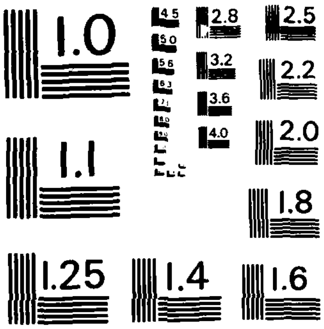
AD-A142 070 FIELD VERIFICATION PROGRAM COASTAL FLOODING AND STORM PROTECTION PROGRAM..(U) AERONAUTICAL RESEARCH ASSOCIATES OF PRINCETON INC NJ Y P SHENG APR 84 1/1

UNCLASSIFIED WES/IR/D-84-1 DACW39-80-C-0087 F/G 12/1 NL


1.0
4.5
5.0
5.6
6.3
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

(12)

US Army Corps of Engineers

FIELD VERIFICATION PROGRAM

COASTAL FLOODING AND
STORM PROTECTION PROGRAM

INSTRUCTION REPORT D-84-1

PRELIMINARY USER'S MANUAL 3-D MATHEMATICAL MODEL OF COASTAL, ESTUARINE, AND LAKE CURRENTS (CELC3D)

by

Y. Peter Sheng

Aeronautical Research Associates of Princeton, Inc.
50 Washington Road, P. O. Box 2229
Princeton, N. J. 08540

April 1984

Final Report

Prepared for Office, Chief of Engineers, U. S. Army
Washington, D. C. 20314

Under Contract No. DACW39-80-C-0087

Monitored by Hydraulics Laboratory
U. S. Army Engineer Waterways Experiment Station
P. O. Box 631, Vicksburg, Miss. 39180

ENVIRONMENTAL LABORATORY

Destroy this report when no longer needed. Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

The D-series of reports includes publications of the Environmental Effects of Dredging Programs:

Dredging Operations Technical Support

Long-Term Effects of Dredging Operations

Interagency Field Verification of Methodologies for Evaluating Dredged Material Disposal Alternatives (Field Verification Program)

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Instruction Report D-84-1	2. GOVT ACCESSION NO. AD-A142070	3. RECIPIENT'S CATALOG NUMBER
TLE (and Subtitle) PRELIMINARY USER'S MANUAL, 3-D MATHEMATICAL MODEL OF COASTAL, ESTUARINE, AND LAKE CURRENTS (CELC3D)		5. TYPE OF REPORT & PERIOD COVERED Final report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Y. Peter Sheng		8. CONTRACT OR GRANT NUMBER(s) Contract No. DACW39-80-C-0087
9. PERFORMING ORGANIZATION NAME AND ADDRESS Aeronautical Research Associates of Princeton, Inc. 50 Washington Road, P.O. Box 2229 Princeton, N. J. 08540		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Field Verification Program; Coastal Flooding and Storm Protection Program
11. CONTROLLING OFFICE NAME AND ADDRESS Office, Chief of Engineers, U. S. Army Washington, D. C. 20314		12. REPORT DATE April 1984 13. NUMBER OF PAGES 52
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) U. S. Army Engineer Waterways Experiment Station Hydraulics Laboratory P. O. Box 631, Vicksburg, Miss. 39180		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Available from National Technical Information Service, 5285 Port Royal Road, Springfield, Va. 22161.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

3-Dimensional design--Mathematical models. (WES)	Ocean currents. (LC)
CELC3D (Computer program). (LC)	
Computer programs--Mathematical models. (LC)	
Estuaries. (LC)	
Mathematical models. (LC)	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An efficient and comprehensive three-dimensional finite-difference model of Coastal, Estuarine, and Lake Currents (CELC3D) has been developed and is currently operative on a VAX-11/780 minicomputer. This user's manual discusses the special model features, general structure of the CELC3D computer code, subprogram summary, input/output information, data requirements, and example calculation.

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

PREFACE

The work described in this report was conducted by Aeronautical Research Associates of Princeton, Inc. (A.R.A.P.), Princeton, N. J., for the U. S. Army Engineer Waterways Experiment Station (WES), Vicksburg, Miss., under Contract No. DACW39-80-C-0087. A preceding report (Sheng 1983) described the development of a three-dimensional, finite-difference hydrodynamic model of Coastal, Estuarine, and Lake Currents (CELC3D).

Dr. Y. Peter Sheng, A.R.A.P., served as the principal investigator of this study and prepared this report. The contract was monitored by Mr. H. Lee Butler, WES. Through 30 June 1983, Mr. Butler was in the Wave Dynamics Division (WDD), HL, WES, under the general supervision of Mr. H. B. Simmons, Chief, HL, and Dr. R. W. Whalin and Mr. C. E. Chatham, Jr., former and acting Chiefs, WDD. The WDD and its personnel were transferred to the Coastal Engineering Research Center of the Waterways Experiment Station on 1 July 1983 under the supervision of Dr. R. W. Whalin.

Support for the research came from various sources: the U. S. Army Engineer District, Mobile; Coastal Flooding and Storm Protection Program, Mr. Jay Lockhart, OCE, Technical Monitor. Because of the need to predict the movement of dredged material, the WES also contributed financial support under the Field Verification Program (FVP), which it is conducting under sponsorship of the Office, Chief of Engineers (OCE), Washington, D. C. The WES FVP project manager was Dr. R. K. Peddicord, under the general supervision of Dr. R. M. Engler, Chief, Contaminant Mobility and Regulatory Criteria Group; Mr. D. L. Robey, Chief, Ecosystem Research and Simulation Division; and Dr. John Harrison, Chief, EL. FVP is conducted under the EL management unit entitled the Environmental Effects of Dredging Programs, Mr. C. C. Calhoun, Manager. Technical monitors of the FVP were Dr. J. Hall, Operations Division, OCE, and Dr. W. Klesch, Planning Division, OCE.

Commanders and Directors of WES during the contract performance were COL John L. Cannon, CE; COL Nelson P. Conover, CE; and COL Tilford C. Creel, CE. Technical Director was Mr. F. R. Brown.

This report should be cited as follows:

Sheng, Y. P. 1984. "Preliminary User's Manual, 3-D Mathematical Model of Coastal, Estuarine, and Lake Currents (CELC3D)," Instruction Report D-84-1, prepared by the Aeronautical Research Associates of Princeton, New Jersey, for the U. S. Army Engineer Waterways Experiment Station, CE, Vicksburg, Miss.

TABLE OF CONTENTS

LIST OF FIGURES

PRELIMINARY USER'S MANUAL
3-D MATHEMATICAL MODEL OF COASTAL, ESTUARINE, AND LAKE CURRENTS (CELC3D)

I. INTRODUCTION

This report documents the use of the CELC3D computer code for the computation of two- or three-dimensional unsteady currents in coastal, estuarine, or lake waters based on the free-surface model described in detail by Sheng (1983). The present version of the code is implemented on a VAX 11/780, a virtual machine. It therefore contains no buffering scheme and is programmed to run entirely in the central memory. Since it has no buffering scheme, it contains no machine dependent I/O and, except for minor program changes, should run on any other virtual machine, e.g., the CDC cyber 203, or other non-virtual machine with sufficient memory.

The CELC3D program solves the mean equations of fluid motion within a water body with respect to a right-handed Cartesian coordinate system located at the nominal water surface (see Figure 1.1). These equations consist of the partial differential equations for surface displacement (ζ), vertically integrated velocities, ~~(U,V)~~, three-dimensional velocities (u,v,w), temperature (T), salinity (S), density (ρ), and sediment concentration (C). The equations of motion are solved in dimensionless forms and are given in Appendix A in detail. The major variables and their corresponding FORTRAN names are given in Appendix B.

Sheng (1983) presented numerous example calculations of the CELC3D model, including: tidal currents in an open bight, wind-driven currents in a shallow lake, wind-driven currents in an open channel, formation and deepening of a thermocline, and tide- and wind-driven currents in the Mississippi coastal waters, etc. In this report, for simplicity, we will only briefly present the application to Mississippi coastal waters.

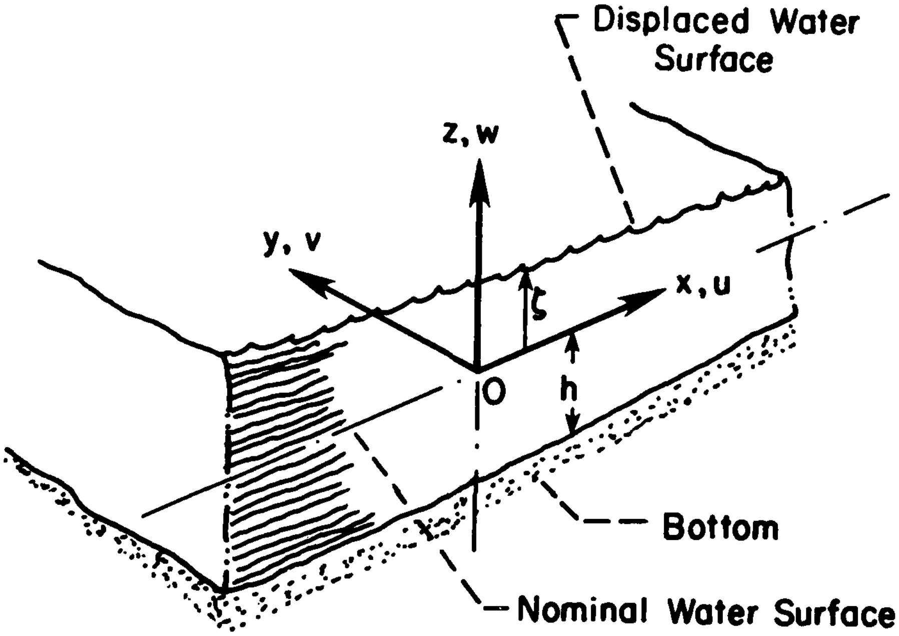


Figure 1.1. Cartesian coordinates at the nominal water surface

II. SPECIAL MODEL FEATURES

Details of the numerical model are given in a preceding report (Sheng 1983) and hence will not be completely repeated here. However, for ease of application and for completeness, several salient model features will be described.

Vertical Grid Resolution

A vertically stretched grid, the so-called σ-stretching, is used in the code to allow smooth representation of the bottom topography and, additionally, the same order of vertical resolution for the shallow and deeper parts of the water body. Basically, the vertical coordinate z is transformed into a new coordinate σ:

$$\sigma = \frac{z-\zeta(x,y,t)}{h(x,y) + \zeta(x,y,t)} \tag{2.1}$$

Using this relationship, the water column between $z=\zeta$ and $z=-h$ at any horizontal location is transformed into a layer between $\sigma=0$ and $\sigma=-1$ (Figure 2.1). This transformation is responsible for introducing the extra terms in the horizontal diffusion terms of the original equations. In the present version of the code only the surface slopes are assumed to be small while the surface displacements are not required to be small compared to the local water depths.

The transformed equations shown in Appendix A are derived by applying the chain rule to the original equations containing spatial derivatives in x, y, and z directions. Alternatively, one could derive a set of transformed equations by performing a tensor transformation.

