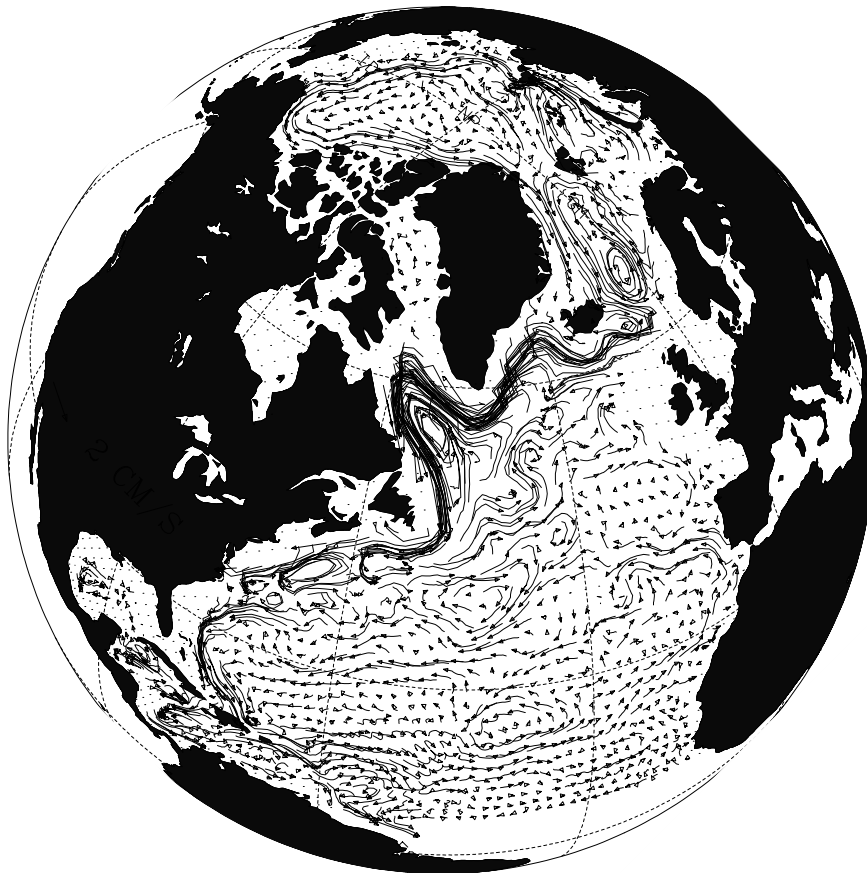


USERS GUIDE for
***A THREE-DIMENSIONAL, PRIMITIVE EQUATION,
NUMERICAL OCEAN MODEL***

George L. Mellor
Program in Atmospheric and Oceanic Sciences
Princeton University, Princeton, NJ 08544-0710



This revision: July 1998

About This Revision This version of the users guide recognizes changes that have occurred since 1991. The code itself incorporates some recent changes. The Fortran names, TMEAN, SMEAN have been changed (globally) to TCLIM, SCLIM in order to distinguish the function and treatment of these variables from that of RMEAN. The names, TRNU, TRNV, have been changed to DRX2D, DRY2D and the names, ADVUU, ADVVV, to ADX2D, ADY2D to more clearly indicate their functions. Instead of a wind driven closed basin, pom97.f now solves the problem of the flow through a channel which includes an island or a seamount at the center of the domain. Thus, subroutine BCOND contains active open boundary conditions. These illustrative boundary conditions, however, are one set of many possibilities and, consequently, open boundary conditions for regional models pose difficult choices for users of the model.

This June 1998 revision contains a fuller discussion of open boundary conditions in section 16.

As of this revision date, there are over 340 POM users of record.

This revision: JUNE 1998

Sponsor Acknowledgment: The development and application of the program has had many sponsors since 1977. They include the Geophysical Fluid Dynamics Laboratory/NOAA, Princeton University, Sea Grant/NOAA through the New Jersey Marine Sciences Consortium, the Department of Energy, Minerals Management Services/DOI, the National Ocean Services/NOAA, the Institute of Naval Oceanography and the Office of Naval Research/DOD.

Web site: <http://www.aos.princeton.edu/WWWPUBLIC/htdocs.pom/>

Title Page Illustration: North Atlantic velocity field on the 32.45 potential density surface. Courtesy Dr. Sirpa Häkkinen.

CONTENTS

	Page
1. INTRODUCTION	4
2. THE BASIC EQUATIONS	6
3. FORTRAN SYMBOLS	13
4. THE NUMERICAL SCHEME	16
5. comblk.h	22
6. PROGRAM MAIN and the external mode	22
7. SUBROUTINE ADVAVE	23
8. SUBROUTINE ADVT	23
9. SUBROUTINE PROFT	23
10. SUBROUTINE BAROPG	26
11. SUBROUTINES ADVCT, ADVU AND ADVV	26
12. SUBROUTINES PROFU AND PROFV	27
13. SUBROUTINE ADVQ	27
14. SUBROUTINE PROFQ	27
15. SUBROUTINE VERTVL	28
16. SUBROUTINE BCOND	28
17. SUBROUTINE DENS	33
18 SUBROUTINE SLPMIN	33
19. Utility Subroutines	33
20. PROGRAM CURVIGRID	32
APPENDIX A	35
REFERENCES	39

1. INTRODUCTION

This report is documentation for a numerical ocean model created by Alan Blumberg and me around 1977. Subsequent contributions were made by Leo Oey, Jim Herring, Lakshmi Kantha and Boris Galperin and others. In recent years Tal Ezer has been an important force in research using the model and in helping others use it. Institutionally, the model was developed and applied to oceanographic problems in the Atmospheric and Oceanic Sciences Program of Princeton University, the Geophysical Fluid Dynamics Laboratory of NOAA and Dynalysis of Princeton. Many sponsors, as acknowledged above, have supported the effort. Papers that either describe the numerical model (Blumberg and Mellor, 1987) or made use of the model are contained in the Reference Section and a more complete list is available on the POM home page at <http://www.aos.princeton.edu/WWWPUBLIC/htdocs.pom>.

The model is oftentimes referenced as the Princeton Ocean Model (POM). The principal attributes of the model are as follows:

- o It contains an imbedded second moment turbulence closure sub-model to provide vertical mixing coefficients.
- o It is a sigma coordinate model in that the vertical coordinate is scaled on the water column depth.
- o The horizontal grid uses curvilinear orthogonal coordinates and an "Arakawa C" differencing scheme.
- o The horizontal time differencing is explicit whereas the vertical differencing is implicit. The latter eliminates time constraints for the vertical coordinate and permits the use of fine vertical resolution in the surface and bottom boundary layers.
- o The model has a free surface and a split time step. The external mode portion of the model is two-dimensional and uses a short time step based on the CFL condition and the external wave speed. The internal mode is three-dimensional and uses a long time step based on the CFL condition and the internal wave speed.
- o Complete thermodynamics have been implemented.

The turbulence closure sub-model is one that I introduced (Mellor, 1973) and then was significantly advanced in collaboration with Tetsuji Yamada (Mellor and Yamada, 1974; Mellor and Yamada, 1982). It is often cited in the literature as the Mellor-Yamada turbulence closure model (but, it should be noted that the model is based on

turbulence hypotheses by Rotta and Kolmogorov which we extended to stratified flow cases). Here, the Level 2.5 model is used together with a prognostic equation for the turbulence macroscale. The closure model is contained in subroutines PROFQ and ADVQ. A list of papers pertaining to the closure model is also included in the Reference section. A much more extensive list of references by user of POM is on the web site.

By and large, the turbulence model seems to do a fair job simulating mixed layer dynamics although there have been indications that calculated mixed layer depths are a bit too shallow (Martin, 1985). Also, wind forcing may be spatially smoothed and temporally smoothed. It is known that the latter process will reduce mixed layer thicknesses (Klein, 1980). Further study is required to quantify these effects.

The sigma coordinate system is probably a necessary attribute in dealing with significant topographical variability such as that encountered in estuaries or over continental shelf breaks and slopes. Together with the turbulence sub-model, the model produces realistic bottom boundary layers which are important in coastal waters (Mellor, 1985) and in tidally driven estuaries (Oey et al., 1985a, b) which the model can simulate since it does have a free surface. More recently, we find that bottom boundary layers are important for deep water formation processes (Zavatarelli and Mellor, 1995; Jungclaus and Mellor, 1996; Baringer and Price, 1996) and, possibly, for the maintenance of the baroclinicity of oceans basins (Mellor and Wang, 1996).

The horizontal finite difference scheme is staggered and, in the literature, has been called an Arakawa C-grid. The horizontal grid is a curvilinear coordinate system, or as a special case, a rectilinear coordinate system may be easily implemented. The advection, horizontal diffusion and, in the case of velocity, the pressure gradient and Coriolis terms are contained in subroutines ADVT, ADVQ, ADVCT, ADVU, ADVV and ADVAVE. The horizontal differencing could be changed without affecting the overall logic of the program or the remaining subroutines. The vertical diffusion is handled in subroutines, PROFT, PROFQ, PROFU and PROFV.

The specific program that is now supplied to outside users (as of June 1996) simulates the flow, east to west across a seamount with a prescribed vertical temperature stratification, constant salinity, zero surface heat and salinity flux and a zero wind stress distribution although wind stress may be easily applied. The program should run with no additional data requirements. The open boundary conditions specified in SUBROUTINE BCOND for this problem are a sampling of many possible open boundary conditions. I leave it to users to invent their own problems, defined by topography, horizontal grid (rectilinear, where $DX(I, J)$ is specified as a function of I and $DY(I, J)$ as a function of J , or a more general orthogonal curvilinear grid), vertical sigma grid and boundary

conditions. Users may need to alter PROGRAM MAIN and SUBROUTINE BCOND; in principal, there should be no need to alter any of the other subroutines.

The present program code is written in standard FORTRAN 77. There are other versions in existence, but we only support and maintain the single version.

Provision has been made so that the 2-D (external mode) portion of the model can be run *cum sole*. In this case, the bottom shear stress, normally a consequence of the 3-D calculation and the turbulence mixing coefficient, is replaced by a quadratic drag relation. The code may also be run in a diagnostic mode where the thermodynamic properties are invariant in time.

Users will need to write their own code to set up their own problem dependent, initial conditions and lateral and surface boundary conditions. We can, however, supply simple subroutines that convert data for a constant z-level coordinate system to a sigma coordinate system and vice versa.

To access pom.f and other files through Internet, type **ftp ftp.gfdl.gov**; when prompted for your name, type **anonymous**; when prompted for a password, type your internet address; after receiving a guest login ok, type **cd pub/glm**. You may list filenames with the **ls** command. You may download with the command **get filename**. Type **quit** to terminate.

The current code as of February 1998 is called pom97.f. To run the code, transfer pom97.f and comblk97.h to a directory, compile and run.

