} + G_{i-1/2}]. \tag{F5}$$

Because the leap-frog scheme used in the model is unconditionally unstable if the diffusion terms are evaluated at the central time level, the isopycnal diffusion and eddy-induced transport terms are evaluated at the previous time level. In the following discrete relations, we omit the temporal index for clarity.

A. Isopycnal Diffusion

The diffusion operator R, given in (F1), is defined by

$$R(A_I, G) \equiv \nabla_{3D} \cdot [A_I \mathcal{K} \cdot \nabla_{3D} G], \tag{F6}$$

where ∇_{3D} is the three-dimensional gradient operator in spherical coordinates and $\mathcal{K}$ is a three-dimensional second-rank tensor,

$$\mathcal{K} \equiv \begin{pmatrix} \mathbf{I} & \mathbf{S}^T \\ \mathbf{S} & \mathbf{S}\cdot\mathbf{S} \end{pmatrix}. \tag{F7}$$

Here, $\mathbf{I}$ is the 2×2 identity matrix, T denotes transpose, and

$$\mathbf{S} \equiv -\nabla\varrho/\varrho_z. \tag{F8}$$

In the derivation of (F7), the small slope approximation is used, i. e., $|\mathbf{S}| \ll 1$.

Local potential densities, ϱ, are computed relative to some selected reference depths, differing from Cox (1987). This approach expedites the parameterization implementation in a general coordinate model. In the current 2×2 and 3×3 models, we use 6 and 8 reference depth (pressure) levels. For example, 150, 650, 1500, 2500, 3500, and $4500m$ are used as reference levels in the 3×3 model. Thus, ϱ is referenced to a depth at most $500m$ away. The test cases with more reference depths showed no significant changes in the solutions.

Whenever a new set of local potential density gradients is computed, the slopes of the isopycnal surfaces are checked to prevent numerical instability (Cox, 1987). For this purpose, we consider two approaches. The first method, already implemented in the GFDL model, is based on changing the value of ϱ_z if the slope is larger than a permissible maximum slope, S_{max}, e. g.,

$$check = -\frac{\sqrt{\varrho_\lambda^2 + \varrho_\phi^2}}{S_{max}}, \tag{F9}$$

$$if \quad \varrho_z > check \quad \Longrightarrow \quad \varrho_z = check. \tag{F10}$$

Because the slopes of the isopycnals are limited by S_{max}, slopes larger than this produce spurious diapycnal diffusion. This can be avoided to some extent by increasing S_{max} at the expense of smaller time steps. In our present solutions, we use $S_{max} = 0.01$, allowing the inclusion of high slopes encountered in frontal regions. This first approach is the default when the ifdef option *isopycmix* is defined.

As stated above, a major disadvantage of the maximum slope approach is that it will produce diapycnal diffusion for slopes larger than S_{max}. To remedy this problem, the second method, while still allowing mixing along the computed slopes, actually tapers A_I and A_{ITD} to zero. To activate this method, the ifdef option *isopycmixspatialvar* must be defined in addition to the *isopycmix* option. The tapering is accomplished by the application of two functions,

$$A_I \quad \Longrightarrow f_1\ f_2\ A_I, \quad A_{ITD} \quad \Longrightarrow f_1\ f_2\ A_{ITD}. \tag{F11}$$

Here, f_1 is a hyperbolic tangent function,

$$f_1 = 0.5\left(1 + tanh\left[\frac{-|\mathbf{S}| + S_c}{S_d}\right]\right), \tag{F12}$$

which diminishes the diffusion coefficients as $|\mathbf{S}|$ gets large. In (F12), S_c is the slope at which $f_1 = 0.5$, and S_d is the half length of the interval in which f_1 changes with a steep slope. (F12) is applied in the interval $0 \geq -|\mathbf{S}| \geq -S_{max}$. Outside of this interval, f_1 is set to zero. The second function, f_2, is designed to turn off the isopycnal mixing scheme near the ocean surface, thus allowing the specified vertical mixing processes to function fully. It is based on the depth of the water parcel, the local Rossby deformation radius, and the local potential density slope. The details of this function can be found in the *isopyc.F* subroutine. We strongly recommend the use of the second approach especially for models with high vertical resolution near the ocean surface. Note that in both (F9) and (F12), the negative sign is used to denote that the isopycnal transport parameterization is applicable only in stably stratified regions.