Lateral Grid Resolution

A laterally stretched grid (Figure 2.2) is used in the code to allow fine resolution of the shoreline geometry and internal features (e.g., barrier islands) with coarser resolution in open waters far from the areas of interest. Specifically, the following piecewise reversible transformations

$$\sigma = \frac{z-\zeta}{\zeta+h} \approx \frac{z}{h}$$

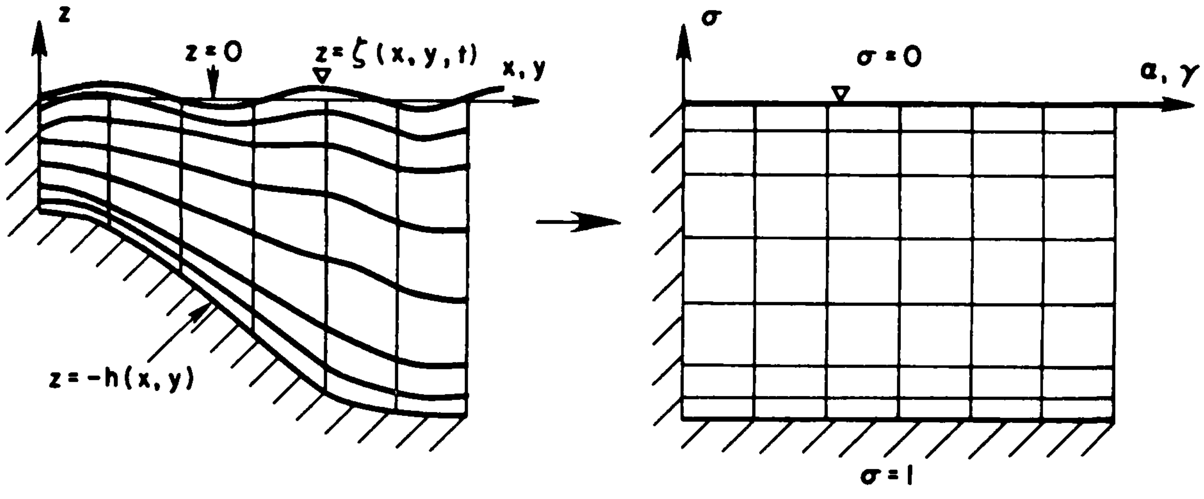


Figure 2.1. Vertical stretching of the coordinates

$$x = a_x + b_x\,\alpha^{c_x};\; y = a_y + b_y\,\gamma^{c_y}$$

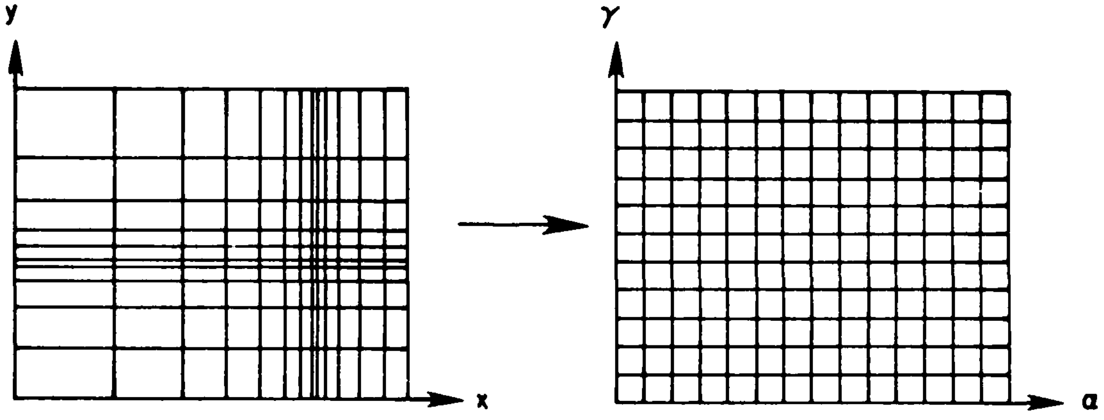


Figure 2.2. Lateral stretching of the coordinates

allow one to map a non-uniform horizontal grid (x,y) in the real space into a uniform horizontal grid (α,γ):

$$x = a_x + b_x\alpha^{c_x} \quad , \quad y = a_y + b_y\gamma^{c_y} \tag{2.2}$$

This transformation does not introduce extra terms to the equations of motion, although the stretching coefficients as defined by $\mu_x = d\alpha/dx$ and $\mu_y = d\gamma/dy$ now appear in all of the spatial derivative terms.

Grid Alignment

A staggered grid as shown in Figure 2.3 is used in the present code. The advantages of using such a staggered grid are twofold: (1) it simplifies the specification of boundary conditions, and (2) it reduces the computational effort.

Mode Splitting

As described in Sheng (1983) and in Appendix A, the present code computes the fast varying external variables (ζ,U,V) separately from the slowly varying internal variables (u', v', w, T, S, and ρ). The purpose of this mode-splitting procedure is to remove the stringent time step associated with the propagation of surface gravity waves which otherwise would limit the internal flow computations.

External Mode Algorithm

The external equations (A.1), (A.2), and (A.3) are computed with the following two sweeps:

$$\text{x-sweep:} \quad (I+\phi\lambda_x)W^* = [I+(I-\phi)\lambda_x+(I-2\phi)\lambda_y]W^n + \Delta t D^n \tag{2.3}$$

$$\text{y-sweep:} \quad (I+\phi\lambda_y)W^{n+1} = W^*+\phi\lambda_y W^n \tag{2.4}$$

where

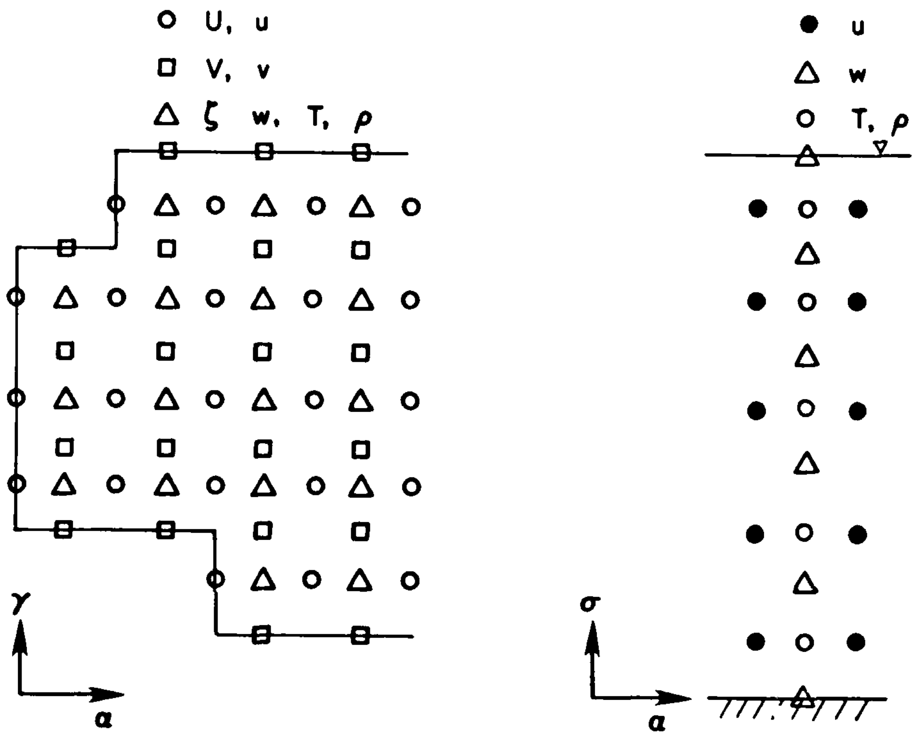


Figure 2.3. Staggered numerical grid

$$\lambda_x = \frac{A\Delta t}{\mu_x \Delta x}\,\delta_x; \qquad \lambda_y = \frac{B\Delta t}{\mu_y \Delta y}\,\delta_y \;\;;$$

$$A = \begin{pmatrix} o & \beta & o \\ H & o & o \\ o & o & o \end{pmatrix}; \quad B = \begin{pmatrix} o & o & \beta \\ o & o & o \\ H & o & o \end{pmatrix} \;; \quad D = \begin{pmatrix} o \\ D_x \\ D_y \end{pmatrix} \;; \quad W = \begin{pmatrix} \zeta \\ U \\ V \end{pmatrix} \qquad (2.5)$$

Where I is the identify matrix, Φ is a weighting factor usually taken to be 1, (Δx, Δy) are the horizontal grid spacings, and Δt is the time step.

This implicit scheme allows efficient computations of the external variables on two counts: (1) only ζ and U are computed in the x-sweep while only ζ and V are computed in the y-sweep, and (2) a time step several times larger than that for an explicit scheme, which is dictated by the propagation of surface gravity waves, can be used. The maximum time step is now governed by the advection speed in the computational domain according to the CFL (*Courant-Friedrichs-Lewy*) *condition* (*e.g.*, *Richtmyer and Morton 1967*):

$$\underset{x,y}{\mathrm{Min}} \left[\frac{\Delta x, \Delta y}{(g\overline{H})^{0.5}}\right] << \Delta t < \underset{x,y}{\mathrm{Min}} \left(\frac{U}{\overline{H}\Delta x} + \frac{V}{\overline{H}\Delta y}\right)^{-1} \qquad (2.6)$$

Internal Mode Algorithm

The perturbation horizontal velocities u' and v' are solved by:

$$\frac{H^{n+1}}{H^n}\, u'^{\,n+1} = u'^{\,n} + \Delta t \left(B_x - \frac{D_x}{H}\right) + \frac{\Delta t}{H^2}\frac{\partial}{\partial\sigma}\left[A_v \frac{\partial}{\partial\sigma}\left(H^{n+1} u'^{\,n+1}/H^n + U^{n+1}/H^n\right)\right] \qquad (2.7)$$

$$\frac{H^{n+1}}{H^n}\, v'^{\,n+1} = v'^{\,n} + \Delta t \left(B_y - \frac{D_y}{H}\right) + \frac{\Delta t}{H^2}\frac{\partial}{\partial\sigma}\left[A_v \frac{\partial}{\partial\sigma}\left(H^{n+1} v'^{\,n+1}/H^n + V^{n+1}/H^n\right)\right] \qquad (2.8)$$

The computations of these two equations are governed by a stability criterion similar to equation (2.6) with the vertically averaged velocities substituted by the local velocities u and v.

After u' and v' are computed, u and v can be obtained from:

$$u^{n+1} = u'^{n+1} + U^{n+1}/H^{n+1} \tag{2.9}$$

$$v^{n+1} = v'^{n+1} + V^{n+1}/H^{n+1} \tag{2.10}$$

Following these, ω^{n+1} and w^{n+1} can be computed from:

$$\omega^{n+1} = -\frac{\beta}{H^{n+1}} \sum_{K=1}^{K} \left(\frac{\Delta Hu}{\mu_x \Delta x} + \frac{\Delta Hv}{\mu_y \Delta y}\right)^{n+1} \Delta\sigma$$

$$+ \frac{\beta(1+\sigma)}{H^{n+1}} \sum_{K=1}^{KM} \left(\frac{\Delta Hu}{\mu_x \Delta x} + \frac{\Delta Hv}{\mu_y \Delta y}\right)^{n+1} \Delta\sigma \tag{2.11}$$

$$w^{n+1} = H^{n+1}\, \omega^{n+1} + \frac{1+\sigma}{\beta} \frac{\zeta^{n+1}-\zeta^{n}}{\Delta t} \tag{2.12}$$

where $\beta \equiv gD/L^2f^2$, and D and L are the reference length scales in vertical and horizontal directions, respectively.

After the velocities are computed, the code computes the temperature, salinity, and sediment concentration if the user so desires. These equations are solved by the same basic time-differencing and spatial-differencing schemes which will be briefly discussed in the following subsection.

General Numerical Consideration

In the present version of the code, a two-time-level scheme is used to complete the time integration from the n th time step to the n+1 th time step. This scheme is favored over a three-time-level scheme, the so-called leapfrog scheme, to avoid the "time-splitting" problem associated with the three-time-level scheme, and to reduce the storage requirement of the computer code on a small computer.

A second-order accurate advective scheme is used in the code. The advective terms are computed as the average of their conservative forms and non-conservative forms. Although higher order advection schemes (e.g., the fourth-order scheme) are supposed to be formally more accurate than the second-order scheme currently used in the code, they also require considerably more computational effort. Our study indicated that higher order schemes only give significantly better overall efficiency at a 1% error level or less.

Anticipating shallow water applications of the code, a special effort was made to remove the stringent time step limit associated with an explicit treatment of the vertical diffusion terms. The vertical implicit scheme implemented in the code is rather efficient since it involves the inversion of sparse matrices.

For a more detailed discussion on the general numerical consideration of the code, the reader is referred to Sheng (1983).

Turbulence Parameterization

The present code allows several choices of the vertical turbulent eddy coefficients. In non-stratified flow situations, either a constant eddy coefficient or a variable eddy coefficient of the following form can be used:

$$A_{v_0} = \frac{U_r \Lambda^2}{H} \left[\left(\frac{\partial u}{\partial \sigma} \right)^2 + \left(\frac{\partial v}{\partial \sigma} \right)^2 \right]^{0.5} \qquad (2.13)$$

where Λ is assumed to be a parabolic function of σ with its peak value at mid-depth not exceeding a certain fraction of the local depth. The precise form of Λ to be used in the code depends on the problem of interest.

Calibration is usually required.