2. THE BASIC EQUATIONS

The basic equations have been cast in a bottom following, sigma coordinate system which is illustrated in Figure 1. The reader is referred to Phillips (1957) or Blumberg and Mellor (1980,1987) for a derivation of the sigma coordinate equations which are based on the transformation,

$$x^* = x, y^* = y, \sigma = \frac{z - \eta}{H + \eta}, t^* = t \quad (1a, b, c, d)$$

where x, y, z are the conventional cartesian coordinates; $D \equiv H + \eta$ where $H(x, y)$ is the bottom topography and $\eta(x, y, t)$ is the surface elevation. Thus, σ ranges from $\sigma = 0$ at $z = \eta$ to $\sigma = -1$ at $z = -H$. After conversion to sigma coordinates and deletion of the asterisks, the basic equations may be written (in horizontal cartesian coordinates),

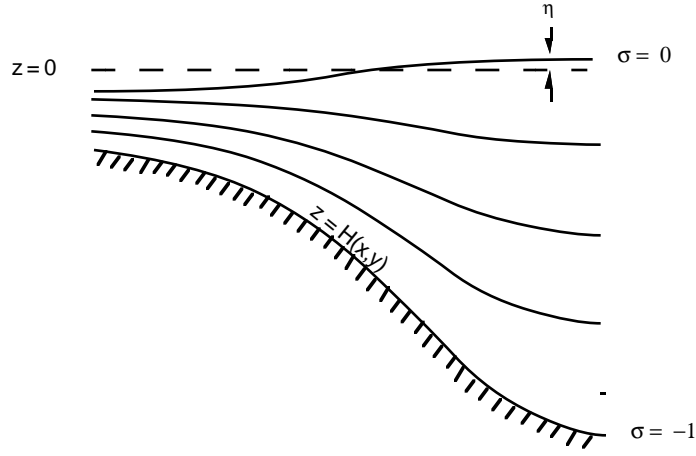


Figure 1. The sigma coordinate system.

$$\frac{\partial DU}{\partial x} + \frac{\partial DV}{\partial y} + \frac{\partial \omega}{\partial \sigma} + \frac{\partial \eta}{\partial t} = 0 \quad (2)$$

$$\begin{aligned} \frac{\partial UD}{\partial t} + \frac{\partial U^2 D}{\partial x} + \frac{\partial UV D}{\partial y} + \frac{\partial U \omega}{\partial \sigma} - fVD + gD \frac{\partial \eta}{\partial x} \\ + \frac{gD^2}{\rho_o} \int_{\sigma}^{\sigma'} \left[\frac{\partial \rho'}{\partial x} - \frac{\sigma'}{D} \frac{\partial D}{\partial x} \frac{\partial \rho'}{\partial \sigma'} \right] d\sigma' = \frac{\partial}{\partial \sigma} \left[\frac{K_M}{D} \frac{\partial U}{\partial \sigma} \right] + F_x \end{aligned} \quad (3)$$

$$\begin{aligned} \frac{\partial VD}{\partial t} + \frac{\partial UV D}{\partial x} + \frac{\partial V^2 D}{\partial y} + \frac{\partial V \omega}{\partial \sigma} + fUD + gD \frac{\partial \eta}{\partial y} \\ + \frac{gD^2}{\rho_o} \int_{\sigma}^{\sigma'} \left[\frac{\partial \rho'}{\partial y} - \frac{\sigma'}{D} \frac{\partial D}{\partial y} \frac{\partial \rho'}{\partial \sigma'} \right] d\sigma' = \frac{\partial}{\partial \sigma} \left[\frac{K_M}{D} \frac{\partial V}{\partial \sigma} \right] + F_y \end{aligned} \quad (4)$$

$$\frac{\partial TD}{\partial t} + \frac{\partial TUD}{\partial x} + \frac{\partial TVD}{\partial y} + \frac{\partial T \omega}{\partial \sigma} = \frac{\partial}{\partial \sigma} \left[\frac{K_H}{D} \frac{\partial T}{\partial \sigma} \right] + F_T - \frac{\partial R}{\partial z} \quad (5)$$

$$\frac{\partial SD}{\partial t} + \frac{\partial SUD}{\partial x} + \frac{\partial SVD}{\partial y} + \frac{\partial S \omega}{\partial \sigma} = \frac{\partial}{\partial \sigma} \left[\frac{K_H}{D} \frac{\partial S}{\partial \sigma} \right] + F_S \quad (6)$$

$$\frac{\partial q^2 D}{\partial t} + \frac{\partial Uq^2 D}{\partial x} + \frac{\partial Vq^2 D}{\partial y} + \frac{\partial \omega q^2}{\partial \sigma} = \frac{\partial}{\partial \sigma} \left[\frac{K_q}{D} \frac{\partial q^2}{\partial \sigma} \right]$$

$$+ \frac{2K_M}{D} \left[\left(\frac{\partial U}{\partial \sigma} \right)^2 + \left(\frac{\partial V}{\partial \sigma} \right)^2 \right] + \frac{2g}{\rho_o} K_H \frac{\partial \tilde{\rho}}{\partial \sigma} - \frac{2Dq^3}{B_1 \ell} + F_q \quad (7)$$

$$\begin{aligned} \frac{\partial q^2 \ell D}{\partial t} + \frac{\partial U q^2 \ell D}{\partial x} + \frac{\partial V q^2 \ell D}{\partial y} + \frac{\partial \omega q^2 \ell}{\partial \sigma} &= \frac{\partial}{\partial s} \left[\frac{K_q}{D} \frac{\partial q^2 \ell}{\partial \sigma} \right] \\ + E_1 \ell \left(\frac{K_M}{D} \left[\left(\frac{\partial U}{\partial \sigma} \right)^2 + \left(\frac{\partial V}{\partial \sigma} \right)^2 \right] + E_3 \frac{g}{\rho_o} K_H \frac{\partial \tilde{\rho}}{\partial \sigma} \right) &\tilde{W} - \frac{Dq^3}{B_1} + F_\ell \end{aligned} \quad (8)$$

Definitions of the variables are contained in section 3. Note that ω is the transformed vertical velocity; physically, ω is the velocity component normal to sigma surfaces. The transformation to the Cartesian vertical velocity is

$$W = \omega + U \left(\sigma \frac{\partial D}{\partial x} + \frac{\partial \eta}{\partial x} \right) + V \left(\sigma \frac{\partial D}{\partial y} + \frac{\partial \eta}{\partial y} \right) + \sigma \frac{\partial D}{\partial t} + \frac{\partial \eta}{\partial t}$$

The so-called wall proximity function is prescribed according to $\tilde{W} = 1 + E_2(\ell / kL)$ where $L^{-1} = (\eta - z)^{-1} + (H - z)^{-1}$. Also, $\partial \tilde{\rho} / \partial \sigma \equiv \partial \rho / \partial \sigma - c_s^{-2} \partial p / \partial \sigma$ (see discussion of static stability in Appendix A) where c_s is the speed of sound. Note that T is potential temperature (see Appendix A).

In equations (3) and (4), RMEAN should be subtracted from ρ to form ρ' before the integration is carried out in subroutine BAROPG. RMEAN is generally the initial density field which is area averaged on z-levels and then transferred to sigma coordinates in the exact same way as the initial density field. This procedure should reduce the truncation errors associated with the calculation of the pressure gradient term in sigma coordinate over steep topography (see Mellor et al., 1994, for evaluation of this error in POM).

The horizontal viscosity and diffusion terms are defined according to:

$$F_x \equiv \frac{\partial}{\partial x} (H \tau_{xx}) + \frac{\partial}{\partial y} (H \tau_{xy}) \quad (9a)$$

$$F_y \equiv \frac{\partial}{\partial x} (H \tau_{xy}) + \frac{\partial}{\partial y} (H \tau_{yy}) \quad (9b)$$

where

$$\tau_{xx} = 2A_M \frac{\partial U}{\partial x}, \quad \tau_{xy} = \tau_{yx} = A_M \left(\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x} \right), \quad \tau_{yy} = 2A_M \frac{\partial V}{\partial y} \quad (10a,b,c)$$

Also,

$$F_\phi \equiv \frac{\partial}{\partial x}(Hq_x) + \frac{\partial}{\partial y}(Hq_y) \quad (11)$$

where

$$q_x \equiv A_H \frac{\partial \phi}{\partial x}, \quad q_y \equiv A_H \frac{\partial \phi}{\partial y} \quad (12a,b)$$

and where ϕ represents T , S , q^2 or $q^2 \ell$. It should be noted that these horizontal diffusion terms are not what one would obtain by transforming the conventional forms to the sigma coordinate system. Justification for the present forms will be found in Mellor and Blumberg (1985) and relate to the fact that we wish to maintain a valid bottom boundary layer simulation in the face of horizontal diffusion which may be large. The penalty for this is that (12a,b) in sigma coordinates can introduce vertical fluxes even when isotherms and isohalines are flat in cartesian coordinates. The remedy for this is, first, the use of a Smagorinsky diffusivity (see below) so that, at least when velocities are small or nil, so are the values of q_x and q_y . The second remedy is that, before executing (12a, b) for temperature or salinity, we first subtract T_{CLIM} and S_{CLIM} which are "climatologies" of T and S . The latter may be true climatologies (e.g.; Levitus) or approximations such as temperature and salinities which are area averaged prior to transfer to sigma coordinates (in which case, they are treated the same as ρ_{MEAN}). If something like a Levitus climatology is used, then most of the vertical component of the diffusion is removed; furthermore, the diffusion terms tend to drive the scalars back to climatology rather than to a horizontally homogeneous state as in the case of z - level models. As resolution improves, the whole issue disappears. In the meantime, one has a three-dimensional model that can model bottom boundary layers. The bottom boundary layer is important in tidally driven regions, in wind driven coastal regions and according to Mellor and Wang (1996), in deep ocean basins.

In (9a, b) and (11), H is used in place of D for the small algorithmic simplification it offers for terms whose physical significance is questionable.

The Smagorinsky Diffusivity

We generally use the Smagorinsky diffusivity for horizontal diffusion although a constant or biharmonic diffusion can and has been used instead. The Smagorinsky formula is,

$$A_M = C\Delta x\Delta y \frac{1}{2} |\nabla\mathbf{V} + (\nabla\mathbf{V})^T|$$

where $|\nabla\mathbf{V} + (\nabla\mathbf{V})^T|/2 = [(\partial u / \partial x)^2 + (\partial v / \partial x + \partial u / \partial y)^2 / 2 + (\partial v / \partial y)^2]^{1/2}$. Values of C (the HORCON parameter) in the range, 0.10 to 0.20 seem to work well, but, if the grid spacing is small enough (Oey *et al*, 1985a,b), C can be nil. An advantage of the Smagorinsky is that C is non-dimensional; related advantages are that A_M decreases as resolution improves and that A_M is small if velocity gradients are small.

Vertical Boundary Conditions.

The vertical boundary conditions for (2) are

$$\omega(0) = \omega(-1) = 0 \quad (13a,b)$$

The boundary conditions for (3) and (4) are

$$\frac{K_M}{D} \left(\frac{\partial U}{\partial \sigma}, \frac{\partial V}{\partial \sigma} \right) = - \langle wu(0) \rangle, \langle wv(0) \rangle, \sigma \rightarrow 0 \quad (14a,b)$$

where the right hand side of (14a,b) is the input values of the surface turbulence momentum flux (the stress components are opposite in sign), and

$$\frac{K_M}{D} \left(\frac{\partial U}{\partial \sigma}, \frac{\partial V}{\partial \sigma} \right) = C_z [U^2 + V^2]^{1/2} (U, V), \sigma \rightarrow -1 \quad (14c,d)$$

where

$$C_z = \text{MAX} \left[\frac{k^2}{[\ln\{(1 + \sigma_{kb-1})H / z_o\}]^2}, 0.0025 \right] \quad (14e)$$

$k = 0.4$ is the von Karman constant and z_o is the roughness parameter. Equations (14c,d,e) can be derived by matching the numerical solution to the "law of the wall". Numerically, they are applied to the first grid points nearest the bottom. Where the bottom

is not well resolved, $(1+\sigma_{\text{kb}-1})H/z_o$ is large and (14e) reverts to an ordinary drag coefficient formulation. The boundary conditions on (5) and (6) are

$$\frac{K_H}{D} \left(\frac{\partial T}{\partial \sigma}, \frac{\partial S}{\partial \sigma} \right) = - \langle w\theta(0) \rangle, \quad \sigma \rightarrow 0 \quad (15a,b)$$

$$\frac{K_H}{D} \left(\frac{\partial T}{\partial \sigma}, \frac{\partial S}{\partial \sigma} \right) = 0, \quad \sigma \rightarrow -1 \quad (15c,d)$$

The boundary conditions for (7) and (8) are

$$(q^2(0), q^2\ell(0)) = (B_1^{2/3} u_\tau^2(0), 0) \quad (16a,b)$$

$$(q^2(-1), q^2\ell(-1)) = (B_1^{2/3} u_\tau^2(-1), 0) \quad (16c,d)$$

where B_1 is one of the turbulence closure constants and u_τ is the friction velocity at the top or bottom as denoted.