The (1-3) component of the tensor, $\mathcal{K}_{13}$, is obtained at the center of the eastern face of the tracer grid box. Consequently, the associated local potential density gradients are obtained there using

$$\varrho_\lambda = \frac{m^T}{a}\delta_\lambda[\varrho], \tag{F13}$$

$$\varrho_\phi = \frac{1}{a}\delta_\phi[\overline{\varrho}^{\lambda,\phi}], \tag{F14}$$

$$\varrho_z = \delta_z[\overline{\varrho}^{\lambda,z}]. \tag{F15}$$

In (F13), m^T is $sec\phi$ evaluated at the tracer grid points.

After checking the slopes of the isopycnal surfaces, $\mathcal{K}_{13}$ is computed from

$$\mathcal{K}_{13} = -\frac{\varrho_\lambda}{\varrho_z}. \tag{F16}$$

Note that (F14) is exclusively used during the slope check procedure, and it is set to zero if any one of the neighboring tracer grid points is land.

The (2-3) component of the tensor, $\mathcal{K}_{23}$, is obtained at the center of the northern face of the tracer grid box. Hence, the required potential density gradients are computed at these locations. The discrete forms are

$$\varrho_\lambda = \frac{m^u}{a}\delta_\lambda[\overline{\varrho}^{\lambda,\phi}], \tag{F17}$$

$$\varrho_\phi = \frac{1}{a}\delta_\phi[\varrho], \tag{F18}$$

$$\varrho_z = \delta_z[\overline{\varrho}^{\phi,z}], \tag{F19}$$

where m^u is $sec\phi$ computed at the velocity grid points. After slope check, $\mathcal{K}_{23}$ is computed from

$$\mathcal{K}_{23} = -\frac{\varrho_\phi}{\varrho_z}. \tag{F20}$$

(F17) is used only for slope check, and $\varrho_\lambda = 0$ if any one of the neighboring tracer grid points is land.

$\mathcal{K}_{13}$ and $\mathcal{K}_{23}$ are subject to the zero flux condition at horizontal boundaries. In order to compute the vertical gradients of ϱ at the ocean surface and bottom, an extrapolative estimator is used to obtain ϱ.

$\mathcal{K}_{31}$, $\mathcal{K}_{32}$, and $\mathcal{K}_{33}$ are computed at the center of the vertical face of the tracer grid box. The discrete forms for the local potential density gradients are

$$\varrho_\lambda = \frac{m^T}{a}\frac{\overline{\Delta\lambda^u\delta_\lambda[\varrho]}^{\lambda,z}}{\Delta\lambda^T}, \tag{F21}$$

$$\varrho_\phi = \frac{1}{a}\frac{\overline{\Delta\phi^u\delta_\phi[\varrho]}^{\phi,z}}{\Delta\phi^T}, \tag{F22}$$

$$\varrho_z = \delta_z[\varrho]. \tag{F23}$$

Here, the superscripts u and T refer to the velocity and tracer grid points, respectively. Note that in case of equal mesh spacing in the horizontal directions, the use of $\Delta\lambda^u\delta_\lambda$ or $\Delta\phi^u\delta_\phi$ is equivalent to computing averages of the gradients. After slope check, we use

$$\mathcal{K}_{31} = -\frac{\varrho_\lambda}{\varrho_z}, \tag{F24}$$

$$\mathcal{K}_{32} = -\frac{\varrho_\phi}{\varrho_z}, \tag{F25}$$

$$\mathcal{K}_{33} = \frac{\varrho_\lambda^2 + \varrho_\phi^2}{\varrho_z^2}. \tag{F26}$$

For $\mathcal{K}_{31}$, $\mathcal{K}_{32}$, and $\mathcal{K}_{33}$ components of the tensor, in addition to the horizontal boundaries, we impose the zero flux condition at the top and bottom boundaries.