In stratified flow situations, the effect of buoyancy on eddy coefficients can be parameterized in terms of several Richardson-number-dependent stability functions:

$$A_v = A_{vo}\ \Phi_1(Ri); \quad K_v = K_{vo}\ \Phi_2(Ri); \quad D_v = D_{vo}\ \Phi_3(Ri) \tag{2.14}$$

where

$$Ri = \frac{-gHh}{U_r^2(1+\rho)} \frac{\partial \rho}{\partial \sigma} \left[\left(\frac{\partial u}{\partial \sigma}\right)^2 + \left(\frac{\partial v}{\partial \sigma}\right)^2 \right]^{-1} \tag{2.15}$$

where A_{vo}, K_{vo}, and D_{vo} are eddy coefficients in the absence of any density stratification and Φ_1, Φ_2, and Φ_2 are stability functions traditionally assumed to be of the following forms:

$$\Phi_1 = (1+\sigma_1 Ri)^{m1}; \quad \Phi_2 = (1+\sigma_2 Ri)^{m2}; \quad \Phi_3 = (1+\sigma_3 Ri)^{m3} \tag{2.16}$$

As discussed in Sheng (1983), there exist great discrepancies among the many available empirical forms of the stability functions. To determine the coefficients in these stability functions, one needs sufficient data to achieve the "best fit" between modeled and measured results. As such, the specific formula/coefficients to be used in the code depend on the problem, the environment, and the available data. Munk and Anderson (1948) used the following formula in a study on the marine thermocline:

$$\Phi_1 = (1+10Ri)^{-1/2}; \qquad \Phi_2 = (1+3.33Ri)^{-3/2} \tag{2.17}$$

N[illegible]ly the form of the stability function may vary with the problem, but [illegible] Richardson numbers may also be used for different types of probl[illegible] example, the formation and deepening of the thermocline in a relatively shallow basin depends strongly on the relative importance of wind stress and heat flux at the free surface. In such a case, the following Richardson number could be used:

$$Ri = \frac{\kappa^2 gHh\sigma^2}{U_r^2 u_*^2} \frac{\partial\rho/\partial\sigma}{1+\rho} \qquad (2.18)$$

where κ is the von-Karman constant, u_* is the dimensionless friction velocity at the free surface, and the density gradient $\partial\rho/\partial\sigma$ is related to the surface heat flux.

Since a universally valid eddy viscosity formulation is not available, a modeler must judiciously choose a "best" formulation which hopefully can be validated with sufficient data. Seeking to remove some of the empiricism involved in the choice of stability functions, the present author has attempted another option to model the effect of stratification on turbulent mixing. The formulation is derived from a complete second-order closure turbulent transport model (Sheng 1982) by assuming the local equilibrium condition (Sheng 1983). A graphical comparison of this formulation versus some empirical formulae is shown in Figure 2.4. The author does not believe that any one of the empirical formulae is better than the others at all times. However, if a choice has to be made, it is probably safer to use the Munk and Anderson (1948) formulae given in (2.17).

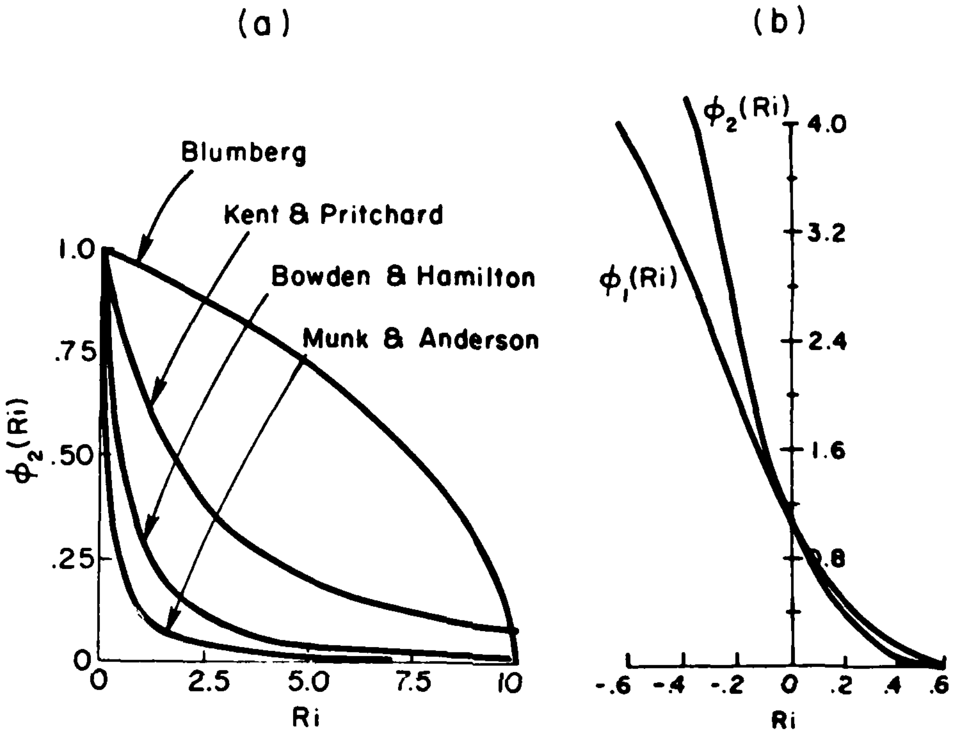


Figure 2.4. (a) Empirical stability functions of vertical turbulent eddy coefficients

(b) Stability functions determined from a second-order closure model of turbulent transport

III. GENERAL STRUCTURE OF THE CODE

The code is modular, and has nine major components: 1. initialization (CELCIN), 2. river routine (CELCRI), 3. density routine (CELCDE), 4. eddy coefficients computation (CELCED), 5. external (barotropic) mode computation (CELCBT), 6. internal (baroclinic) mode computation (CELCBC), 7. salinity routine (CELCSA), 8. temperature routine (CELCTE), and 9. output routine (CELCOT). They are summarized in the flow chart shown in Figure 3.1. The code can be run exclusively in the external mode with fairly large time step and several choices of bottom friction formulation. The internal mode can be updated as often as desired so long as the CFL condition based on the maximum advection speed is not violated.

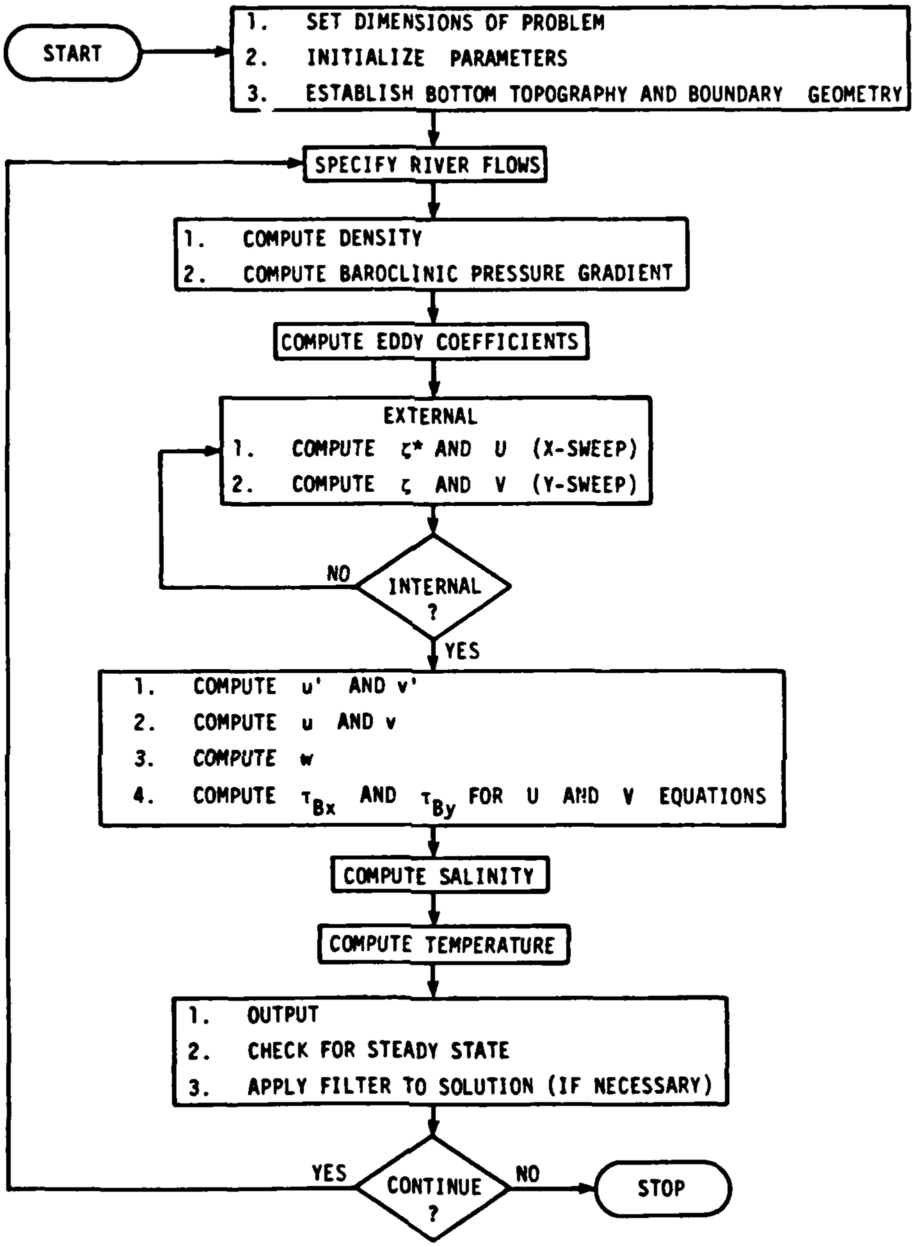

START
1. SET DIMENSIONS OF PROBLEM
2. INITIALIZE PARAMETERS
3. ESTABLISH BOTTOM TOPOGRAPHY AND BOUNDARY GEOMETRY
SPECIFY RIVER FLOWS
1. COMPUTE DENSITY
2. COMPUTE BAROCLINIC PRESSURE GRADIENT
COMPUTE EDDY COEFFICIENTS
EXTERNAL
1. COMPUTE ζ* AND U (X-SWEEP)
2. COMPUTE ζ AND V (Y-SWEEP)
NO
INTERNAL ?
YES
1. COMPUTE u' AND v'
2. COMPUTE u AND v
3. COMPUTE w
4. COMPUTE τBx AND τBy FOR U AND V EQUATIONS
COMPUTE SALINITY
COMPUTE TEMPERATURE
1. OUTPUT
2. CHECK FOR STEADY STATE
3. APPLY FILTER TO SOLUTION (IF NECESSARY)
YES
CONTINUE ?
NO
STOP

Figure 3.1. Flow chart of the three-dimensional hydrodynamic model

IV. SUBROUTINE SUMMARY

The following summarizes tasks performed by each of the major program elements:

CELCAI - This is a block data subroutine used to provide auxiliary input.

CELCBC - This subroutine updates the three-dimensional velocities based on the latest available external variables and three-dimensional velocities at the previous time steps. Using a vertical implicit scheme, the perturbation velocities u' and v' are first computed by matrix inversion via subroutine CELCEL. The latest available vertically averaged velocities are then added onto the perturbation velocities to obtain the horizontal velocities u and v. Since these velocities are aligned with the horizontal coordinates in the original grid (x,y,z), rather than those in the vertically stretched grid (x',y',σ), no transformation is required for these velocities. The last part of the subroutine computes the vertical velocities in the vertically stretched grid (ω) and in the original grid (w).

CELCBT - This subroutine computes the external variables (surface displacements and vertically integrated velocities). Surface displacements along the open boundaries are first computed. The subroutine then performs the x-sweep which updates the surface displacements (S) and the vertically integrated u-velocities (UI). The surface displacements along each row of the grid are computed by matrix inversion via subroutine CELCGS. During the y-sweep, surface displacements (S) and the vertically integrated v-velocities (VI) are updated. Again, subroutine CELCGS is accessed to compute surface displacements along each column of the grid.

CELCC1 - This subroutine computes the horizontal advective fluxes (FX, FY) in the conservation equations of various water quality parameters (temperature, salinity, and sediment) at all water points.

CELCC2 - This subroutine computes the vertical advective fluxes (FZ) in the conservation equations of various water quality parameters at all water points. After returning to the mainline program (CELCML), FZ is combined with FX and FY at each point and stored in FZ again.

CELCC3 - This subroutine computes the horizontal turbulent diffusive fluxes (FX, FY) for water quality parameters at all water points.

CELCC4 - This subroutine updates the water quality parameters (except temperature) by combining the explicitly computed advective fluxes and horizontal diffusive fluxes with the implicit vertical turbulent diffusive fluxes. At each horizontal location, the species concentration values in the water column are determined by matrix inversion via subroutine CELCGS.

CELCCN - This subroutine supervises the computation of water quality parameters (excluding temperature distribution). It accesses the subroutines CELCC1, CELCC2, CELCC3,and CELCC4.