The Vertically Integrated Equations

The equations, governing the dynamics of coastal circulation, contain fast moving external gravity waves and slow moving internal gravity waves. It is desirable in terms of computer economy to separate the vertically integrated equations (external mode) from the vertical structure equations (internal mode). This technique, known as mode splitting (Simons, 1974; Madala and Piacsek, 1977) permits the calculation of the free surface elevation with little sacrifice in computational time by solving the velocity transport separately from the three-dimensional calculation of the velocity and the thermodynamic properties.

The velocity transport, external mode equations are obtained by integrating the internal mode equations over the depth, thereby eliminating all vertical structure. Thus, by integrating Equation (2) from $\sigma = -1$ to $\sigma = 0$ and using the boundary conditions (13a,b), an equation for the surface elevation can be written as

$$\frac{\partial \eta}{\partial t} + \frac{\partial \bar{U}D}{\partial x} + \frac{\partial \bar{V}D}{\partial y} = 0 \quad (17)$$

After integration, the momentum equations, (3) and (4), become

$$\begin{aligned} \frac{\partial \bar{U}D}{\partial t} + \frac{\partial \bar{U}^2 D}{\partial x} + \frac{\partial \bar{U}\bar{V}D}{\partial y} - \tilde{F}_x - f\bar{V}D + gD \frac{\partial \eta}{\partial x} = -\langle wu(0) \rangle + \langle wu(-1) \rangle \\ + G_x - \frac{gD}{\rho_o} \int_{-1}^o \int_{\sigma}^o \left[D \frac{\partial \rho'}{\partial x} - \frac{\partial D}{\partial x} \sigma' \frac{\partial \rho'}{\partial \sigma} \right] d\sigma' d\sigma \end{aligned} \quad (18)$$

$$\begin{aligned} \frac{\partial \bar{V}D}{\partial t} + \frac{\partial \bar{U}\bar{V}D}{\partial x} + \frac{\partial \bar{V}^2 D}{\partial y} - \tilde{F}_y + f\bar{U}D + gD \frac{\partial \eta}{\partial y} = -\langle wv(0) \rangle + \langle wv(-1) \rangle \\ + G_y - \frac{gD}{\rho_o} \int_{-1}^o \int_{\sigma}^o \left[D \frac{\partial \rho'}{\partial y} - \frac{\partial D}{\partial y} \sigma' \frac{\partial \rho'}{\partial \sigma} \right] d\sigma' d\sigma \end{aligned} \quad (19)$$

The overbars denote vertically integrated velocities such as

$$\bar{U} \equiv \int_{-1}^o U d\sigma. \quad (20)$$

The wind stress components are $-\langle \omega u(0) \rangle$ and $-\langle \omega u(-1) \rangle$, and the bottom stress components are $-\langle \omega u(0) \rangle$ and $-\langle \omega u(-1) \rangle$. The quantities \tilde{F}_x and \tilde{F}_y are defined according to

$$\tilde{F}_x = \frac{\partial}{\partial x} \left[H2\bar{A}_M \frac{\partial \bar{U}}{\partial x} \right] + \frac{\partial}{\partial y} \left[H\bar{A}_M \left(\frac{\partial \bar{U}}{\partial y} + \frac{\partial \bar{V}}{\partial x} \right) \right] \quad (21a)$$

and

$$\tilde{F}_y = \frac{\partial}{\partial y} \left[H2\bar{A}_M \frac{\partial \bar{V}}{\partial y} \right] + \frac{\partial}{\partial x} \left[H\bar{A}_M \left(\frac{\partial \bar{U}}{\partial y} + \frac{\partial \bar{V}}{\partial x} \right) \right] \quad (21b)$$

The so-called dispersion terms are defined according to

$$G_x = \frac{\partial \bar{U}^2 D}{\partial x} + \frac{\partial \bar{U}\bar{V}D}{\partial y} - \tilde{F}_x - \frac{\partial \bar{U}^2 D}{\partial x} - \frac{\partial \bar{U}\bar{V}D}{\partial y} + \bar{F}_x \quad (22a)$$

$$G_y = \frac{\partial \bar{U}\bar{V}D}{\partial x} + \frac{\partial \bar{V}^2 D}{\partial y} - \tilde{F}_y - \frac{\partial \bar{U}\bar{V}D}{\partial x} - \frac{\partial \bar{V}^2 D}{\partial y} + \bar{F}_y \quad (22b)$$

Note that, if A_M is constant in the vertical, then the "F" terms in (22a) and (22b) cancel. However, we account for possible vertical variability in the horizontal diffusivity; such is the case when a Smagorinsky type diffusivity is used. As detailed below, all of the terms on the right side of (18) and (19) are evaluated at each internal time step and then held constant throughout the many external time steps. If the external mode is executed *cum sole*, then $G_x = G_y = 0$.

3. FORTRAN SYMBOLS

In the following table, we list the FORTRAN symbols followed by their corresponding analytical symbols in parentheses and a brief description of the symbols. Not explicitly tabulated are the suffixes B, blank and F which are appended to many of the variables to denote the time levels $n - 1$, n and $n + 1$ respectively.

Indices

I, J (i, j)	horizontal grid indexes
IM, JM	outer limits of I and J
K (k)	vertical grid index; K = 1 at the top and K = KB at the bottom
IINT (n)	internal mode time step index
IEXT	external mode time step index

Constants

DTE (Δt_E)	external mode time step, (s)
DTI (Δt_I)	internal mode time step, (s)
EXTINC	short wave extinction coefficient, (m^{-1})
HORCON(C)	the coefficient of the Smagorinsky diffusivity
IEND	total internal mode time steps
IPRINT	the interval in IINT at which variables are printed
ISPLIT	DTI/DTE
MODE	if MODE = 2, a 2-D calculation is performed if MODE = 3, a 3-D prognostic calculation is performed if MODE = 4, a 3-D diagnostic calculation is performed
RFE, RFW, RFN, RFS	= 1 or 0 on the four open boundaries; for use in BCOND
SMOTH (α)	parameter in the temporal smoother
TPRNI (A_H/A_M)	inverse, horizontal, turbulence Prandtl number
TR	short wave surface transmission coefficient
UMOL	background vertical diffusivity

One-dimensional Arrays

Z(σ)	sigma coordinate which spans the domain, Z = 0 (surface) to Z = -1 (bottom)
---------------	---

ZZ	sigma coordinate, intermediate between Z
DZ($\delta\sigma$)	= Z(K)-Z(K+1)
DZZ	= ZZ(K)-ZZ(K+1)

Two-dimensional Arrays

AAM2D	vertical average of AAM(m ² s ⁻¹)
ART, ARU, ARV	cell areas centered on the variables, T, U and V respectively (m ²)
ADVUA, ADVVA	sum of the second, third and fourth terms in equations (18,19)
ADX2D, ADY2D	vertical integrals of ADVX, ADVY; also the sum of the fourth, fifth and sixth terms in equations (22a,b)
COR (f)	the Coriolis parameter (s ⁻¹)
CURV2D	the vertical average of CURV
DUM	Mask for the u component of velocity; = 0 over land; = 1 over water
DVM	Mask for the v component of velocity; = 0 over land; = 1 over water
FSM	Mask for scalar variables; = 0 over land; = 1 over water
DX (h_x or δx)	grid spacing (m)
DY (h_y or δy)	grid spacing (m)
EL (η)	the surface elevation as used in the external mode (m)
ET (η)	the surface elevation as used in the internal mode and derived from EL (m)
EG (η)	the surface elevation also used in the internal mode for the pressure gradient and derived from EL (m)
D (D)	= H + EL (m)
DT (D)	= H + ET (m)
DRX2D, DRX2D	vertical integrals of DRHOX and DRHOY
H (H)	the bottom depth (m)
SWRAD	short wave radiation incident on the ocean surface (m s ⁻¹ K)

UA, VA (\bar{U}, \bar{V})	vertical mean of U, V (m s^{-1})
UT, VT (\bar{U}, \bar{V})	UA, VA time averaged over the interval, $DT = DTI$ (m s^{-1})
WUSURF, WVSURF	($\langle wu(0) \rangle, \langle wv(0) \rangle$) momentum fluxes at the surface (m^2s^{-2})
WUBOT, WUBOT	($\langle wu(-1) \rangle, \langle wv(-1) \rangle$) momentum fluxes at the bottom (m^2s^{-2})
WTSURF, WSSURF	($\langle w\theta(0) \rangle, \langle ws(0) \rangle$) temperature and salinity fluxes at the surface ($\text{ms}^{-1} \text{K}, \text{ms}^{-1} \text{psu}$)

Three-dimensional Arrays

ADVX, ADVY	horizontal advection and diffusion terms in equations (3) and (4)
AAM (A_M)	horizontal kinematic viscosity ($\text{m}^2 \text{s}^{-1}$)
AAH (A_H)	horizontal heat diffusivity = TPRNI*AAM
CURV (\tilde{f})	curvature terms; see equation (28)
L (ℓ)	turbulence length scale
KM (K_M)	vertical kinematic viscosity (m^2s^{-1})
KH (K_H)	vertical diffusivity (m^2s^{-1})
DRHOX	x-component of the internal baroclinic pressure gradient $\left(gDh_y\rho_o^{-1} \left[-D \int_{\sigma}^0 \delta_x \rho' \delta\sigma' + \delta_x D \int_{\sigma}^0 \sigma' \delta\rho' \right] \right)$ subtract RMEAN from density before integrating
DRHOY	y-component of the internal baroclinic pressure gradient $\left(gDh_x\rho_o^{-1} \left[-D \int_{\sigma}^0 \delta_y \rho' \delta\sigma' + \delta_y D \int_{\sigma}^0 \sigma' \delta\rho' \right] \right)$ subtract RMEAN from density before integrating
RAD (R)	short wave radiation flux (ms^{-1}K). Sign is the same as WTSURF
Q2 (q^2)	twice the turbulence kinetic energy (m^2s^{-2})
Q2L ($q^2 \ell$)	Q2 x the turbulence length scale (m^3s^{-2})
T (T)	potential temperature (K)
S (S)	salinity (psu)
RHO ($\rho/\rho_o - 1.025$)	density (non-dim.)
U, V (U, V)	horizontal velocities (m s^{-1})
W (ω)	sigma coordinate vertical velocity (m s^{-1})

RMEAN	density field which is horizontally averaged before transfer to sigma coordinates.
TCLIM	a stationary temperature field which approximately has the same vertical structure as T.
SCLIM	a stationary salinity field which approximately has the same vertical structure as S.

The variables UF and VF are used to denote the $n+1$ time level for U and V respectively. However, in order to save memory they are also used to represent the $n+1$ time level for T and S and for $Q2$ and $Q2L$ respectively. As soon as UF , VF are calculated for each representation, the time level is reset.

4. THE NUMERICAL SCHEME

Figure 2 is the flow chart for the program in simplified form. The external mode calculation is contained in PROGRAM MAIN.

External-Internal Mode Interaction.

The external mode calculation in MAIN results in updates for surface elevation, EL , and the vertically averaged velocities, UA , VA . The internal mode calculation results in updates for U, V, T, S and the turbulence quantities.

Fig. 3 illustrates the time stepping process for the external and internal mode. Assume everything is known at t^{n-1} and t^n (the previous leap frog time step having just been completed). Integrals involving the baroclinic forcing and the advective terms are supplied to the external mode along with the bottom stress, a process which is labeled "Feedback" in Fig. 3; their values are held constant during $t^n < t < t^{n+1}$. The external mode "leap frogs" many times, with the time step, DTE , until $t = t^{n+1}$. The vertical and time averaged velocities, UTF , VTF , and those from the previous time step, UTB, VTB , are time averages of the external variables, UA, VA . The internal and external modes have different truncation errors so that the vertical integrals of the internal mode velocity may depart slightly from (UA, VA) during the course of a long integration. We therefore adjust the internal velocities, U, V , so that their vertical integrals are the mean of UTF, VTF and UTB, VTB . Care is taken to relate ETF to ELF so that together with ETB , saved from a previous time step, the internal velocities and ETF and ETB correctly satisfy the continuity equation, (17). Otherwise, the sigma coordinate equations for T, S will not be conservative.