Finally, (F6) is discretized as follows,

$$
\begin{aligned}
R(A_I, G) = {} & \frac{m^T}{a}\delta_\lambda\left(A_I\frac{m^T}{a}\delta_\lambda[G] + A_I\mathcal{K}_{13}\delta_z[\overline{G}^{\lambda,z}]\right) \\
& + \frac{m^T}{a}\delta_\phi\left(A_I\frac{1}{m^u a}\delta_\phi[G] + \frac{1}{m^u}A_I\mathcal{K}_{23}\delta_z[\overline{G}^{\phi,z}]\right) \\
& + \delta_z\left(\frac{m^T}{a}A_I\mathcal{K}_{31}\frac{\overline{\Delta\lambda^u\delta_\lambda[G]}^{\lambda,z}}{\Delta\lambda^T} + \frac{1}{a}A_I\mathcal{K}_{32}\frac{\overline{\Delta\phi^u\delta_\phi[G]}^{\phi,z}}{\Delta\phi^T} + A_I\mathcal{K}_{33}\delta_z[G]\right).
\end{aligned}
\tag{F27}
$$

Equation (F27) is subject to the no tracer flux condition at all boundaries. The necessary vertical gradients at the first level are computed assuming that the surface tracer values are the same as those at the first level. For the bottom level, an extrapolative estimator is used to compute the tracer values at the ocean floor.

B. Eddy-Induced Transport Velocity and Advection

The eddy-induced transport velocity components, u^*, v^*, and w^* are computed at the centers of the eastern, northern, and vertical faces of the tracer grid box, respectively. The discrete forms for u^* and v^* are

$$
u^* = -\delta_z[\overline{A_{ITD}\mathcal{K}_{13}}^z], \tag{F28}
$$

$$
v^* = -\delta_z[\overline{A_{ITD}\mathcal{K}_{23}}^z]. \tag{F29}
$$

The vertical component of the eddy-induced transport velocity is directly obtained by integration of the continuity equation for this velocity. Note that the application of (F28) and (F29) at the top and bottom levels requires the values of the tensor elements at the ocean top and bottom. The requirement that the vertical eddy-induced velocity is zero at the ocean top and bottom shows that $A_{ITD}\mathcal{K}_{13}$ and $A_{ITD}\mathcal{K}_{23}$ must be zero at these surfaces.

Finally, the eddy-induced advection terms in (F2) are discretized as

$$
\mathcal{L}(G) = \frac{m^T}{a}\left[\delta_\lambda(u^*\overline{G}^\lambda) + \delta_\phi\left(\frac{v^*\overline{G}^\phi}{m^u}\right)\right] + \delta_z(w^*\overline{G}^z). \tag{$F30$}
$$

REMARKS:

1. In order to eliminate any horizontal diffusion of tracers, the horizontal diffusivity coefficient, ah, must be set to zero.
2. A_I and A_{ITD} have the same spatial dependency.
3. Because the current implementation uses an extrapolative formula near the ocean surface and bottom, the values in the bottom topography array, kmt, must be ≥ 2 in the active ocean.

4. If the ifdef option *isopycmix* is specified, the ifdef option *skipland* must not be used.
5. Although it has the same option name as the Redi (1982) and Cox (1987) isopycnal mixing parameterization in MOM 1.0 and MOM 1.1, it is NOT the Redi-Cox version.
6. A_I and A_{ITD} may have vertical variation (*fzisop* in the model). The default is no depth variation. To utilize this capability, the user must insert a function into the 2586 do-loop in *ocean.F*. The vertical variation is ineffective when the *isopycmixspatialvar* option is defined.

Bryan, K., 1969: A numerical method for the study of the circulation of the world ocean. *J. Computational Phys.*, **4**, 347-376.

Cox, M. D., 1984: A primitive equation, 3-dimensional model of the ocean. GFDL Ocean Group Tech. Rep. No. 1, GFDL/Princeton University, Princeton.

Cox, M. D., 1987: Isopycnal diffusion in a z-coordinate ocean model. *Ocean Modelling*, **74**, 1-5.

Gent, P. R. & J. C. McWilliams, 1990: Isopycnal mixing in ocean circulation models. *J. Phys. Oceanogr.*, **20**, 150-155.

Pacanowski, R. C., K. Dixon, & A. Rosati, 1991, 1993: The GFDL modular ocean model users guide. GFDL Ocean Group Tech. Rep. No. 2, GFDL, Princeton.

Redi, M. H., 1982: Oceanic isopycnal mixing by coordinate rotation. *J. Phys. Oceanogr.*, **12**, 1154-1158.