CELCDE - This subroutine computes the water density at all water points based on formulae by Cox et al. (1967). It also computes the baroclinic pressure gradient terms for the horizontal momentum equations.

CELCDH - This is a user-modified subroutine which allows one to construct a desired bottom topography for the domain of computation.

CELCED - This subroutine computes the variable vertical turbulent eddy coefficients when ISPAC(9)=1 and KM≠1. It first computes the eddy coefficients at the temperature points. Eddy coefficients at the velocity points can be computed separately or by spatially averaging the ones at the temperature points.

CELCEL - This subroutine performs matrix inversion for the u' and v' velocities at any horizontal location.

CELCFI - This subroutine invokes the spatial smoother (CELCSM) for the three-dimensional variables, every ISPAC(2)th step for the salinity or every ISPAC(8)th step for the velocity.

CELCF2 - This subroutine invokes the spatial smoother (CELCSM) for the vertically integrated variables.

CELCGS - This subroutine performs matrix inversion for the surface displacements along each row/column during the x/y sweep.

CELCIN - This is the initializer subroutine. It reads the input variables from unit 4 file and combines them with those provided through the auxiliary input subroutine CELCAI. It then checks the starting flags on files unit 5 and unit 16, opens up time file for storing transient data if IST(1)≠0, and then establishes bottom topography and shoreline geometry. The initial values of the variables are either read in from file unit IR and/or ICONC for a restart run, or prescribed for a new start run. Major input variables are written onto a disc file, unit 6, for printer output.

CELCML - This is the mainline program which supervises the overall operation of the complete code CELC3D. It starts with a parameter statement which defines the number of grid points (IM, JM, KM) in spatial (x', y', σ) directions. If KM is set to 1, only the vertically integrated variables are computed and many of the subroutines such as CELCED and CELCBC are not accessed. The maximum allowable number of grid points is only limited by the available computer storage. The program starts with the initializer CELCIN and the river subroutine CELCRI, which are followed by computations of density in CELCDE and eddy coefficients in CELCED. It then proceeds to compute the barotropic variables in CELCBT and the baroclinic variables in CELCBC, CELCSA, and CELCTE. Sediment or other species can be computed via CELCCN. The program ends with output to printer and/or disc files in CELCOT.

CELCOP - This is a user-modified subroutine which defines the surface displacements along the open boundaries.

CELCOT - This is the output subroutine. It first checks to see if residual currents are to be computed (ISW≠0). If so, the residual currents are written to disc file unit IWS at the end of the run when the time counter IT has reached IT2. The subroutine then computes the maximum Courant number (based on the advection speed) in the computational domain and prints it out when exceeding 0.9. It then performs a dynamic time stepping algorithm by first checking the maximum rate of change of major variables in the computational domain. If the weighted rate of change is smaller than EPSILON, the time step is allowed to increase by 20% of the previous time step so long as it is less than DELTMAX. Otherwise, the time step is cut back proportional to the ratio between EPSILON and the actual rate of change so long as it is greater than DELTMIN. At every IP1th time step, subroutine CELCSS is then accessed to see if the velocity field has reached a steady state in time for the run to be stopped. Output is written to the time file unit 8 if IST(1)>0. Every IP2th time step, number output or simplified contour plot of the major variables are written to the print file unit 6. The rest of the subroutine checks for conservation and steady-state, and applies spatial smoothing to the water quality parameters. At every ISPAC(1)th time step, if IWR=1 , major output are written to disc file unit IW and/or ICONC.

CELCRI - This subroutine computes the horizontal velocities at river inflows/outflows based on specified volumetric flow rates in ft^3/sec. Twenty river points are allowed in the horizontal directions.

CELCSA - This subroutine supervises the computation of salinity. Subroutines CELCC1, CELCC2, and CELCC3 are accessed to compute the advective fluxes and the horizontal diffusive fluxes. If ISPAC(9)=0, subroutine CELCC4 is called to compute the new salinity values. If ISPAC(9)≠0, subroutine CELCT4 is called instead.

CELCSM - This subroutine performs the spatial smoothing as described in Sheng et al. (1978). The purpose of this smoothing is to filter out the $2\Delta x$ wavelength point-to-point oscillations which, if not properly controlled, may lead to instability of the solution. The current version of the smoother is applicable to a non-uniform grid.

CELCSS - This subroutine checks for steady-state of the dependent variables.

CELCT4 - If ISPAC(9)≠0, this subroutine computes the temperature and the salinity distribution based on explicitly computed advective fluxes and horizontal diffusive fluxes, but implicit vertical diffusive fluxes.

CELCTE - This subroutine supervises the computation of temperature distribution. It accesses the subroutines CELCC1, CELCC2, CELCC3, and CELCT4.

CELCW3 - This is a utility subroutine for printing out three-dimensional variables.

CELCWR - This is a utility subroutine for printing out two-dimensional variables.

V. DISC FILES

Unit 4 - This is the main input file providing the essential input information via formatted card images which are described in detail in Section VI.

Unit 5 - This is a single-card file which specifies the start up flag ISTART with I4 format. ISTART=0 signifies a new run and starts with the initial flow data prescribed in CELCIN. ISTART=1 signifies a restart run and the code picks up the initial flow data from disc file unit 9.

Unit 6 - This is the file containing the major printouts.

Unit 8 - This is an unformatted sequential file which stores the surface displacements, vertically integrated velocities and three-dimensional velocities at selected stations.

Unit 9 - This is an unformatted sequential file which stores the major flow output data at desired intervals. It also contains the necessary information for restarting the run. The variables stored in this file are listed in Table 5.1.

Unit 11 - This file contains the variable bottom topography provided by the user. It is an unformatted sequential file containing HU, HV, and HS each dimensioned as (JM, IM).

Unit 12 - This is an unformatted sequential file containing the grid parameters NS, MS, JU1, JU2, JV1, JV2, IU1, IU2, IV1, and IV2. NS and MS are both dimensioned as (JM, IM) and specify the alignments of each grid point with respect to the x and y coordinates, respectively. JU1 and JU2, both dimensioned as (IM), specify the grid indices of the first and the last water points along each grid column I. JV1 and JV2 are defined at the v-velocity points as the counterparts of JU1 and JU2. IU1 and IU2, both dimensioned as (JM), specify the grid indices of the first and the last water points along each grid row J. IV1 and IV2 are defined at the v-velocity points as the counterparts of IU1 and IU2.

Unit 13 - This is an unformatted sequential file which stores the major sediment or species output data at desired intervals. The variables stored in this file are described in Table 5.2.

Unit 14 - This sequential file contains the necessary information for generating the surface displacements along the open boundaries. The variables are defined in Table 5.3.

Unit 15 - This is the output file for storing residual flow data and contains the same variable groups as defined in Table 5.1.

Table 5.1 Output Variables on Unit 9 File

Group #1: TIME, IT, FNAME(3), FNAME(4), IM, JM, KM, XREF, ZREF, UREF, COR, AV

Variables FNAME(3), FNAME(4), XREF, ZREF, UREF, COR, and AV are explained in Section VI on the main input cards. Variables IM, JM, and KM are explained in the main program (page 20).

Variable	Definition
TIME	Real time in hours
IT	Time index

Group #2: XS, XU, YS, YV, HU, HV, HS, FMU, FMV, FMS, FMSV

Variable	Dimension	Definition
XS	IM	x positions of the ζ points
XU	IM	x positions of the u points
YS	JM	y positions of the ζ points
YV	JM	y positions of the v points
HU	JM, IM	water depths at the u points
HV	JM, IM	water depths at the v points
HS	JM, IM	water depths at the ζ points
FMU	IM	x-stretching coefficients at the u points
FMV	JM	y-stretching coefficients at the v points
FMS	IM	x-stretching coefficients at the ζ points
FMSV	JM	y-stretching coefficients at the ζ points

Group #3: U, V, W, WUW

Variable	Dimension	Definition
U	KM, JM, IM	u-velocities
V	KM, JM, IM	v-velocities
W	KM, JM, IM	ω-velocities
WUW	KM, JM, IM	w-velocities at the u points

Table 5.1 (Concluded)

Group #4: UI, VI

Variable	Dimension	Definition
UI	JM, IM	Vertically integrated u-velocities
VI	JM, IM	Vertically integrated v-velocities

Group #5: S

This group stores the surface displacements, S (JM, IM).

Group #6: T

This group stores the temperature distribution T (KM, JM, IM).

Group #7: SA

This group stores the salinity distribution SA (KM, JM, IM).

Group #8: GA, GB

Variable	Dimension	Definition
GA	KM, JM, IM	Vertical eddy viscosity
GB	KM, JM, IM	Vertical eddy diffusivity

Group #9: TBX, TBY

Variable	Dimension	Definition
TBX	JM, IM	Bottom shear stress in x-direction
TBY	JM, IM	Bottom shear stress in y-direction

Table 5.2 Output Variables on Unit 13 File

Group #1: IT, TIME, CC, DERO, DEPO, DNET

Variable	Dimension	Definition
CC	KM, JM, IM	Sediment/species concentration
DERO	JM, IM	Amount of sediment entrained during the present step.
DEPO	JM, IM	Amount of sediment deposited during the present step.
DNET	JM, IM	Net thickness of deposited sediment since the beginning of the run.

Group #2: TIME, IT, FNAME(5), FNAME(6), IM, JM, KM, XREF, ZREF, UREF, COR, AV

These variables are defined in Section VI on the main input cards.

Group #3: XS, XU, YS, YV, HU, HV, HS

These variables are defined in Table 5.1.

Table 5.3 Variables Stored on Unit 14 File

Group #1: IOBST(J), J=1, JMAX

This defines the x-grid indices of the open boundary points.

Group #2: JOBST(J), J=1, JMAX

This defines the y-grid indices of the open boundary points.

Group #3: KSTAT, TLON, TM, HO

Variable	Dimension	Definition
KSTAT	JMAX	Station number
TLON	JMAX	Longitude of station
TM	JMAX	Dummy variable
HO	JMAX	Dummy variable

Group #4: HM

This defines the amplitudes of major tidal constituents at the open boundary stations. HM is dimensioned (NCONT, JMAX) where NCONT is the total number of constituents considered.

Group #4: FKAPPA

This defines the phase angles of major tidal constituents at the open boundary station. FKAPPA is dimensioned (NCONT, JMAX).

VI. MAIN INPUT CARDS

#1: IOPEN, ISALT, IBTM, ITIDE, ITEST, ISMALL, JWIND (7I4)

IOPEN - Open boundary flag.
= 0 No open boundary.
= 1 Western boundary is open.
= 2 Southern boundary is open.
= 3 Eastern boundary is open.
= 4 Northern boundary is open.
= 8 Northern and eastern boundaries are open.

ISALT - Salinity flag.
= 0 No salinity.
= 1 With salinity.

IBTM - Bottom flag.
= 0 Depth changes linearly from H1 along western boundary to H2 along eastern boundary.
= 1 Depth changes linearly from H1 along southern boundary to H2 along northern boundary.
= 2 Variable bottom topography is read from file unit 11.

ITIDE - Tide flag.
= 0 No tide.
= 1 With tide.

ITEST - Test flag.
= 0 Operational run with minimal output
= 1 Test run with extra output.

ISMALL - Small amplitude flag.
= 0 Invokes small amplitude assumption.
= 1 Does not invoke small amplitude assumption.

JWIND = 0 Zero surface displacement along open boundary. No tide.
= 2 Zero surface slope along open boundary. No tide.

#2: IAV, ITEMP, IRAD, ISTEP, IEXP (5I4)

IAV - Eddy viscosity flag.
= 0 Constant vertical eddy viscosity specified by AV.
= 1 Constant vertical eddy viscosity determined as a linear function of wind stress from AVA + τ_W x AVB.

ITEMP - Temperature flag.
= 0 Thermally homogeneous.
= 1 Thermally stratified.

IRAD - Radiation flag.
= 0 No solar radiation.
= 1 With solar radiation.

ISTEP - Stepping flag.
= 0 Fixed time step.
= 1 Dynamic stepping, i.e., variable time step is determined by the CFL condition at each instant of time.

IEXP - Vertical eddy coefficient flag.
= 0 Munk-Anderson-type eddy coefficients.
= 1 If both ITEMP and ISALT are zero, simple variable eddy coefficients. If either ITEMP or ISALT is non-zero, Munk-Anderson-type eddy coefficients.
= 2 Sundaram-type eddy coefficients.
= -2 Superequilibrium turbulence.