Aside from the above, numerically important details, $0.5*(EGF + EGB)$ is used to obtain the elevation gradients for the internal mode "leap frog" from t^{n-1} to t^{n+1} . EGF and

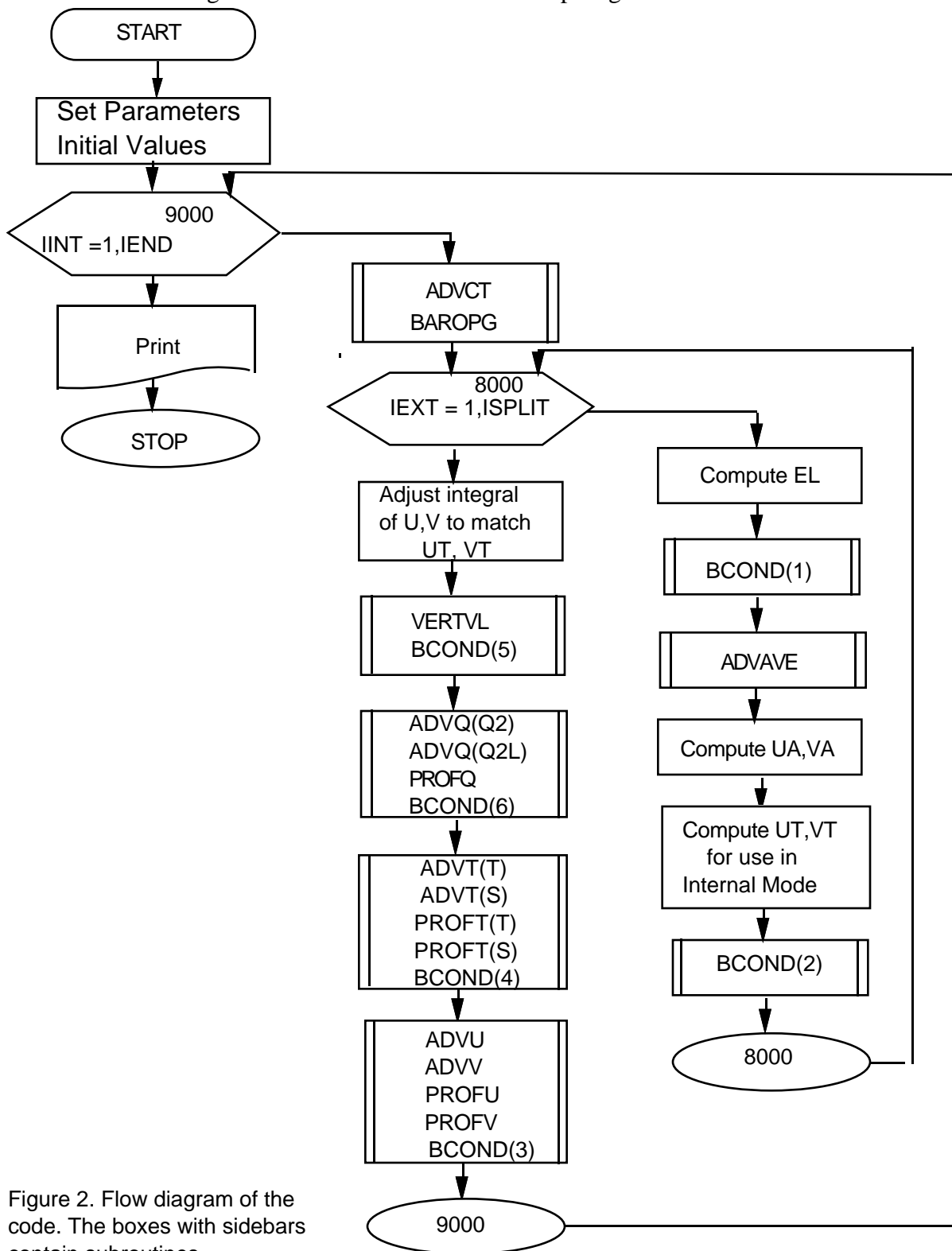


Figure 2. Flow diagram of the code. The boxes with sidebars contain subroutines.

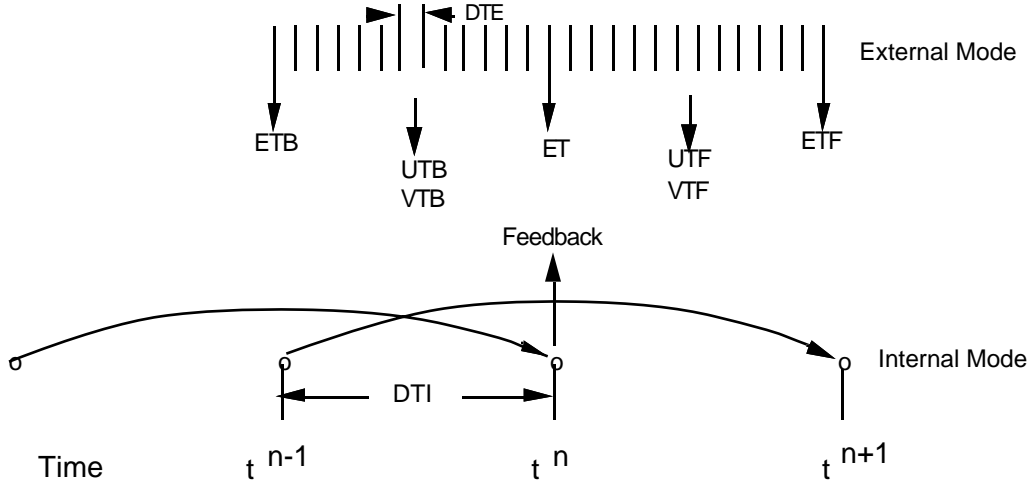


Figure 3. A simplified illustration of the interaction of the External Mode and the Internal Mode. The former uses a short time step, DTE, whereas the latter uses a long time step, DTI. The external mode primarily provides the surface elevation to the internal mode whereas, as symbolized by "Feedback", the internal mode provides integrals of momentum advection, density integrals and bottom stress to the external mode.

EGB are EL averaged over the intervals, t^n to t^{n+1} and t^{n-1} to t^n , respectively. It is this maneuver that renders the internal mode immune to the CFL condition based on the barotropic wave speed. The governing wave speed is the baroclinic wave speed.

Structure of the Internal Mode Calculation.

The calculation of the three-dimensional (internal) variables is separated into a vertical diffusion time step and an advection plus horizontal diffusion time step. The former is implicit (to accommodate small vertical spacing near the surface) whereas the latter is explicit. To illustrate, consider the temperature equation,

$$\frac{\partial DT}{\partial T} + Adv(T) - Dif(T) = \frac{1}{D} \frac{\partial}{\partial \sigma} \left(K_H \frac{\partial T}{\partial \sigma} \right) - \frac{\partial R}{\partial \sigma} \quad (23)$$

$Adv(T)$ and $Dif(T)$ represents the advection and horizontal diffusion terms. The solution is carried out in two steps. Thus, the advection and horizontal diffusion parts are differenced according to

$$\frac{D^{n+1} \tilde{T} - D^{n-1} T^{n-1}}{2\Delta t} = - Adv(T^n) + Dif(T^{n-1}) \quad (24)$$

and is solved by SUBROUTINE ADVT. The vertical diffusion part is differenced according to

$$\frac{D^{n+1}T^{n+1} - D^{n+1}\tilde{T}}{2\Delta t} = \frac{1}{D^{n+1}} \frac{\partial}{\partial \sigma} \left(K_H \frac{\partial T^{n+1}}{\partial \sigma} \right) - \frac{\partial R}{\partial \sigma} \quad (25)$$

and is solved by SUBROUTINE PROFT as detailed in section 9 wherein (25) is first divided by D^{n+1} . Note that, in this subroutine, T^{n-1} is stored in TB, T^n in T and T^{n+1} in UF.

In the "leap frog" time differencing scheme, the solutions at odd time steps can diverge slowly from the solutions at the even time steps. This time splitting is removed by a weak filter (Asselin, 1972) where the solution is smoothed at each time step according to

$$T_s = T + \frac{\alpha}{2} (T^{n+1} - 2T^n + T^{n-1})$$

where T_s is the smoothed solution; frequently, we use $\alpha = 0.05$. This technique introduces less damping than either the Euler-backward or forward stepping techniques. After smoothing, T_s is reset to T^{n-1} and T^{n+1} to T^n .

Grid Arrangement

The staggered grid arrangement for the external mode is depicted in Fig. 4 and 5 for the external and internal grid respectively. These diagrams will be useful in understanding the coding in MAIN and in the "PROF" and "ADV" subroutines. Although the fortan nomenclature in the code may appear to be cartesian coordinates, the grid can be an orthogonal curvilinear grid. One merely needs to specify $h_x(=DX(I, J))$ and $h_y(=DY(I, J))$ as that associated with a particular grid. The advective operators in equations (2) to (8) and (17) to (19) are then described in a finite volume sense; i.e. Equation (5) or, rather, the Adv operator in (24), is written

$$-Adv(T)h_xh_y = \delta_x(Dh_yUT) + \delta_y(Dh_xVT) + h_xh_y \frac{\delta_\sigma(\omega T)}{\delta \sigma} \quad (26)$$

(where it might be more consistent to multiply through by $\delta \sigma$, but this has not been effected in the code). Thus Dh_yUT represents the transport of T and δ_x represents the difference in this quantity through the opposing faces of the volume element. We leave it to the code listing in SUBROUTINE ADVT to describe the exact method of differencing.

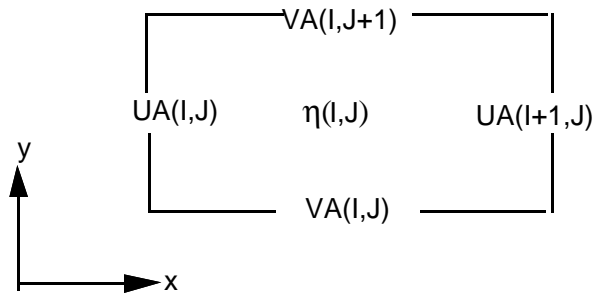


Figure 4. The two-dimensional external mode grid.

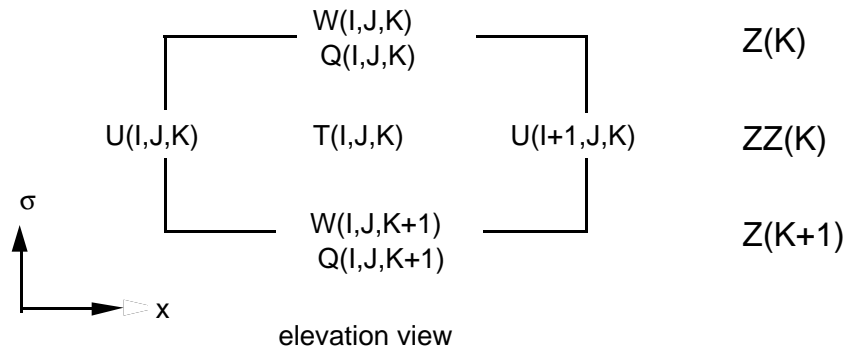
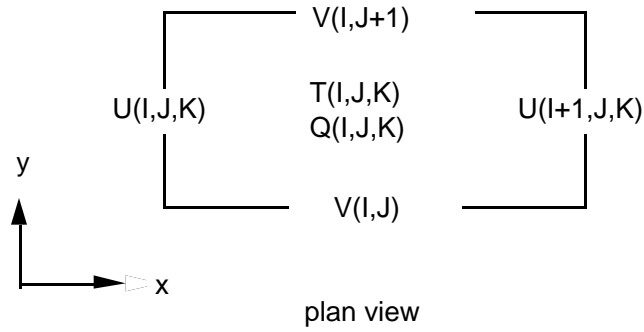


Figure 5. The three-dimensional internal mode grid. Q represents KM, KH, Q2, or Q2L. T represents T,S or RHO.