#3: IFI, IFA, IFB, IFC, IFD, IWS, IWC, ICI (8I4)

IFI - Nonlinear inertia flag.
= 0 No nonlinear inertia terms.
= 2 With nonlinear inertia terms in weighted conservative and non-conservative forms.

IFA - Flag for the first higher order term in lateral turbulence.
= 0 No first higher order term in lateral turbulent diffusion.
= 1 With first higher order term in lateral turbulent diffusion.

IFB - Flag for the second higher order term in lateral turbulence.
= 0 No second higher order term in lateral turbulent diffusion.
= 1 With second higher order term in lateral turbulent diffusion.

IFC - Flag for the third higher order term in lateral turbulence.
= 0 No third higher order term in lateral turbulent diffusion.
= 1 With third higher order term in lateral turbulent diffusion.

IFD - Flag for the leading term in lateral turbulence.
= 0 No leading term in lateral turbulent diffusion.
= 1 With leading term in lateral turbulent diffusion.

IWS - Disc file unit number for storing residual currents.

IWC - Concentration output flag.
= 0 Does not write concentration output on disc file.
= 1 Writes concentration output on disc file.

ICI - Concentration input flag.
= 0 Does not read concentration from input disc file unit ICONC.
= 1 Reads concentration from input disc file unit ICONC.

#4: ITC, IZETA, IREAD, ICONC, IPU, IPW, IPWUW, IVLCY, ICC (9I4)

ITC - Bottom stress law flag.
= 1 Bottom stress computed from linear relationship with eddy coefficient formulation.
= 2 Quadratic bottom stress law.

IZETA - Dummy variable.

IREAD - If IVLCY=0 and ICC≠0, IREAD is the time interval at which velocity data are read from input disc file unit IR.

ICONC - Unit number of the output disc file for storing concentration variables.

IPU - Print flag.
= 0 Does not print out horizontal velocities.
= 1 Prints out horizontal velocities.

IPW - Print flag.
= 0 Does not print out vertical velocities $(^{\omega})$.
= 1 Prints out vertical velocities $(^{\omega})$.

IPWUW - Print flag.
= 0 Does not print out vertical velocities (w).
= 1 Prints out vertical velocities (w).

IVLCY - Velocity flag.
= 0 Does not compute velocities.
= 1 Computes velocities.

ICC - Concentration flag.
= 0 No concentration computation.
= 1 Computes dissolved species concentration.
= 2 Computes suspended sediment concentration.

#5: IT1, IT2, IP1, IP2, IP3, IR, IW, IWR, ITS, ITB (10I4)

IT1 - Initial time step.

IT2 - Final time step.

IP1 - Time step interval for brief printout.

IP2 - Time step interval for total printout.

IP3 - Time step interval for printout within each internal time step.

IR - Unit number of input disc file for flow variables.

IW - Unit number of output disc file for flow variables.

IWR - Output flag.
= 0 Does not write output on disc file unit IW.
= 1 Writes output on disc file unit IW every IP2TH step.

ITS - The ratio between the internal time step and the external time step.

ITB - Bottom stress law flag.
= 1 Linear bottom stress law.
> 3 Quadratic bottom stress law, implicit scheme.

#6: IGI, IGT, IGS, IGU, IGW, IGC, IGH, (7I4)

IGI = 0 If IGH=1, numerical printout for topography.
= 1 If IGH=0, simple contour printout for topography.

IGT = 0 Numerical printout for temperature and density, etc.
= 1 Simple contour printout for temperature and density, etc.

IGS = 0 Numerical printout for surface displacement.
= 1 Simple contour printout for surface displacement.

IGU = 0 Numerical printout for horizontal velocities.
= 1 Simple contour printout for horizontal velocities.

IGW = 0 Numerical printout for vertical velocities.
= 1 Simple contour printout for vertical velocities.

IGC = 0 Numerical printout for concentration and related variables.
= 1 Simple contour printout for concentration and related variables.

IGH = 0 No printout for bottom depths, slopes, and curvatures.
= 1 With printout for bottom depths, slopes, and curvatures.

#7: IPA, IPB, ID, JPA, JPB, JD, KPA, KPB, KD (9I4)

These are indices controlling the range of printouts in x (IPA, IPB, ID), y (JPA, JPB, JD), and z (KPA, KPB, KD) directions. For example, printouts in the x direction start from I=IPA to I=IPB at every IDTH interval.

#8: IBL, IBR, JBM, JBP (4I4)

Concentration computation does not have to be performed through the entire grid. One can start the initial concentration computation from I=IBL to I=IBR and J=JBM to J=JBP.

#9: IC1, IC2, JC1, JC2, ID1, ID2, JD1, JD2 (8I4)

If ICC≠0, user can specify two regions (first region covers I=IC1 to I=IC2 and J=JC1 to J=JC2) within which the initial concentration is C0.

#10: ISPAC (10I4)

ISPAC(1) - Disc output flag.
= 0 No output to disc files.
≠ 0 Every ISPAC(1) time step, output data are dumped onto disc files.

ISPAC(2) - Smoothing flag.
= 0 No smoothing is applied.
≠ 0 Every ISPAC(2) time step, apply spatial smoothing to salinity results.

ISPAC(3) - Eddy coefficient flag.

ISPAC(4) - External mode flag.
= 2 Implicit scheme.

ISPAC(5) - External bottom friction flag in CELCBC.
= 2 Implicit bottom friction.

ISPAC(6) - Two-dimensional flag.
= 0 Does not compute y-component of the variables.
= 1 Computes y-component of variables.

ISPAC(7) - Basin geometry flag.
= 0 Simple rectangular basin. Grid indices NS, MS, JU1, JU2, JV1, JV2, IU1, IU2, IV1, IV2 defined by the input subprogram.
= 1 Complex basin. Grid indices must be read from input disc file unit 12.

ISPAC(8) - Smoother flag.
= 0 No smoothing is applied to velocities.
≠ 0 Every ISPAC(8) step, apply spatial smoothing to velocities.

ISPAC(9) - Vertical eddy coefficient flag.
= 0 Constant vertical eddy coefficients.
= 1 Variable vertical eddy coefficients.

ISPAC(10)- Residual current flag.
= 0 Does not compute residual currents.
= 1 Computes residual currents.

#11: JSPAC (10I4)

JSPAC(1) - Vertical grid flag.
= 0 Uniform vertical grid spacing.

JSPAC(2) - Bottom stress coefficient flag.
= 0 Constant bottom stress coefficient.
= 1 Variable bottom stress coefficient.

JSPAC(3) - Coriolis flag.
= -1 No Coriolis terms.
= 0 Coriolis terms evaluated.

JSPAC(4) - Correction terms flag.
= 0 No correction terms for the implicit external algorithm.
= 1 With correction terms for the implicit external algorithm.

JSPAC(5) - External bottom friction flag in CELCBT.
= 2 Implicit bottom friction in external equations.
= 4 Explicit bottom friction in external equations.

JSPAC(6) - Variable bottom topography input flag.
= 0 If IBTM>1, read bottom topography from input disc file unit 11.
≠ 0 If IBTM>1, define bottom topography from subprogram CELCOP.

JSPAC(7) = 0 Use the latest term of the vertically integrated velocity in the bottom friction term of the internal equations.

JSPAC(8) - External bottom friction flag.
= 0 If KM=1, explicit bottom friction.
= 1 If KM=1, implicit bottom friction.

JSPAC(9) - Time-differencing flag.
= 0 2 time-level scheme.

JSPAC(10)- Dummy index.

#12: ICON, IVER (2I4)

ICON - Advection flag for computation of water quality parameters (dissolved species, temperature, salinity, and sediment).
= 1 Upwind advective scheme.
= 2 Combined upwind and central differencing scheme.
= 3 Slightly different upwind advective scheme.

IVER - Vertical diffusion flag.
= 1 Explicit vertical diffusion terms for water quality parameter computation.
= 2 Implicit vertical diffusion terms.

#13: NRIVER (I4)

Number of rivers in the computational domain.

#14: IRIVER (NRIVER*I4)

x-grid indices for river points. If IRIVER(1)=0, no rivers.

#15: JRIVER (NRIVER*I4)

y-grid indices for river points.

#16: LRIVER (NRIVER*I4)

LRIVER - River alignment flag.
= 1 River flows in the x-direction.
= 2 River flows in the y-direction.

#17: URIVER (NRIVER*F8.0)

Volumetric flow rates in ft^3/sec for rivers with LRIVER=1.

#18: VRIVER (NRIVER*F8.0)

Volumetric flow rates in ft^3/sec for rivers with LRIVER=2.

#19: FNAME (6A4)

A six-element vector specifying the input disc file for flow variables, the output file for flow variables, and the disc file for concentration variables.

#20: RSPAC (10F8.0)

RSPAC(1) - Manning's n (in c.g.s. unit)

RSPAC(2) - Bottom friction coefficient for comparison with analytical results.

RSPAC(3) - An infinitesimal number (~0.001) used in checking steady-state.

RSPAC(4) - An infinitesimal number (~0.001) used in checking steady-state.

RSPAC(5) - Not used; set to zero.
RSPAC(6) - Not used; set to zero.

RSPAC(7) - Open boundary condition flag for velocities.
= 0 Along an open boundary, sets normal gradient of normal velocity to zero.
= 1 Along an open boundary, sets normal curvature of normal velocity to zero.
= 4 Along an open boundary, sets normal gradients of normal and tangential velocities to zero.

RSPAC(8) - Not used; set to zero.

RSPAC(9) - Coefficient for the spatial smoother.

RSPAC(10)- Coefficient for the curvature check of spatial smoother.

#21: XREF, ZREF, UREF, COR (4F8.0)

XREF - Reference length in x-direction.
ZREF - Reference depth.
UREF - Reference velocity.
COR - Coriolis acceleration.

#22: GR, AV,TAUX, TAUY (4F8.0)

GR - Gravitational acceleration.
AV - Vertical eddy viscosity.
TAUX - Wind stress in the x-direction.
TAUY - Wind stress in the y-direction.

#23: H1, H2, RO, CBS (4F8.0)

H1 - Water depth along one boundary.
H2 - Water depth along the opposite boundary.
RO - Reference water density.
CBS - Constant bottom friction coefficient.

#24: AH, SMAX, AVA AVB, AVM (5F8.0)

AH - Lateral turbulent eddy viscosity.
SMAX - Maximum allowable surface displacement. The run terminates if the surface displacement exceeds this value.
AVA - Background vertical eddy viscosity when wind stress is zero.
AVB - If IAV=1, vertical eddy viscosity is AVA+ τ_w *AVB.
AVM - Minimum allowable vertical eddy coefficient.

#25: DELT, ZRUF (2F8.0)

DELT - Time step.
ZRUF - Reference height above the bottom.

#26: DELTMIN, DELTMAX, EPSILON (3F8.0)

DELTMIN - Minimum allowable time step.
DELTMAX - Maximum allowable time step.
EPSILON - Maximum allowable rate of change of major variables.

#27: WTS, WTU, WTV (3F8.0)

Weighting factors for computing relative changes in surface displacement, u-velocity, and v-velocity in determining a new time step.

#28: TSURF, SSURF, XMAP, WSET (4F8.0)

TSURF - Surface temperature.
SSURF - Surface salinity.
XMAP - Mapping ratio of horizontal grid.
WSET - Settling velocity.

#29: TBASE, TWE, TWH, FKB (4F8.0)

TBASE - Background temperature.
TWE - Temperature in epilimnion.
TWH - Temperature in hypolimnion.
FKB - Vertical grid index of the initial thermocline location.

#30: BVR, S1, S2, TMAX, PR, PRV (6F8.0)

BVR - Reference vertical eddy diffusivity.
S1, S2 - Empirical constants used in variable vertical eddy coefficients.
TMAX - Maximum allowable temperature.
PR - Turbulent Prandtl number.

PRV	-	Vertical turbulent Prandtl number.

#31: SAB (KM*F8.0)

Initial vertical profile of salinity. Used only for simple test runs. For realistic simulations, initial salinity over the entire grid may be specified.

#32: BZ1, BZ2 (2F8.0)

Bottom roughness heights at ζ points and u points, respectively.
Of course, one may specify the complete array of BZ0(JM,IM) and BZ0U(JM,IM).

#33: C0, Q0, SSS0 (3F8.0)

C0	-	Initial concentration value for test runs. One may specify the initial concentration over the entire grid instead.
Q0	-	Initial surface heat flux.
SSS0	-	Initial surface displacement.