The differencing for the velocity is accomplished in a similar way but involves Coriolis and curvature terms. The advective term for U in equation (3) is

$$-Adv(U)h_xh_y = \delta_x(Dh_yUU) + \delta_y(Dh_xUV) + h_xh_y \frac{\delta\sigma(\omega U)}{\delta\sigma} - \tilde{f}VD h_xh_y \quad (27)$$

where

$$\tilde{f} = \frac{V\delta_x(h_y)}{h_xh_y} - \frac{U\delta_y(h_x)}{h_xh_y} \quad (28)$$

is the curvature term. In ADVCT, the horizontal advection, diffusion and curvature terms are calculated (and stored in ADVX, ADVY) well in advance of ADVU and ADVV so that their vertical averages can be used in the external mode calculation. In ADVU and ADVV, the pressure gradient, Coriolis and vertical advection are included along with the terms imported from ADVCT.

Time Step Constraints.

The Courant-Friedrichs-Levy (CFL) computational stability condition on the vertically integrated, external mode, transport equations limits the time step according to

$$\Delta t_E \leq \frac{1}{C_t} \left| \frac{1}{\delta x^2} + \frac{1}{\delta y^2} \right|^{-1/2} \quad (29)$$

where $C_t = 2(gH)^{1/2} + U_{max}$; U_{max} is the expected, maximum velocity. There are other restrictions but in practice the CFL limit is the most stringent. The model time step is usually 90% of this limit. The internal mode has a much less stringent time step since the fast moving external mode effects have been removed. The time step criteria is analogous to that for the external mode given by Equation (26) and is

$$\Delta t_I \leq \frac{1}{C_T} \left| \frac{1}{\delta x^2} + \frac{1}{\delta y^2} \right|^{-1/2} \quad (30)$$

where $C_T = 2C + U_{max}$; C_T is the maximum internal gravity wave speed based on the gravest mode, commonly of order 2m/s, and U_{max} is the maximum advective speed. For typical coastal ocean conditions the ratio of the time steps, $\Delta t_I / \Delta t_E = DTI/DTE$, is often a factor of 50 - 80 or larger.

Additional limits are imposed by horizontal diffusion of momentum or scalars are, for $A = A_M$ or $A = A_H$

$$\Delta t_I \leq \frac{1}{4A} \left| \frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right|^{-1} \quad (31)$$

A limit imposed by rotation is

$$\Delta t_I < \frac{1}{f} = \frac{1}{2\Omega \sin \Phi} \quad (32)$$

where A_H is the horizontal diffusivity, Ω is the angular velocity of the earth and Φ is the latitude. (31) and (32) are, however, not restrictive compared to (29) and (30).

5. comblk.h

Common block definitions are contained in the file, comblk.h. The file is then "include"d in each subroutine.

6. PROGRAM MAIN and the external mode

The main program contains model initialization and, subsequently, internal mode time stepping via the index, IINT. All of the internal mode (three-dimensional) subroutines, ADVQ, PROFQ, ADVU, PROFU, ADVV, PROFV, ADVT (for temperature or salinity), PROFT (for temperature or salinity) and DENS are called once for each value of IINT = 1 to IEND.

Imbedded in the IINT loop (which terminates at S.N. 9000) is the external mode IEXT loop which cycles ISPLIT times. Note that DTI/DTE = ISPLIT. The external mode solves for the vertically mean velocities and surface elevation using the density created by the internal mode which is held constant throughout the many external mode, time steps.

The advective and horizontal diffusion terms in the external model are calculated by vertical integration of the corresponding internal terms, ADVX and ADVY, created in subroutine ADVCT. The latter are available every internal time step. However, they are updated by virtue of similar terms (but derived from the mean velocity) in SUBROUTINE ADVAVE. We find that this need not be done every external time step to maintain a stable calculation. SUBROUTINE ADVAVE which calculates these terms are called at intervals of ISPADV; a typical value is ISPADV = 5.

7. SUBROUTINE ADVAVE

This subroutine calculates the advective and horizontal diffusion terms for the external mode calculation contained in equations (18) and (19). If $MODE = 2$, it also calculates the bottom friction from a quadratic drag equation; otherwise, in the standard three dimensional calculation, the bottom friction is determined by PROFU and PROFV quite naturally as a byproduct of the bottom boundary layer.

8. SUBROUTINE ADVT

This subroutine solves equation (24) for temperature or salinity (or any other scalar variable) which are labeled F internally. \tilde{D} has been set to D^{n+1} . The operator, $Adv(F)$, is written in the form of equation (26). As shown in the code listing, horizontal advective transports through the faces of the grid elements are computed in the form, $D*U*F*DY$ and $D*V*F*DX$, using appropriate cell averages. Note that, in the code listing, DT is simply the external value, D, averaged over the internal time step. To the advective fluxes are added the horizontal diffusion fluxes. Before this occurs, TCLIM is subtracted from the actual temperature [see Mellor and Blumberg, 1986, and discussion after equation (12a, b)]. Then, the diffusion terms will (gently) drive the calculated field back to climatology. As resolution improves, the diffusion terms decrease as DX DY decreases. The vertical advective flux divergence is determined (and temporally stored in FF) and then combined with the horizontal transport divergence. Finally, the time step is executed and the new value is stored in FF.

9. SUBROUTINE PROFT

This subroutine solves (25) for temperature and salinity. We use the method described on p.198-201 of Richtmeyer and Morton (1967). The procedure here will be a model for U,V,Q2 and Q2L in which case the radiation term in (25) is either null or is replaced by source/sink terms. Subroutine PROFT as well as ADVT can be used to solve for other geochemical constituents besides temperature and salinity.

First, finite difference (25) with respect to σ . [We note that $\tilde{D} = D^{n+1}$; the choice is irrelevant so long as the same value of \tilde{D} is used in (24) and (25)]. Thus, with reference to the elevation view of Fig. 5.

$$F_k - \tilde{F}_k = \frac{DTI2}{D^{**2} * DZ_k} \left[\frac{KH_k}{DZZ_{k-1}} * (F_{k-1} - F_k) - \frac{KH_{k+1}}{DZZ_k} * (F_k - F_{k+1}) \right] - \frac{DTI2}{D * DZ_k} [RAD_k - RAD_{k+1}] \quad (9-1)$$

where $DZ_k = Z_k - Z_{k+1}$, $DZZ_k = ZZ_k - ZZ_{k+1}$ and F_k represents either temperature or salinity. In the above, we use subscripts for k instead of parenthetical enclosure to save space; we also omit the I, J indicies.

Solution Technique

Equation (9-1) can now be written as

$$-F_{k+1} * A_k + F_k * (A_k + C_k - 1) - F_{k-1} * C_k = D_k \quad (9-2)$$

where

$$A_k = - \frac{DTI2 * KH_{k+1}}{D^{**2} * DZ_k * DZZ_k} \quad (9-3a)$$

$$C_k = - \frac{DTI2 * KH_k}{D^{**2} * DZ_k * DZZ_{k-1}} \quad (9-3b)$$

$$D_k = - \tilde{F}_k + \frac{DTI2}{D * DZ_k} [RAD_k - RAD_{k+1}] \quad (9-3c)$$

Now assume a solution of the form

$$F_k = E_k * F_{k+1} + G_k \quad (9-4)$$

Inserting F_{k+1} directly from (9-4) and F_{k-1} , obtained from (9-4), into (9-2) and collecting coefficients of F_k and 1 yields

$$E_k = \frac{A_k}{A_k + C_k * (1 - E_{k-1}) - 1} \quad (9-5a)$$

$$G_k = \frac{C_k * G_{k-1} + D_k}{A_k + C_k * (1 - E_{k-1}) - 1} \quad (9-5b)$$

The way the system works is as follows: All A_k 's, C_k 's and D_k 's are calculated from (9-3a,b,c). Surface boundary conditions, discussed below, provide E_1 and G_1 and all of the

necessary E_k 's and G_k 's are obtained from the descending (as k increases towards the bottom) recursive relations (9-5a,b). Bottom boundary conditions provide F_{kb-1} where $kb-1$ is the grid point nearest the bottom. Thereafter all of the F_k 's may be obtained from the ascending recursive relation (9-4).

Short Wave Radiation

To specify the short wave radiation, we use the classification of Jerlov(1976), but approximate his methodology such that short wave radiation, absorbed in the first several meters, is added to the surface boundary condition. Thus, we will shortly install $WTSURF + (1.-TR)*SWRAD$ as the surface boundary condition so that the remainder is attenuated according to

$$RAD_k = SWRAD * TR * EXP(EXTINC * Z_k * D) \quad (9-6)$$

A very good fit to Jerlov's tabulated attenuation function is given by (9-6) if we set the constants TR and EXTINC according to the following table:

NTP	TYPE	TR	EXTINC(m ⁻¹)
1	I	.32	.037
2	IA	.31	.042
3	IB	.29	.056
4	II	.26	.073
4	III	.24	.127

Surface and Bottom Boundary Conditions

To apply the surface boundary conditions where the surface flux is prescribed (prescribing the surface temperature is much simpler since $E_1 = 0$, $G_1 = F_1$) begins with (9-1) where for $k=1$

$$F_1 - \tilde{F}_1 = \frac{DTI2}{D * DZ_1} (-WTSURF - (1.-TR) * SWRAD) + A_1(F_1 - F_2)$$

Using (9-4) to eliminate F_2 and collecting coefficients of F_1 and 1 yields

$$E_1 = \frac{A_1}{A_1 - 1} \quad (9-7a)$$

$$G_1 = \left[\frac{DTI2 * (WTSURF + (1.-TR) * SWRAD)}{D * DZ_1} - \tilde{F}_1 \right] * \left[\frac{1}{A_1 - 1} \right] \quad (9-7b)$$

At the bottom, we specify zero heat flux. A repeat of the above procedure leads to

$$F_{kb-1} = \frac{C_{kb-1} * G_{kb-2} - \tilde{F}_{kb-1}}{C_{kb-1} * (1 - E_{kb-1}) - 1} \quad (9-8)$$

Four different surface boundary conditions can be selected by choosing the appropriate NBC parameter when calling PROFT:

NBC=1 - surface BC is WTSURF or WSSURF (heat or salt flux BC)

NBC=2 - surface BC is WTSURF+SWRAD (heat flux and short wave radiation penetration)

NBC=3 - surface BC is TSURF or SSURF (SST or SSS BC)

NBC=4 - surface BC is TSURF+SWRAD (SST and short wave radiation penetration)

Note that WTSURF and SWRAD are *negative* when water column is *warming*. (To transfer values of heat flux given in Wm^{-2} to WTSURF in Km/s , divide by the factor 4.1876×10^6).

10. SUBROUTINE BAROPG

This subroutine calculates the baroclinic, vertical integrals involving density in equation (3) and (4) after the equations have been written in finite volume form.

We note the fact that, in the code, $\bar{\rho} = RMEAN$ has been subtracted from ρ before the integrals are calculated. $\bar{\rho}$ is the basin area average density which is mapped onto the sigma-grid just as the initial conditions were similarly mapped. This procedure removes most of the truncation error in the transformed baroclinic terms which arise due to the subtraction of the two large terms involving $\partial\rho / \partial x$ and $D^{-1}(\partial D / \partial x)\sigma\partial\rho / \partial\sigma$ in (3) and similarly in (4)

11. SUBROUTINES ADVCT, ADVU AND ADVV

ADVCT calculates the horizontal advection (including curvature terms) and the diffusion parts of (3) and (4) which are differenced in the manner of equation (27) and saved as ADVX and ADVY. These terms are vertically integrated and saved as ADX2D and ADY2D for use in the external mode calculation in program MAIN. Originally, ADVCT had been incorporated in ADVU and ADVV. However, it was determined by

Oregon State University colleagues that advancing the calculation of horizontal advection terms (see Figure 2) for use in the external mode increased the model's intrinsic stability.