#34: RIC, FZS, GAMAX, GBMAX (4F8.0)

RIC	-	Critical Richardson number
FZS	-	Multiplication factor in scale length formula.
GAMAX	-	Maximum allowable vertical eddy viscosity.
GBMAX	-	Maximum allowable vertical eddy diffusivity.

#35: IYEAR, IMONTH, IDATE, IHR (4I4)

Year, month, date,and hour at the initiation of a tidal computation.

#36: CREF, F, HT, CMAX (4F8.0)

CREF	-	Reference concentration.
F	-	Surface flux of species.
HT	-	Initial water content of bottom sediment.
CMAX	-	Maximum allowable concentration.

#37: BT, TCR, CK, TBDRY (4F8.0)

BT	-	Deposition velocity.
TCR	-	Critical shear stress.
CK	-	Coefficient of entrainment relationship (see Sheng 1983).
TBDRY	-	Coefficient of entrainment relationship.

#38: CK2, TCR2, TBDRY2, CK3, TCR3 (5F8.0)

CK2	-	Coefficient of entrainment relationship.
TCR2	-	Coefficient of entrainment relationship.
TBDRY2	-	Coefficient of entrainment relationship.
CK3	-	Coefficient of entrainment relationship.
TCR3	-	Coefficient of entrainment relationship.

VII. AUXILIARY INPUT INFORMATION

A blockdata subroutine CELCAI provides the necessary auxiliary information required for the code. The subroutine defines several groups of input data through DATA statements.

Group #1: IST, JST, KST

IST - Dimensioned 40, it defines the I-grid index of stations where continuous flow output are to be stored on time file (unit 8).

JST - Dimensioned 40, it defines the J-grid index of stations where continuous flow output are to be stored on time file (unit 8).

KST - Dimensioned 40, it defines the K-grid index of stations where continuous flow output are to be stored on time file (unit 8).

Group #2: NXEND, ALREF, IN, XXX, XAL, AMU, XA, XB, XC

NXEND - This defines the total number of x-grid regions.

ALREF - Reference length in the stretched x-grid (α).

IN - Dimensioned NXEND, it defines the I-grid index of the first grid point in each x-grid region.

XXX - Dimensioned NXEND, it defines the x-coordinate of the first grid point in each x-grid region.

XAL - Dimensioned NXEND, it defines the α_x coordinate of the first grid point in each x-grid region.

AMU - Dimensioned NXEND, it defines the x-stretching coefficient (μ_x) in each grid region.

XA - This defines the coefficient a_x in Equation (2.2) for each grid region.

XB - This defines the coefficient b_x in Equation (2.2) for each grid region.

XC - This defines the coefficient c_x in Equation (2.2) for each grid region.

Group #3: NYEND, ALYREF, JN, YYY, YAL, AMV, YA, YB, YC

NYEND - This defines the total number of y-grid regions.

ALYREF - Reference length in the stretched y-grid (α_y).

JN - Dimensioned NYEND, it defines the J-grid index of the first grid point in each y-grid region.

YYY - Dimensioned NYEND, it defines the y-coordinate of the first grid point in each y-grid region.

YAL - Dimensioned NYEND, it defines the α_y-coordinate of the first grid point in each y-grid region.

AMV - Dimensioned NYEND, it defines the y-stretching coefficient (μ_y) in each grid region.

YA - This defines the coefficient a_y in Equation (2.2) for each grid region.

YB - This defines the coefficient b_y in Equation (2.2) for each grid region.

YC - This defines the coefficient c_y in Equation (2.2) for each grid region.

Group #4: T1W, T1E, PH1W, PH1E

T1W - Dimensioned JM, this defines the tidal amplitude along the western open boundary.

T1E - Dimensioned JM, this defines the tidal amplitude along the eastern open boundary.

PH1W - Dimensioned JM, this defines the tidal phase angle along the western open boundary.

PH1E - Dimensioned JM, this defines the tidal phase angle along the eastern open boundary.

Group #5: T2W, T2E, PH2W, PH2E

T2W - Dimensioned JM, this defines the amplitude of x-mass flux along the western open boundary.

T2E - Dimensioned JM, this defines the amplitude of x-mass flux along the eastern open boundary.

PH2W - Dimensioned JM, this defines the phase angle of x-mass flux along the western open boundary.

PH2E - Dimensioned JM, this defines the phase angle of x-mass flux along the eastern open boundary.

Group #6: T3S, T3N, PH3S, PH3N

T3S - Dimensioned IM, this defines the tidal amplitude along the southern open boundary.

T3N - Dimensioned IM, this defines the tidal amplitude along the northern open boundary.

PH3S - Dimensioned IM, this defines the tidal phase angle along the southern open boundary.

PH3N - Dimensioned IM, this defines the tidal phase angle along the northern open boundary.

Group #7: T4S, T4N, PH4S, PH4N

T4S - Dimensioned IM, this defines the amplitude of y-mass flux along the southern open boundary.

T4N - Dimensioned IM, this defines the amplitude of y-mass flux along the northern open boundary.

PH4S - Dimensioned JM, this defines the phase angle of y-mass flux along the southern open boundary.

PH4N - Dimensioned JM, this defines the phase angle of y-mass flux along the northern open boundary.

VIII. DATA REQUIREMENTS OF THE CODE

Several different types of data are required to perform a successful simulation with the code:

1. Data Required to Initiate the Code

1) Grid information which discretely defines the lateral geometry and bottom topography of the computational domain. The spatial scales of physical processes that the model can properly resolve depend on this information as well as the governing equations.

2) Time step information. The temporal scales of physical processes that the model can properly resolve depend on this information as well as the governing equations.

3) State variables at the initiation of the simulation. These include the flow variables as well as the water quality parameters. The importance to accurately define the initial state variables depends on the physical processes of interest and the intended duration of simulation.

2. Data Required to Operate the Code

1) Vertical boundary conditions. These include the specification of fluxes of momentum, heat, and species at the air-sea interface as well as the bottom. Alternatively, these conditions could be given in terms of the state variables instead of their fluxes.

2) Lateral boundary conditions. These include the specification of solid wall, river flows, and open boundary conditions. To make a successful simulation, the boundary conditions provided should be valid at all times of the simulation process.

3. Data Required to Calibrate the Code

Physical parameters of the code, e.g., eddy coefficients and bottom roughness elements, may vary significantly with time and space. To select the best physical parameters for a given environment, sufficient data are required for comparing with model results.

4. Data Required to Verify the Code

Extreme care should be exercised to ensure the proper comparison between model results and field/lab data. Both the processed field/lab data and the processed model results should contain the same temporal and spatial scales describing the physical phenomena of interest. Measurements of flow and concentration data usually consist of a finite set of data in a random field. At any given site, a long time series of data needs to be processed with proper time-averaging to produce proper mean values and variances for comparison with model results.

In reality, the amount of data that should be collected for a specific study often depends on many factors. The primary factors include 1) money available, 2) time available, 3) type of problem, 4) definition of impact criteria, 5) specific environment, 6) uncertainties in data, and 7) uncertainties in model results.

IX. APPLICATIONS OF THE CODE

The hydrodynamic model described herein has been applied to a variety of problems. Numerous applications of the model are described in Sheng (1983) and include the following: (1) Vicksburg tidal flume, (2) tidal currents in an open bight, (3) wind-driven currents in a shallow lake, (4) density-driven currents in an estuary, (5) wind-driven currents in a laboratory open channel, (6) formation and deepening of a thermocline, and (7) tide- and wind-driven currents in the Mississippi coastal waters.

As an example here, model simulation of the tide- and wind-driven currents in the Mississippi coastal waters of the Gulf of Mexico will be briefly described. The grid used is similar to the one used by Schmalz (1983) and contains a total of 116 grid points in the y-direction and 60 grid points in the x-direction (Figure 9.1). Either four or eight layers are used in the vertical. The water depths vary from a few meters deep inside the Mississippi Sound to over a 1000 m deep along the open boundaries. Boundary conditions along the eastern and southern boundaries are provided from a numerical tide model for the entire Gulf of Mexico (Reid and Whitaker 1981). Surface displacement data at several points along the boundaries are determined as a linear combination of the five major tidal constituents: K1, O1, P1, M2, and S2. In Figure 9.2, the model results for tides from September 20 to September 25, 1980 (GMT) are compared with measurements at four stations within the Mississippi Sound. Diurnal tides, which dominated during the earlier days, became less predominant while the semi-diurnal tides became gradually more apparent towards the end of the 5-day period. The measured data were filtered to remove short-period oscillations on the order of a few hours or less. A time step of 720 sec was generally used in the numerical computations. The tide-driven currents at mid-depth are also compared with data in Figure 9.3 for two stations in the Mississippi Sound. Currents on the order of 30 cm/sec exist at both stations. The horizontal velocity field at 1 m depth, after 3 days of simulation (Figure 9.4(a)), indicates relatively strong currents at the various tidal inlets and other shallow areas. Figure 9.4(b) shows the horizontal velocity field at 10 m depth. Winds were generally quite mild during the period and a numerical compuation with both the tide and the wind showed relatively little effect due to the wind.

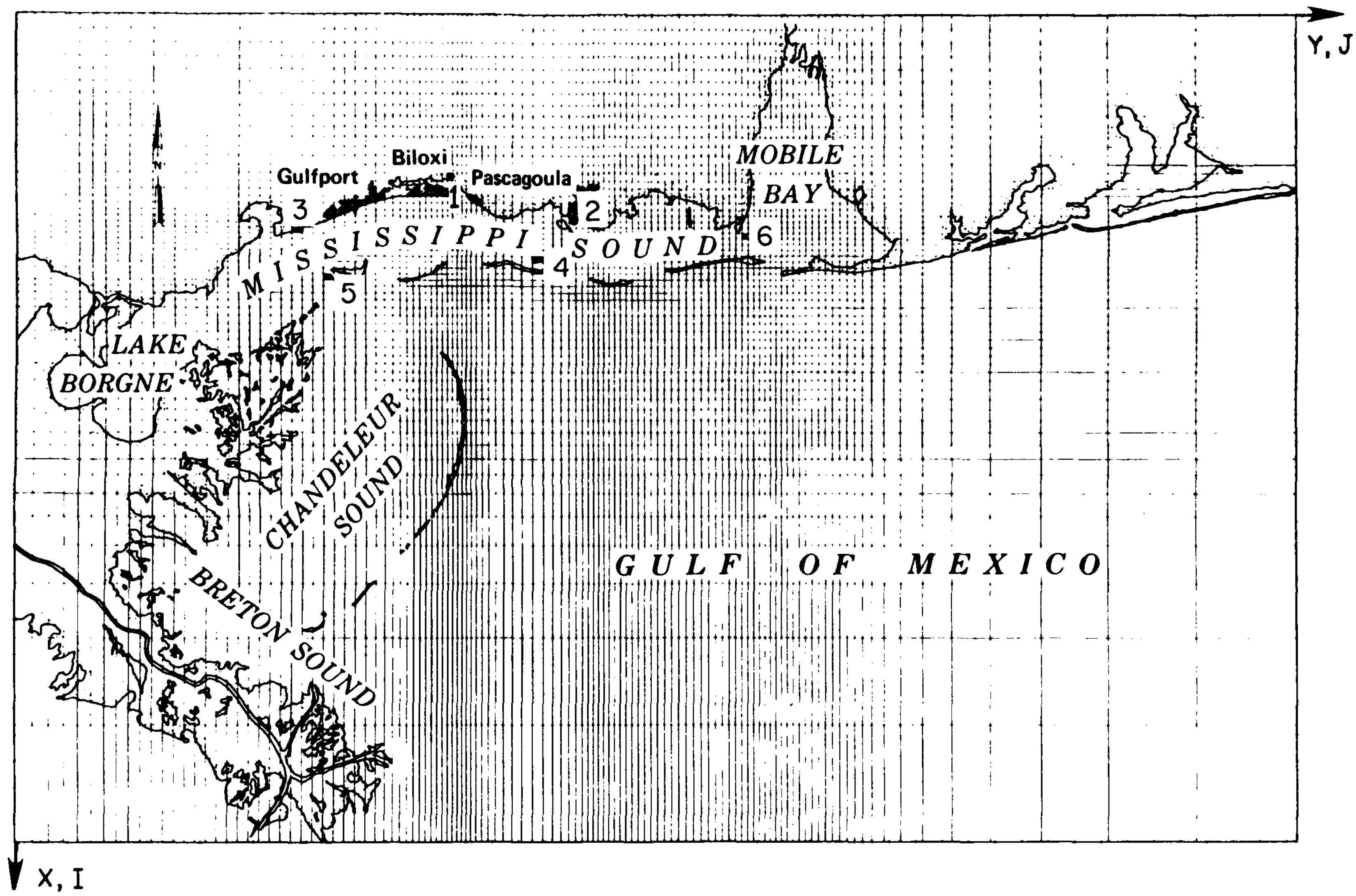


Figure 9.1 Lateral numerical grid used for dynamic simulation of coastal currents within the Mississippi coastal waters.