12. SUBROUTINES PROFU AND PROFV

These subroutines are virtually identical to SUBROUTINE PROFT. However, the bottom boundary conditions are obtained from equations (14c,d,e).

13. SUBROUTINE ADVQ

This subroutine is very similar to all the other "ADV-" subroutines in that it calculates the advective terms for the the turbulence quantities, Q2 and Q2L.

14. SUBROUTINE PROFQ

This subroutine first solves for the vertical part of the equations (7) and (8) for Q2 and Q2L in the manner of equation (25). The numerical procedure is the same as SUBROUTINE PROFT. The turbulence closure scheme as described by Mellor and Yamada(1982) is contained in this subroutine. A somewhat simplified version of the level 2 1/2 model is used here and is discussed in Galperin *et al* (1988) and Mellor (1989).

The vertical diffusivities, K_M and K_H , are defined according to

$$K_M = q\ell S_M \quad (14-1a)$$

$$K_H = q\ell S_H \quad (14-1b)$$

The coefficients, S_M and S_H , are functions of a Richardson number given by

$$S_H[1 - (3A_2B_2 + 18A_1A_2)G_H] = A_2[1 - 6A_1 / B_1] \quad (14-2a)$$

$$S_M[1 - 9A_1A_2G_H] - S_H[(18A_1^2 + 9A_1A_2)G_H] = A_1[1 - 3C_1 - 6A_1 / B_1] \quad (14-2b)$$

where

$$G_H = - \frac{\ell^2}{q^2} \frac{g}{\rho_o} \left[\frac{\partial \rho}{\partial z} - \frac{1}{c_s^2} \frac{\partial p}{\partial z} \right] \quad (14-2c)$$

is a Richardson number. The five constants in (14-2a,b) are mostly evaluated from near surface turbulence data (law-of-the-wall region) and are found (Mellor and Yamada, 1982)

to be $(A_1, A_2, B_1, B_2, C_1) = (0.92, 16.6, 0.74, 10.1, 0.08)$. The stability functions limit to infinity as G_H approaches the value, 0.0288, a value larger than one expects to find in nature. The quantity, c_s^2 , in the square brackets of (14-2c) is the speed of sound squared. In the code the vertical pressure gradient is obtained from the hydrostatic relation, of course, but here, the density is taken as a constant consistent with the pressure determination in SUBROUTINE DENS; i.e., $\partial p / \partial z = - \rho_o g$

15. SUBROUTINE VERTVL

This short subroutine integrates equation (2) to obtain the sigma coordinate transformed "vertical velocity" which, actually, is the velocity normal to sigma surfaces. Occasionally, check $W(I, J, KB)$; if all is well, the code should yield very small values ($\sim 10^{-11}$).

16. SUBROUTINE BCOND

Lateral boundary conditions contiguous to coastlines are handled automatically by the masks DUM, DVM and FSM. They set to zero the velocities normal to land boundaries. The landward tangential velocities in the horizontal friction terms are also set to zero. For a sigma coordinate system, the latter is of little importance since the minimum water depth next to the coast can be quite shallow so that bottom friction dominates over lateral friction. We generally set the minimum depth in the range, 10 to 20 m, but a numerically acceptable minimum depth has not been firmly established.

Open boundaries are considerably more demanding and uncertain and there is a need for boundary condition specification for both the external and internal modes.

Table A collects a variety of open boundary conditions; they are by no means inclusive. If (A - 1) is used around all open boundaries, then it is necessary to insure that the horizontal integral of BC around the boundary is zero; otherwise, the average basin elevation can increase or decrease, possibly disastrously. This can also happen with the exclusive use of (A - 4) or (A - 5).

Calculations do not seem overly sensitive to the velocity component tangential to the boundary, at least for low Rossby number flows. We often set it to zero; alternatively advective boundary condition similar to (B - 3) has been used.

Table A: A list of possible external mode open boundary conditions. In the formulations, $c_e = \sqrt{gH}$. The variable BC is user specified and may be equated to the left sides of (A-1) - (A-3) where \bar{U} and η are known *a priori*. The right sides of (A-4) and (A-5) need not necessarily be zero. This table greatly augmented from the original by Peter Holloway (School of Geography and Oceanography, University College, University of New South Wales, Australian Defence Force Academy, Australia) and edited by George Mellor. The table does not exhaust the list of possible boundary conditions. Please report errors.

Formula	Boundary	Code
Inflow condition: $D\bar{U} = BC$ (A - 1)	EAST	UAF(IM,J) = 2*BC(J)/(H(IM,J)+ELF(IM,J) + H(IMM1,J) + ELF(IMM1,J)) ELF(IM,J) = ELF(IMM1,J) VAF(IM,J) = set ¹
	WEST	UAF(2,J) = 2*BC(J)/(H(1,J)+ELF(1,J) + H(2,J)+ELF(2,J)) ELF(1,J) = ELF(2,J) VAF(1,J) = set
	NORTH	VAF(I,JM) = 2*BC(I)/(H(I,JM)+ELF(I,JM) + H(I,JMM1) + ELF(I,JMM1)) ELF(I,JM) = ELF(I,JMM1) UAF(I,JM) = set
	SOUTH	VAF(I,2) = 2*BC(I)/(H(I,1)+ELF(I,1) + H(I,2)+ELF(I,2)) ELF(I,1) = ELF(I,2) UAF(I,1) = set
Elevation condition: $\eta = BC$ (A - 2)	EAST	ELF(IMM1,J) = BC(J) ELF(IM,J) = ELF(IMM1,J) cosmetic UAF(IM,J) = UAF(IMM1,J) VAF(IM,J) = set
	WEST	ELF(2,J) = BC(J) UAF(2,J) = UAF(3,J) VAF(1,J) = set
	NORTH	ELF(I,JMM1) = BC(I) ELF(I,JM) = ELF(I,JMM1) cosmetic VAF(I,JM) = VAF(I,JMM1) UAF(I,JM) = set
	SOUTH	ELF(I,2) = BC(I) VAF(I,2) = VAF(I,3) UAF(I,1) = set

¹ We use "set" to denote the prescription for the along-boundary component of velocity. If it is a known value then that value can be used. More often it is not known and the value, 0, is used.

Radiation: $H\bar{U} \pm c_e \eta = BC^2$ (A-3)	EAST	UAF(IM,J) = SQRT(GRAV/H(IMM1,J))* EL(IMM1,J) + BC(J) ELF(IM,J) = ELF(IMM1,J) VAF(IM,J) = set
	WEST	UAF(2,J) = - SQRT(GRAV/H(2,J))* EL(2,J)+BC(J) ELF(1,J) = ELF(2,J) VAF(1,J) = set
	NORTH	VAF(I,JM) = SQRT(GRAV/H(I,JMM1))* EL(I,JMM1) + BC(I) ELF(I,JM) = ELF(I,JMM1) UAF(I,JM) = set
	SOUTH	VAF(I,2) = - SQRT(GRAV/H(I,2))* EL(I,2)+BC(I) ELF(I,1) = ELF(I,2) UAF(I,1) = set
Radiation: $\frac{\partial \bar{U}}{\partial t} \pm c_e \frac{\partial \bar{U}}{\partial x} = 0$ (A-4)	EAST	GAE = DTE*SQRT(GRAV*H(IM,J))/DX(IM,J) UAF(IM,J) = GAE*UA(IMM1,J) + (1.-GAE)*UA(IM,J) ELF(IM,J) = ELF(IMM1,J) VAF(IM,J) = set
	WEST	GAE = DTE*SQRT(GRAV*H(2,J))/DX(2,J) UAF(2,J) = GAE*UA(3,J) + (1.-GAE)*UA(2,J) ELF(1,J) = ELF(2,J) VAF(1,J) = set
	NORTH	GAE = DTE*SQRT(GRAV*H(I,JM))/DX(I,JM) VAF(I,JM) = GAE*VA(I,JMM1) + (1.-GAE)*VA(I,JM) ELF(I,JM) = ELF(I,JMM1) UAF(I,JM) = set
	SOUTH	GAE = DTE*SQRT(GRAV*H(I,2))/DX(I,2) VAF(I,2) = GAE*VA(I,3) + (1.-GAE)*VA(I,2) ELF(I,1) = ELF(I,2) UAF(I,1) = set

Table B are open boundary conditions for the internal mode.

As in the external mode, the choice for the normal velocities is unclear. One might presume that (B - 2) is to be preferred over (B - 1) since internal waves can pass through the boundary with little reflection. In some applications, that may be the case. However, we have seen cases (open boundaries with substantial inflows) where the "freedom" of (B - 2) can set up unphysical, but numerically valid, baroclinic structures interior to the boundary.

² The boundary forcing can be set to known values approximately balancing the left side; e. g., on the east $BC(J) = UABE(J) - \text{SQRT}(\text{GRAV}/H(\text{IMM1},J)) * \text{ELE}(J)$ where $UABE(J)$ and $\text{ELE}(J)$ are specified values.

Radiation: $\frac{\partial \eta}{\partial t} \pm c_e \frac{\partial \eta}{\partial x} = 0$ (A-5)	EAST	GAE = DTE*SQRT(GRAV*H(IMM1,J))/DX(IMM1,J) ELF(IMM1,J) = GAE*EL(IMM2,J) + (1.-GAE) *EL(IMM1,J) ELF(IM,J) = ELF(IMM1,J) UAF(IM,J) = UAF(IMM1,J) VAF(IM,J) = set
	WEST	GAE = DTE*SQRT(GRAV*H(2,J))/DX(1,J) ELF(2,J) = GAE*EL(3,J) + (1.-GAE)*EL(2,J) UAF(2,J) = UAF(3,J) VAF(2,J) = set
	NORTH	GAE = DTE*SQRT(GRAV*H(I,JMM1))/DX(I,JMM1) ELF(I,JMM1) = GAE*EL(I,JMM2) + (1.-GAE) *EL(I,JMM1) ELF(I,JM) = ELF(I,JMM1) VAF(I,JM) = VAF(I,JMM1) UAF(I,JM) = set
	SOUTH	GAE = DTE*SQRT(GRAV*H(I,2))/DX(I,2) ELF(I,2) = GAE*EL(I,3) + (1.-GAE)*EL(I,2) VAF(I,2) = VAF(I,3) UAF(I,1) = set
Cyclic (A-6)	EAST (I=IM)	ELF(IM,J) = ELF(3,J) UAF(IM,J) = UAF(3,J) VAF(IM,J) = VAF(3,J)
	WEST (I=1)	ELF(1,J) = ELF(IMM2,J) ELF(2,J) = ELF(IMM1,J) UAF(2,J)=UAF(IMM1,J) VAF(2,J)=VAF(IMM1,J)
	NORTH (J=JM)	ELF(I,JM) = ELF(I,3) UAF(I,JM) = UAF(I,3) VAF(I,JM) = VAF(I,3)
	SOUTH (J=1)	ELF(I,1) = ELF(I,JMM2) ELF(I,2) = ELF(I,JMM1) UAF(I,2)=UAF(I,JMM1) VAF(I,2)=VAF(I,JMM1)

The finite difference expression one gets for the EAST version of (B - 2) is

$$U_{im}^{n+1} = \gamma U_{im-1}^n + (1 - \gamma)U_{im}^n; \quad \gamma \equiv c_i \Delta t_i / \Delta x$$

where one might like c_i to be the gravest mode, baroclinic phase speed. However, it is assumed that: *a*) the user has found and is using a Δt_i such that the maximum value of γ is near unity, corresponding approximately to the maximum depth and *b*) that c_i is proportional to \sqrt{H} . This is a seemingly crude approximation, but may perform fairly well; it at least guarantees that $0 < \gamma \leq 1$.