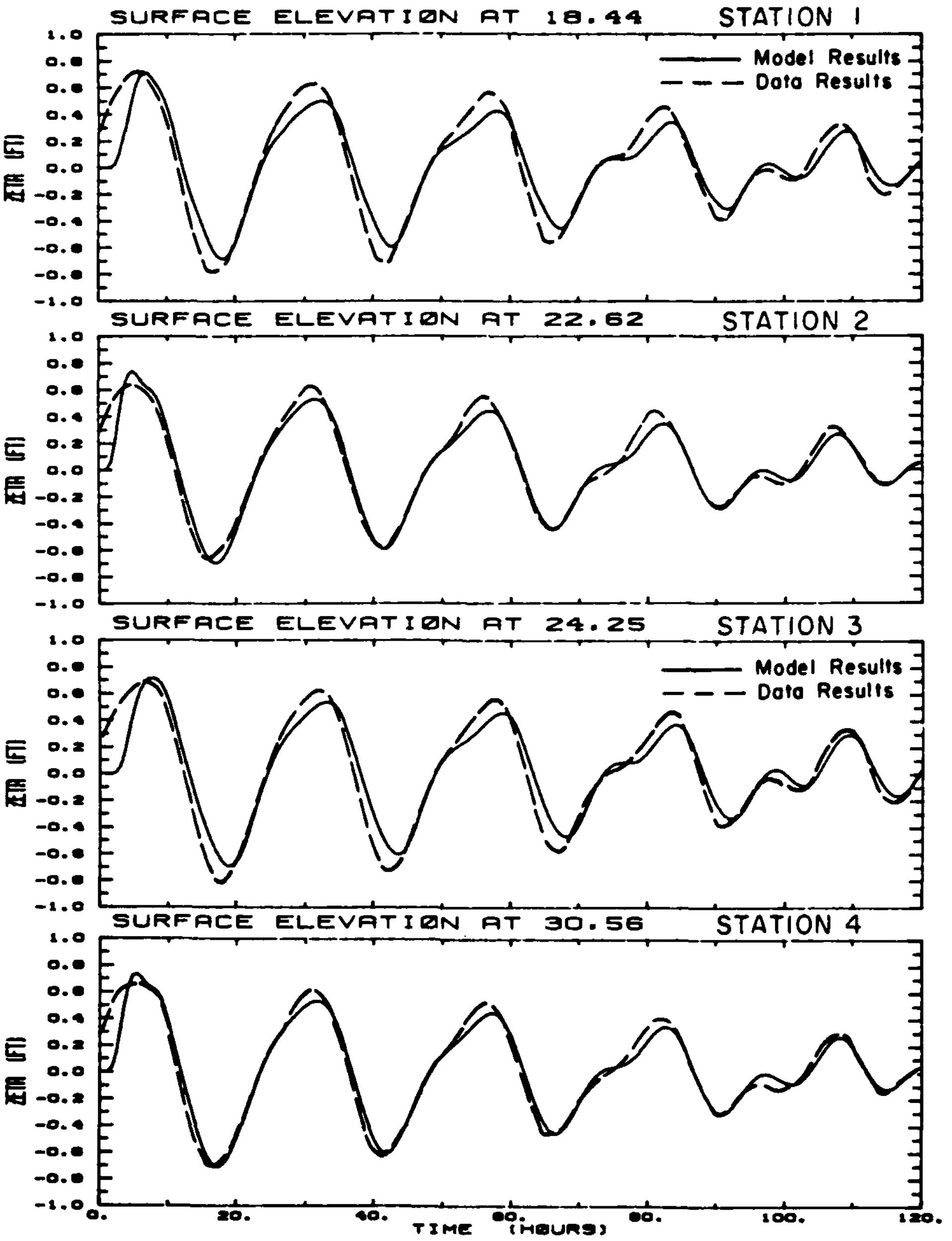


Figure 9.2. Transient variation of surface displacements at four stations within the Mississippi Sound from 9/20/80 to 9/25/80

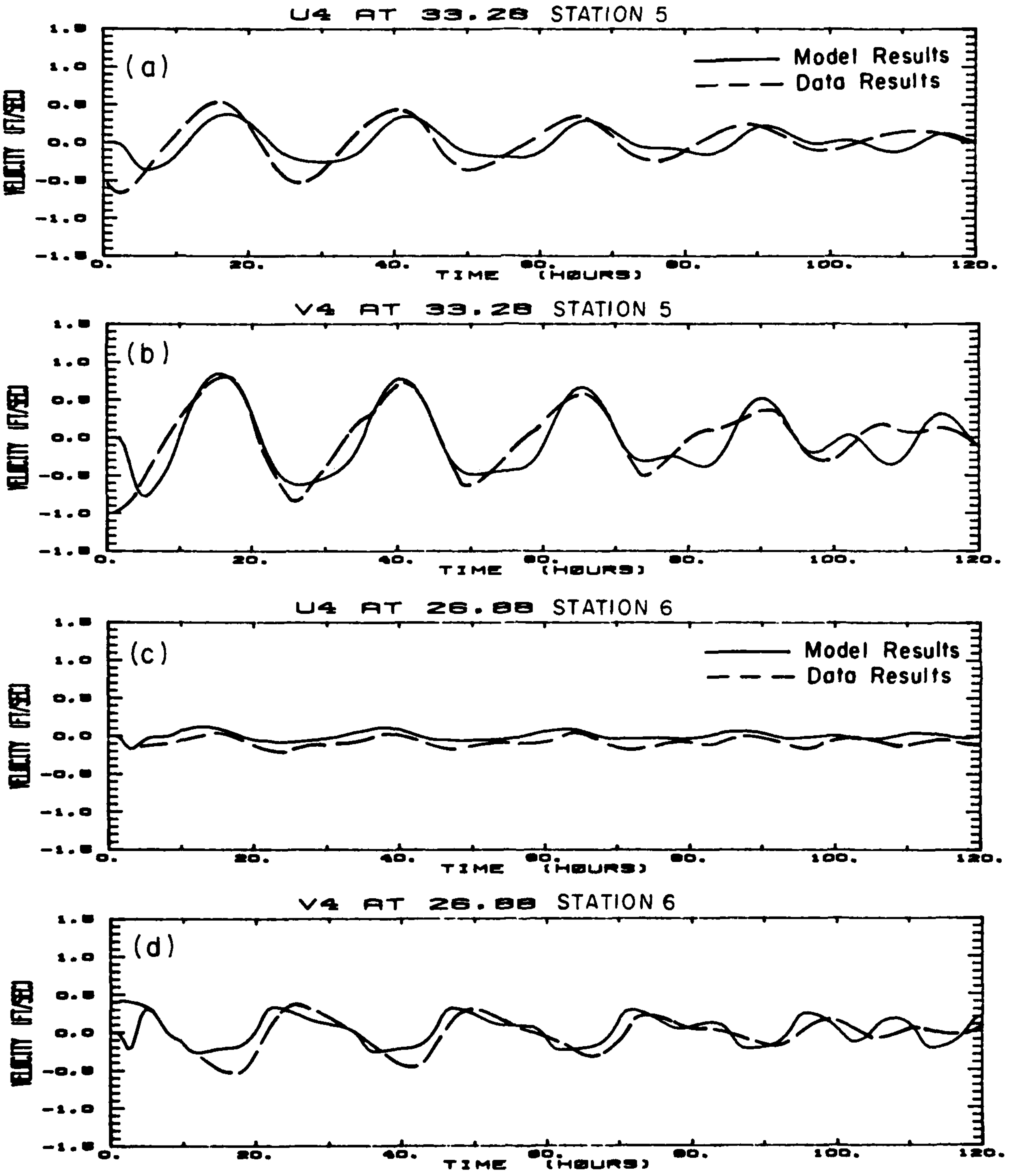


Figure 9.3. Transient variation of mid-depth velocities at two stations within the Mississippi Sound from 9/20/80 to 9/25/80.

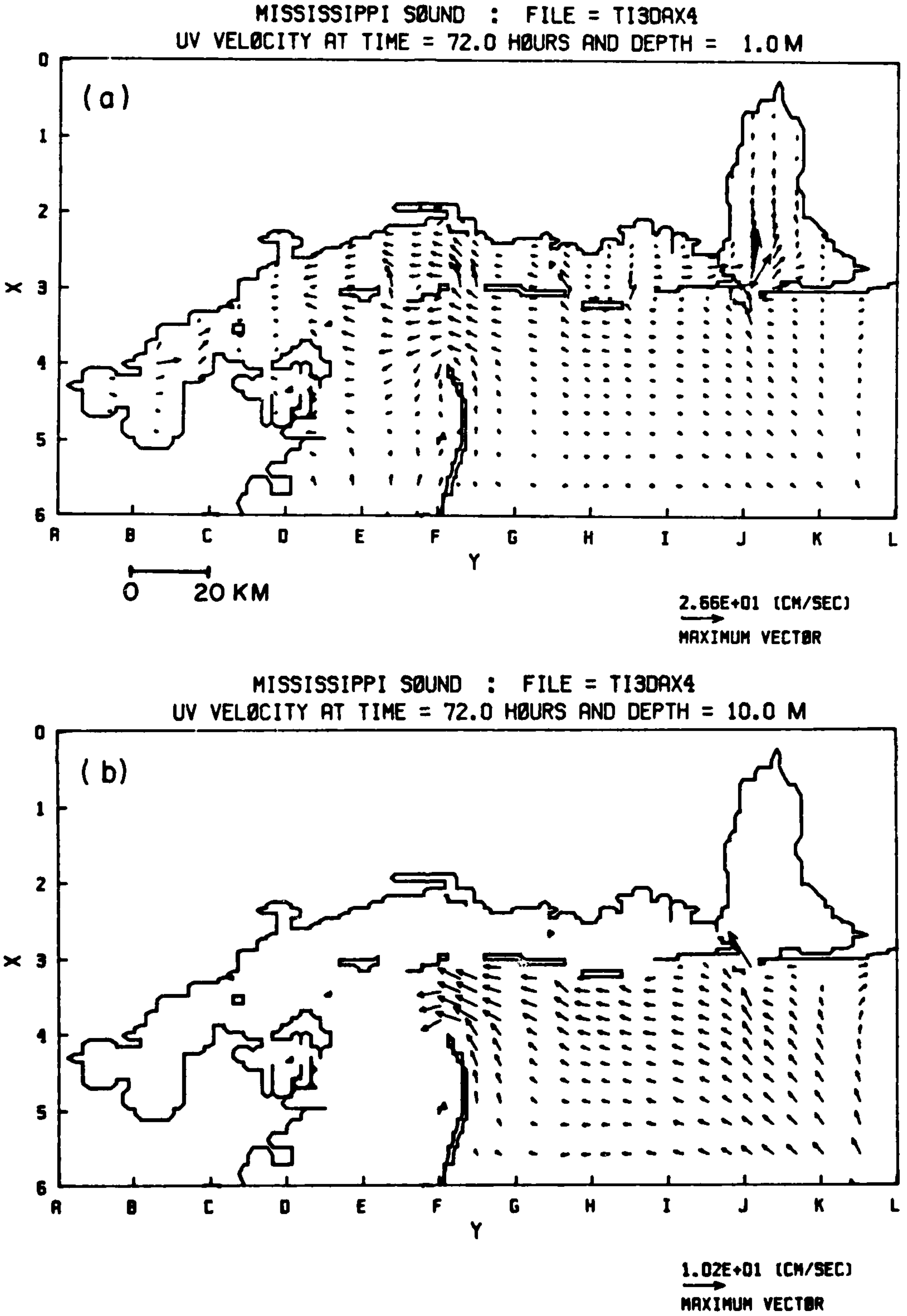


Figure 9.4 Horizontal velocity field at 0 hr, 9/23/80:

(a) 1 m depth

(b) 10 m depth

X. REFERENCES

Blumberg, A.F., 1975; "A Numerical Investigation into the Dynamics of Estuarine Circulation," Chesapeake Bay Institute Report No. 91, Baltimore, MD.

Bowden, K.F. and P. Hamilton, 1975; "Some Experiments with a Numerical Model of Circulation and Mixing in a Tidal Estuary," Est. Coastal Mar. Sci., 3, 281.

Cox, R.A., F. Culkin, and J.P. Riley, 1967; "The Electrical Conductivity/Chlorinity Relationship in Natural Sea Water," Deep-Sea Research, 14, pp. 203-220.

Kent, R.E. and D.W. Pritchard, 1959; "A Test of Mixing Length Theories in a Coastal Plain Estuary," J. Mar. Res., 18, pp. 62-72.

Munk, W.H. and E.P. Anderson, 1948; "Notes on the Theory of the Thermocline," J. Mar. Res., 1, pp. 276-295.

Reid, R.O. and R.E. Whitaker, 1981; "Numerical Model for Astronomical Tides in the Gulf of Mexico, Vol. I: Theory and Application," Dept. Oceanog., Texas A & M University, College Station, TX.

Richtmyer, R.D. and K.W. Morton, 1967; Difference Methods for Initial Value Problems, second edition, Interscience, John Wiley & Sons, New York, 405 pp.

Schmalz, R.A., 1983; "Numerical Model Investigation of Mississippi Sound and Adjacent Areas," WES Technical Report in Preparation, U.S. Army Eng. Waterways Experiment Station, Vicksburg, MS.

Sheng, Y.P., 1982; "Hydraulic Applications of a Second-Order Closure Model of Turbulent Transport," in Applying Research to Hydraulic Practice, (P. Smith, Ed.), Amer. Soc. Civil Eng., New York, pp. 106-119.

Sheng, Y.P., 1983; "Mathematical Modeling of Three-Dimensional Coastal Currents and Sediment Dispersion: Model Development and Application," A.R.A.P. Report No. 486, A.R.A.P. Inc., Princeton, NJ, 287 pp; also WES Technical Report (in Press), U.S. Army Eng. Waterways Experiment Station, Vicksburg, MS.

Sheng, Y.P., H. Segur, and W.S. Lewellen, 1978; "Application of a Spatial Smoothing Scheme to Control Short-Wave Numerical Oscillation," A.R.A.P. Tech Memo 78-8.

APPENDIX A: MEAN EQUATIONS OF MOTION

The modeled Cartesian equations of motion for the mean variables may be written in the vertically and horizontally stretched coordinate system (α,γ,σ) instead of the original system (x,y,z) shown in Figure 1. For clarity in the following, α and γ have been substituted by x and y.

$$\frac{\partial \zeta}{\partial t} + \beta \left(\frac{\partial U}{\mu_x \partial x} + \frac{\partial V}{\mu_y \partial y} \right) = 0 \qquad \text{(A.1)}$$

$$\frac{\partial U}{\partial t} = - \frac{H}{\mu_x} \frac{\partial \zeta}{\partial x} + V - \frac{Ro}{H} \left[\frac{\partial}{\mu_x \partial x} \left(\frac{U^2}{H} \right) + \frac{\partial}{\mu_y \partial y} \left(\frac{UV}{H} \right) \right]$$

$$- \frac{Ro}{Fr^2} \int_{-1}^{0} \left[H \int_{\sigma}^{0} \frac{\partial \rho}{\mu_x \partial x} d\sigma + \frac{\partial H}{\mu_x \partial x} \left(\int_{\sigma}^{0} \rho d\sigma + \sigma \rho \right) + \frac{H}{\mu_x} \frac{\partial p_a}{\partial x} \right] H d\sigma$$

$$+ \tau_{sx} - \tau_{bx} + \int_{-1}^{0} H \, (H.D.)_x \, d\sigma \equiv - \frac{H}{\mu_x} \frac{\partial \zeta}{\partial x} + D_x \qquad \text{(A.2)}$$

$$\frac{\partial V}{\partial t} = - \frac{H}{\mu_y} \frac{\partial \zeta}{\partial y} - U - \frac{Ro}{H} \left[\frac{\partial}{\mu_x \partial x} \left(\frac{UV}{H} \right) + \frac{\partial}{\mu_y \partial y} \left(\frac{V^2}{H} \right) \right]$$

$$- \frac{Ro}{Fr^2} \int_{-1}^{0} \left[H \int_{\sigma}^{0} \frac{\partial \rho}{\mu_y \partial y} d\sigma + \frac{\partial H}{\mu_y \partial y} \left(\int_{\sigma}^{0} \rho d\sigma + \sigma \rho \right) + \frac{H}{\mu_y} \frac{\partial p_a}{\partial y} \right] H d\sigma$$

$$+ \tau_{sy} - \tau_{by} + \int_{-1}^{0} H \, (H.D.)_y \, d\sigma \equiv - \frac{H}{\mu_y} \frac{\partial \zeta}{\partial y} + D_y \qquad \text{(A.3)}$$

$$\frac{1}{H}\frac{\partial Hu}{\partial t} = -\frac{\partial \zeta}{\mu_x \partial x} + v - \frac{Ro}{H}\left(\frac{\partial Huu}{\mu_x \partial x} + \frac{\partial Huv}{\mu_y \partial y} + \frac{\partial Hu\omega}{\partial \sigma}\right) + \frac{E_v}{H^2}\frac{\partial}{\partial \sigma}\left(A_v \frac{\partial u}{\partial \sigma}\right)$$

$$-\frac{Ro}{Fr^2 \mu_x}\left[H\int_\sigma^0 \frac{\partial \rho}{\partial x}\,d\sigma + \frac{\partial H}{\partial x}\left(\int_\sigma^0 \rho d\sigma + \sigma\rho\right) + \frac{\partial p_a}{\partial x}\right] \qquad (A.4)$$

$$+ E_H\left[\frac{\partial}{\mu_x \partial x}\left(A_H \frac{\partial u}{\mu_x \partial x}\right) + \frac{\partial}{\mu_y \partial y}\left(A_H \frac{\partial u}{\mu_y \partial y}\right) + \text{H.O.T.}\right]$$

$$\equiv -\frac{\partial \zeta}{\mu_y \partial_y} + \frac{E_v}{H^2}\frac{\partial}{\partial \sigma}\left(A_v \frac{\partial u}{\partial \sigma}\right) + B_y$$

$$\frac{1}{H}\frac{\partial Hv}{\partial t} = -\frac{\partial \zeta}{\mu_y \partial y} - u - \frac{Ro}{H}\left(\frac{\partial Huv}{\mu_x \partial x} + \frac{\partial Hvv}{\mu_y \partial y} + \frac{\partial Hv\omega}{\partial \sigma}\right) + \frac{E_v}{H^2}\frac{\partial}{\partial \sigma}\left(A_v \frac{\partial v}{\partial \sigma}\right)$$

$$-\frac{Ro}{Fr^2 \mu_y}\left[H\int_\sigma^0 \frac{\partial \rho}{\partial y}\,d\sigma + \frac{\partial H}{\partial y}\left(\int_\sigma^0 \rho d\sigma + \sigma\rho\right) + \frac{\partial p_a}{\partial y}\right] \qquad (A.5)$$

$$+ E_H\left[\frac{\partial}{\mu_x \partial x}\left(A_H \frac{\partial v}{\mu_x \partial x}\right) + \frac{\partial}{\mu_y \partial y}\left(A_H \frac{\partial v}{\mu_y \partial y}\right) + \text{H.O.T.}\right]$$

$$\equiv -\frac{\partial \zeta}{\mu_y \partial y} + \frac{E_v}{H^2}\frac{\partial}{\partial \sigma}\left(A_v \frac{\partial v}{\partial \sigma}\right) + B_y$$

$$\omega = - \frac{\beta}{H} \int_{-1}^{\sigma} \left(\frac{\partial Hu}{\mu_x \partial x} + \frac{\partial Hv}{\mu_y \partial y} \right) d\sigma + \frac{\beta(1+\sigma)}{H} \int_{-1}^{0} \left(\frac{\partial Hu}{\mu_x \partial x} + \frac{\partial Hv}{\mu_y \partial y} \right) d\sigma \qquad (A.6)$$

$$w = H\omega + \frac{1+\sigma}{\beta} \frac{\partial \zeta}{\partial t} \qquad (A.7)$$

$$\frac{1}{H} \frac{\partial HT}{\partial t} = - \frac{Ro}{H} \left(\frac{\partial HuT}{\mu_x \partial x} + \frac{\partial HvT}{\mu_y \partial y} + H \frac{\partial \omega T}{\partial \sigma} \right)$$

$$+ \frac{Ro}{Pe_v D/L} \frac{\partial}{H^2 \partial \sigma} \left(K_v \frac{\partial T}{\partial \sigma} \right) \qquad (A.8)$$

$$+ \frac{Ro}{Pe_H H} \left[\frac{\partial}{\mu_x \partial x} \left(K_H \frac{\partial HT}{\mu_x \partial x} \right) + \frac{\partial}{\mu_y \partial y} \left(K_H \frac{\partial HT}{\mu_y \partial y} \right) + H.O.T. \right]$$

$$\frac{1}{H} \frac{\partial HS}{\partial t} = - \frac{Ro}{H} \left(\frac{\partial HuS}{\mu_x \partial x} + \frac{\partial HvS}{\mu_y \partial y} + H \frac{\partial \omega S}{\partial \sigma} \right)$$

$$+ \frac{Ro}{Sc_v D/L} \frac{\partial}{H^2 \partial \sigma} \left(D_v \frac{\partial S}{\partial \sigma} \right) \qquad (A.9)$$

$$+ \frac{Ro}{Sc_H H} \left[\frac{\partial}{\mu_x \partial x} \left(D_H \frac{\partial HS}{\mu_x \partial x} \right) + \frac{\partial}{\mu_y \partial y} \left(D_H \frac{\partial HS}{\mu_y \partial y} \right) + H.O.T. \right]$$

$$\rho = \rho(T,S) \tag{A.10}$$

$$\frac{1}{H}\frac{\partial HC}{\partial t} = - \frac{Ro}{H}\left[\frac{\partial HuC}{\mu_x \partial x} + \frac{\partial HvC}{\mu_y \partial y} + H\,\frac{\partial(\omega+\omega_s)C}{\partial\sigma}\right]$$

$$+ \frac{Ro}{Sc_v D/L}\,\frac{\partial}{H^2\partial\sigma}\left(D_v\,\frac{\partial C}{\partial\sigma}\right)$$

$$+ \frac{Ro}{S_{c_H} H}\left[\frac{\partial}{\mu_x\partial x}\left(D_H\,\frac{\partial HC}{\mu_x\partial x}\right) + \frac{\partial}{\mu_y\partial y}\left(D_H\,\frac{\partial HC}{\mu_y\partial y}\right) + \text{H.O.T.}\right] \tag{A.11}$$

where all the variables and dimensionless parameters are defined in Sheng (1983). The exact form of the equation of state can also be found in the same report.

In the actual numerical computations, the external variables (ζ,U,V) are first solved from equations (A.1), (A.2), and (A.3). After that, the perturbation horizontal velocities (which are generally not infinitesimal) $u' \equiv u-U/H$ are solved from the difference between equations (A.4) and (A.2), while $v' \equiv v-V/H$ from the difference between equations (A.5) and (A.3):

$$\frac{1}{H}\frac{\partial Hu'}{\partial t} = B_x - \frac{D_x}{H} + \frac{1}{H^2}\frac{\partial}{\partial\sigma}\left[A_v\,\frac{\partial}{\partial\sigma}\left(\frac{Hu'+U}{H}\right)\right] \tag{A.12}$$

$$\frac{1}{H}\frac{\partial Hv'}{\partial t} = B_y - \frac{D_y}{H} + \frac{1}{H^2}\frac{\partial}{\partial\sigma}\left[A_v\,\frac{\partial}{\partial\sigma}\left(\frac{Hv'+V}{H}\right)\right] \tag{A.13}$$

Following these, u and v are computed by adding u' and v' with U/H and V/H, respectively. Equations (A.6) through (A.11) are solved thereafter.

APPENDIX B: DEPENDENT VARIABLES

Physical Name	FORTRAN Name	Array Size
ζ	S	JM, IM
U	UI	JM, IM
V	VI	JM, IM
ρ	R, RU	KM, JM, IM
A_V, K_V, D_V	GA, GB	KM, JM, IM
A_H, K_H, D_H	AH, AH, AH	
u	U	KM, JM, IM
v	V	KM, JM, IM
ω	W	KM, JM, IM
w	WW	KM, JM, IM
T	T	KM, JM, IM
S	SA	KM, JM, IM
C	C	KM, JM, IM
h	HU, HV, HS	JM, IM