TABLE B: A list of internal mode variables to be set on open lateral boundaries and example boundary conditions. Note that UF and VF are used for the forward time step of U and V, T and S, and Q2 and Q2L. The variables TBE, TBW, TBN, TBS (and similar variables for salinity) are supplied by the user

Formula	Boundary	Code
Inflow condition: $U = BC$ (B-1)	EAST	UF(IM,J,K) = BC(J,K) VF(IM,J,K) = set
	WEST	UF(2,J,K) = BC(J,K) VF(1,J,K) = set
	NORTH	VF(I,JM,K) = BC(I,K) UF(I,JM,K) = set
	SOUTH	VF(I,2,K) = BC(I,K) UF(I,1,K) = set
Radiation: $\frac{\partial U}{\partial t} \pm c_i \frac{\partial U}{\partial x} = 0$ (B-2)	EAST	GAI = SQRT(H(IM,J)/HMAX) ³ UF(IM,J,K) = GAI*U(IMM1,J,K) + (1.-GAI)*U(IM,J,K) VF(IM,J,K) = set
	WEST	GAI = SQRT(H(2,J)/HMAX) UF(2,J,K) = GAI*U(3,J,K) + (1.-GAI)*U(2,J,K) VF(1,J,K) = set
	NORTH	GAI = SQRT(H(I,JM)/HMAX) VF(I,JM,K) = GAI*V(I,JMM1,K) + (1.-GAI)*V(I,JM,K) UF(I,JM,K) = set
	SOUTH	GAI = SQRT(H(I,2)/HMAX) VF(I,2,K) = GAI*V(I,3,K) + (1.-GAI)*V(I,2,K) UF(I,1,K) = set
Upstream advection on T or S: $\frac{\partial T}{\partial t} + U \frac{\partial T}{\partial x} = 0$ (B-3)	EAST	UF(IM,J,K) = T(IM,J,K) -DTI/(DX(IM,J)+DX(IMM1,J)) * ((U(IM,J,K) + ABS(U(IM,J,K))) * (T(IM,J,K)-T(IMM1,J,K)) + (U(IM,J,K) - ABS(U(IM,J,K))) * (TBE(J,K)-T(IM,J,K)))
	WEST	UF(1,J,K) = T(1,J,K) -DTI/(DX(1,J)+DX(2,J)) * ((U(1,J,K) + ABS(U(1,J,K))) * (T(1,J,K)-TBW(J,K)) + (U(1,J,K) - ABS(U(1,J,K))) * (T(2,J,K)-T(1,J,K)))
	NORTH	UF(I,JM,K) = T(I,JM,K) -DTI/(DY(I,JM)+DY(I,JMM1)) * ((V(I,JM,K) + ABS(V(I,JM,K))) * (T(I,JM,K)-T(I,JMM1,K)) + (V(I,JM,K) - ABS(V(I,JM,K))) * (TBN(I,K)-T(I,JM,K)))
	SOUTH	UF(I,1,K) = T(I,1,K) -DTI/(DY(I,1)+DY(I,2)) * ((V(I,1,K) + ABS(V(I,1,K))) * (T(I,1,K)-T(I,2,K)) + (V(I,1,K) - ABS(V(I,1,K))) * (TBS(I,K)-T(I,1,K)))
Cyclic (B-4)		Much the same as (A - 6) except replace UAF with UF, etc. and T, S, Q2 and Q2L are handled similar to ELF.

³ This is a rough approximation to $c_e \Delta t / \Delta x$ and assumes that Δt has been set such that $(c_e)_{\max} \Delta t / \Delta x$ is near unity. A more sophisticated, but sometimes prone to noisy output, is Orlanski's scheme where c_e , or GAI itself, is determined by solving for the GAI at the next inboard location; e.g., $GAI = (UF(IM,J,K) - U(IM,J,K)) / (U(IMM1,J,K) - U(IM,J,K))$. GAI should be constrained according to $0 \leq GAI \leq 1$.

17. SUBROUTINE DENS

The UNESCO equation of state, as adapted by Mellor(1991) is used. The *in situ* density is determined as a function of salinity, potential temperature and pressure; the latter is approximated by the hydrostatic relation and constant density. Initially, the values TBIAS and SBIAS are subtracted from temperature and salinity to reduce round-off error. With 32 bit arithmetic, a suggestion is TBIAS = 10. and SBIAS = 35. for open ocean models; with 64 bits, zero values are appropriate. In DENS, these values are added again before the density is calculated. The actual density is normalized on 1000 kg/m². Since only gradients are needed (in subroutines BAROPG and PROFQ), the value 1.025 is subtracted to reduce round-off error. APPENDIX A includes some discussion of thermodynamics.

18. SUBROUTINE SLPMIN

This subroutine examines the topography and adjusts H(I,J) so that the difference of the depths of any two adjacent cells divided by the sum of the depths is less than or equal to the parameter, SLMIN. In the process, volume is preserved. What generally happens is that the topography in deeper water is not changed whereas the shallower regions are altered depending on resolution.

19. Utility subroutines

There are a number of utility subroutines supplied with the program. For the most part they can be understood by reference to comments written into the code. All of the printing subroutines except SUBROUTINE EPRXYZ print out numbers in integer format. They accept a scale factor in the argument list which is either zero, in which case the code generates its own scale factor, or a finite value which is then used to scale the printed numbers.

20. PROGRAM CURVIGRID

The program is set up to accept values of longitude and latitude, here denoted by x and y to define the four edges of the gridded domain. This can be altered to accommodate rectilinear coordinates by setting the cosine of the latitude, CS = 1, in subroutine ORTHOG or by expunging the variable completely.

The border of the domain is determined by NB, NR or NL points on the J=1, I=1 and J=JM borders respectively. In this version of the program, DATA statements contain this information. Cubic splines are then used to fill in the missing border coordinates.

The program is comprised of two steps:

I. The interior grid points ($1 < J < JM$) are filled such that the values at every I column is distributed proportionately to the y-values at I=1; the interior x value are similarly distributed.

II. Subroutine ORTHOG is called to render the $x_{i,j}$ and $y_{i,j}$ an orthogonal coordinate system. Then, use is made of the orthogonality conditions

$$\left(\frac{\partial x}{\partial s}\right)_j = -\left(\frac{\partial y}{\partial s}\right)_i, \quad \left(\frac{\partial y}{\partial s}\right)_j = \left(\frac{\partial x}{\partial s}\right)_i \quad (20-1a,b)$$

or

$$\frac{\delta_j x}{\delta_j s} = -\frac{\delta_i y}{\delta_i s}, \quad \frac{\delta_j y}{\delta_j s} = \frac{\delta_i x}{\delta_i s} \quad (20-2a,b)$$

With reference to Fig. 6, (20-2a,b) are solved according to

$$x_{i,j} - x_{i,j-1} = \frac{\delta_j s}{\delta_i s} [y_{i+1,j} - y_{i-1,j} + y_{i+1,j-1} - y_{i-1,j-1}] \quad (20-3a)$$

$$y_{i,j} - y_{i,j-1} = \frac{\delta_j s}{\delta_i s} [x_{i+1,j} - x_{i-1,j} + x_{i+1,j-1} - x_{i-1,j-1}] \quad (20-3b)$$

where

$$\delta_i s = \frac{1}{4} [(x_{i+1,j} - x_{i,j})^2 + (y_{i+1,j} - y_{i,j})^2]^{1/2} + \frac{1}{4} [(x_{i+1,j-1} - x_{i-1,j-1})^2 + (y_{i+1,j-1} - y_{i-1,j-1})^2]^{1/2} \quad (20-4a)$$

$$\delta_j s = [(x_{i,j} - x_{i,j-1})^2 + (y_{i,j} - y_{i,j-1})^2]^{1/2} \quad (20-4b)$$

The factor, CS, the cosine of the latitude, is not included in (20-3a,b) and (20-4a,b) but is included in the corresponding code in ORTHOG. Now, the above equations are iterated many times during which $\delta_j s$ is fixed; i.e., $\delta_j s$, $x_{i,1}$ and $y_{i,1}$ are data of the initial field specified in step I which are retained. In the course of iteration, $\delta_j s$, $x_{i,j}$ and $y_{i,j}$ are reevaluated. The shape of the original domain does change but not greatly. During this iteration, CS is held fixed. In fact, CS changes very little so that ORTHOG is called only twice to converge on this factor.

It should be noted that, if the border points contain too much curvature, then the curves normal to the $i = \text{constant}$ curves can focus to a point at some j row after which the calculation is nonsense. Some trial and error is therefore required. A way to avoid this is

to call POISSON after step I which solves for $y_{i,j}$ according to $\partial^2 y / \partial^2 i + \partial^2 y / \partial^2 j = 0$. This avoids the focusing problem but may not yield the most desirable grid.

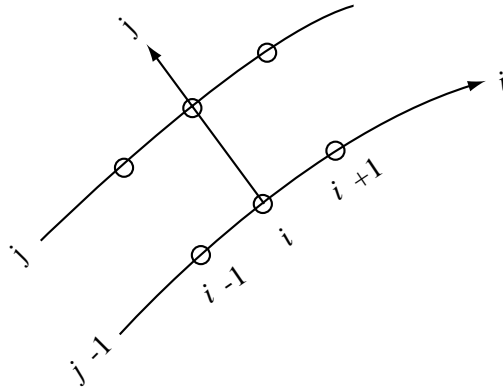


Figure 6. The orthogonal curvilinear grid system.

A good practice is to map the bottom topography on to the grid, then calculate the CFL limiting time step for each grid point; one wishes, of course, to avoid overly small steps.

APPENDIX A: Note on the Equation of State, Potential Temperature and Static Stability

Two equations of state for density are

$$\rho = \rho_1(T, S, p) \quad (\text{A-1})$$

$$\rho = \rho_2(\Theta, S, p) \quad (\text{A-2})$$

where T is *in situ* temperature and Θ is the potential temperature. In the model, (A-2) is used. To relate potential temperature, Θ , to *in situ* temperature, T , recall the thermodynamic relation for entropy.

$$Td\eta = dh - \frac{dp}{\rho} - \mu dS \quad (\text{A-3})$$

where η is the entropy, h , enthalpy and μ the chemical potential for salt taken here as a single average constituent. Furthermore,

$$dh = C_p dT + (1 - \alpha T) \frac{dp}{\rho} \quad (\text{A-4})$$

where we have set

$$\left(\frac{\partial h}{\partial T} \right)_{p,S} \equiv C_p, \quad \left(\frac{\partial h}{\partial p} \right)_{T,S} = \frac{(1 - \alpha T)}{\rho} \quad (\text{A-5a, b})$$

and where the coefficient of thermal expansion is

$$\alpha \equiv - \frac{1}{\rho} \left(\frac{\partial \rho}{\partial T} \right)_p \quad (\text{A-5c})$$

We note that (A-5b) has been obtained from (A-3) and one of Maxwell's relations. Combining (A-3) and (A-4), we have

$$d\eta = C_p \frac{dT}{T} - \alpha \frac{dp}{\rho} - \frac{\mu dS}{T} \quad (\text{A-6})$$

The definition of potential temperature in oceanography* is

$$C_{po} \frac{d\Theta}{\Theta} \equiv C_p \frac{dT}{T} - \alpha \frac{dp}{\rho} \quad (\text{A-7})$$

where $C_p = C_p(T, S, p)$ and $C_{po} = C_p(T, S, 0)$. Combining (A-6) and (A-7),

$$d\eta = C_{po} \frac{d\Theta}{\Theta} - \frac{\mu dS}{T} \quad (\text{A-8})$$

For processes where heat transfer, viscous dissipation and salt diffusion are null, $D\Theta/Dt = DS/Dt = 0$; then, from (A-8), $D\eta/Dt = 0$; i.e. the process is isentropic. An integral relation obtained from (A-7) is

$$T(z) - \Theta = \int_p^0 \frac{\alpha' T'}{C_p'} \frac{dp'}{\rho'} = - \int_z^0 \frac{\alpha' T' g}{C_p'} dz' \quad (\text{A-9})$$

The hydrostatic pressure relation is used to obtain the second expression on the right of the equal sign. In (A-9), $\Theta(z) = \Theta(0) = T(0)$. For $T = 10^\circ\text{C}$, $S = 35$ psu and $p = 0$, one finds

* as contrasted to meteorology, where, for a perfect gas, we have $\alpha T = 1$ and $p = \rho R T$. Potential temperature is then defined as $d\Theta/\Theta = d\eta/C_p = dT/T - (R/C_p) dp/p$ which can be integrated exactly to give $\Theta = T(p_0/p)^{R/C_p}$; p_0 is a reference pressure where $\Theta = T$.

(Gill, 1982, p.603) that $\alpha Tg / C_p \cong 0.12K / 1000m$. Equation (A-9) allows one to initialize potential temperature in the model given *in situ* properties. An algorithm to do this is provided by Bryden (1973).

Static Stability

To conveniently provide further background information and also inquire into an aspect of the Boussinesq approximation, we review the following equations for two-dimensional isentropic flow (see also Mellor and Ezer, 1995, for evaluation of a non-Boussinesq version of POM).

$$\frac{\partial \tilde{u}}{\partial x} + \frac{\partial \tilde{w}}{\partial z} = 0 \quad (\text{A-10})$$

$$\tilde{\rho} \left(\frac{\partial \tilde{u}}{\partial t} + \tilde{\mathbf{u}} \cdot \nabla \tilde{u} \right) = - \frac{\partial \tilde{p}}{\partial z} \quad (\text{A-11})$$

$$\tilde{\rho} \left(\frac{\partial \tilde{w}}{\partial t} + \tilde{\mathbf{u}} \cdot \nabla \tilde{w} \right) = - \frac{\partial \tilde{p}}{\partial z} - \tilde{\rho} g \quad (\text{A-12})$$

where we have made the Boussinesq approximation in (A-10) but have not done so in (A-11) and (A-12). Equation (A-10) is justified by examination of the full equation $\nabla \cdot \mathbf{u} + \tilde{\rho}^{-1} D\tilde{\rho} / Dt = 0$. The first term scales like u_0/L whereas the second scales as $(u_0/L) \delta\tilde{\rho} / \tilde{\rho}$. Since $\delta\tilde{\rho} / \tilde{\rho} \lesssim .05$ in the ocean, the second term can be neglected.

Let mean quantities be denoted by upper case letters and fluctuating quantities by lower case letters; the exception to this is density where ρ and ρ' are the mean and fluctuating values. For this analysis the mean velocity will be zero. Therefore we have $(\tilde{u}, \tilde{w}) = (u, w)$, $\tilde{p} = P + p$, $\tilde{\rho} = \rho + \rho'$, $\partial P / \partial z = -\rho g$ and $\rho = \rho(z)$ so that, for small perturbations,

$$\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} = 0 \quad (\text{A-13})$$

$$\rho \frac{\partial u}{\partial t} = - \frac{\partial p}{\partial z} \quad (\text{A-14})$$

$$\rho \frac{\partial u}{\partial t} = - \frac{\partial p}{\partial z} - \rho' g \quad (\text{A-15})$$

Now for isentropic flows the equations of state yields $D\tilde{\rho} / Dt = c^{-2}D\tilde{p} / Dt$ where $c^2 \equiv (\partial\tilde{p} / \partial\tilde{\rho})_{\theta,S}$ is the speed of sound squared. The corresponding density perturbation equation is

$$\frac{\partial\rho'}{\partial t} + w\frac{\partial\rho}{\partial z} = \frac{1}{c^2}\left(w\frac{\partial p}{\partial z} + \frac{\partial p}{\partial t}\right)$$

or

$$\frac{\partial\rho'}{\partial t} - w\frac{\rho N^2}{g} = \frac{1}{c^2}\frac{\partial p}{\partial t} \quad (\text{A-16})$$

where

$$\rho\frac{N^2}{g} \equiv -\frac{\partial\rho}{\partial z} + \frac{1}{c^2}\frac{\partial p}{\partial z} = -\frac{\partial\rho}{\partial z} - \frac{\rho g}{c^2} \quad (\text{A-17})$$

N^2 is the Brunt-Vassala frequency squared or the static stability. If one eliminates u , p and ρ' from (A-13) to (A-16), the resulting equation for w is

$$\frac{\partial^2}{\partial t^2}\left[\frac{\partial^2 w}{\partial z^2} + \frac{\partial^2 w}{\partial x^2} + \frac{N^2}{g}\frac{\partial w}{\partial z}\right] + N^2\frac{\partial^2 w}{\partial x^2} = 0 \quad (\text{A-18})$$

The last term in the square brackets can be neglected compared with the first. Thus, $g^{-1}N^2w_z/w_{zz} \simeq g^{-1}N^2L_z$ where L_z is the vertical scale height. $g^{-1}N^2L_z$ has two parts as shown in (A-17). If we take $L_z \approx 1000\text{m}$, then the first part, $|\rho^{-1}\rho_z L_z| \approx .010$ and the second part, $c^{-2}gL_z \approx .005$. Tracing back through the original equations, we find that this approximation is equivalent to setting $\rho = \text{constant} = \rho_o$ in (A-14) and (A-15) and neglecting the right side of (A-16).

A solution to (A-18) for $N^2 = \text{constant}$ is $w \propto \exp[i(lz + kx - \sigma t)]$ where the dispersion relation is $\sigma^2 = N^2k^2 / (l^2 + k^2)$. If $N^2 < 0$, the flow is unstable; if $N^2 > 0$, the flow is stable. Thus, N^2 , given by (A-17) is the correct static stability parameter for use in the turbulence closure model which are constructed from perturbation equations like (A-16) together with other equations and terms.

References

- Andre, J. C., G. DeMoor, G. Therry, and R. DuVachat, Modeling the 24-hour evolution of the mean and turbulence structures of the planetary boundary layer, J. Atmos. Sci., 35, 1861-1863, 1978.
- Asselin, R., Frequency filters for time integrations, Mon. Weather Rev., 100, 487-490, 1972.
- Baringer, M. O., and J. F. Price, Mixing and spreading of the Mediterranean outflow, J. Phys. Oceanogr., submitted, 1996.
- Blumberg, A. F., and G. L. Mellor, A coastal ocean numerical model, in Mathematical Modelling of Estuarine Physics, Proc. Int. Symp., Hamburg, Aug. 1978, edited by J. Sunderman and K.-P. Holtz, pp.203-214, Springer-Verlag, Berlin, 1980.
- Blumberg, A.F., and G.L. Mellor, Diagnostic and prognostic numerical circulation studies of the South Atlantic Bight, J. Geophys. Res., 88, 4579-4592, 1983.
- Blumberg, A.F., and G.L. Mellor, A description of a three-dimensional coastal ocean circulation model, in Three-Dimensional Coastal Ocean Models, Vol. 4, edited by N.Heaps, pp. 208, American Geophysical Union, Washington, D.C., 1987.
- Bryden, H. L., New polynomials for thermal expansion, adiabatic temperature gradient, and potential temperature of sea water, Deep-Sea Res., 20, 401-408, 1973.
- Galperin, B., L. H. Kantha, S. Hassid, and A. Rosati, A quasi-equilibrium turbulent energy model for geophysical flows, J. Atmos. Sci., 45, 55-62, 1988.
- Gill, A.E., Atmosphere-Ocean Dynamics, 662 pp., Academic Press, New York, 1982.
- Jerlov, N. G., Marine Optics, 14, 231 pp., Elsevier Sci. Pub. Co., Amsterdam, 1976.
- Jungclaus, H., and G. L. Mellor, A three-dimensional model study of the Mediterranean out flow, J. Mar. Systems, submitted, 1996.

- Klein, P., A simulation of the effects of air-sea transfer variability on the structure of the marine upper layers, J. Phys. Oceanogr., 10, 1824-1841, 1980.
- Knudsen, M., Hydrographical Tables. G.E.C. Gad, Copenhagen, pp., Williams and Norgate, London, 19010.
- Madala, R. V., and S. A. Piacsek, A semi-implicit numerical model for baroclinic oceans, J. Comput. Phys., 23, 167-178, 1977.
- Martin, P.J., Simulation of the mixed layer at OWS November and Papa with several models, J. Geophys. Res., 90, 903-916, 1985.
- Mellor, G.L., and T. Yamada, Development of a turbulence closure model for geophysical fluid problems, Rev. Geophys. Space Phys., 20, 851-875, 1982.
- Mellor, G. L., Retrospect on oceanic boundary layer modeling an second moment closure, Hawaiian Winter Workshop on "Parameterization of Small-Scale Processes", January 1989, University of Hawaii, Honolulu, Hawaii, 1989.
- Mellor, G.L., Analytic prediction of the properties of stratified planetary surface layers., J. Atmos. Sci., 30, 1061-1069, 1973.
- Mellor, G.L., and A.F. Blumberg, Modeling vertical and horizontal diffusivities with the sigma coordinate system, Mon. Wea. Rev., 113, 1380-1383, 1985.
- Mellor, G.L., and T. Yamada, A hierarchy of turbulence closure models for planetary boundary layers, J. Atmos. Sci., 31, 1791-1806, 1974.
- Mellor, G. L., L. H. Kantha, and H. J. Herring, On Gulf Stream frontal eddies. A numerical experiment, Ocean Modelling, 68, 7-11, 1986.
- Mellor, G.L., An equation of state for numerical models of oceans and esturaries. J. Atmos. Oceanic Tech. 8, 609-611, 1991.
- Mellor, G. L., T. Ezer and L. Y. Oey, The pressure gradient conundrum of sigma coordinate ocean models, J. Atmos. Oceanic. Technol., 11, 1126-1134, 1994.

- Mellor, G. L. and T. Ezer, Sea level variations induced by heating and cooling: An evaluation of the Boussinesq approximation in ocean model, J. Geophys. Res., 100(C10), 20,565-20,577, 1995.
- Mellor, G. L., and X. H. Wang, Pressure compensation and the bottom boundary layer, J. Phys. Oceanogr., in press, 1996.
- Oey, L.-Y., G.L. Mellor, and R.I. Hires, A three-dimensional simulation of the Hudson-Raritan estuary. Part I: Description of the model and model simulations, J. Phys. Oceanogr., 15, 1676-1692, 1985a.
- Oey, L.-Y., G.L. Mellor, and R.I. Hires, A three-dimensional simulation of the Hudson-Raritan estuary. Part II: Comparison with observation, J. Phys. Oceanogr., 15, 1693-1709, 1985b.
- Oey, L.-Y., G.L. Mellor, and R.I. Hires, A three-dimensional simulation of the Hudson-Raritan estuary. Part III: Salt flux analyses, J. Phys. Oceanogr., 15, 1711-1720, 1985c.
- Phillips, N. A., A coordinate system having some special advantages for numerical forecasting, J. Meteorol., 14, 184-185, 1957.
- Richtmyer, R. D., and K. W. Morton, Difference Methods for Initial-Value Problems, 2nd Ed., pp., Interscience, New York, 1967.
- UNESCO. Tenth rep. of the joint panel on oceanographic tables and standards. UNESCO Tech. Pap.in Marine Science No. 36. UNESCO, paris, 25pp., 1981
- Simons, T. J., Verification of numerical models of Lake Ontario. Part I, circulation in spring and early summer, J. Phys. Oceanogr., 4, 507-523, 1974.
- Zavatarelli, M., and G. L. Mellor, A numerical study of the Mediterranean Sea Circulation, J. Phys. Oceanogr., 25, 1384-1414, 1995.