SEMI-IMPLICIT FINITE DIFFERENCE METHODS FOR THREE-DIMENSIONAL SHALLOW WATER FLOW

VINCENZO CASULLI

Dipartimento di Matematica, Università di Trento, I-38050 Povo (TN), Italy

AND

RALPH T. CHENG

U.S. Geological Survey, WRD, Menlo Park, CA, U.S.A.

SUMMARY

A semi-implicit finite difference method for the numerical solution of three-dimensional shallow water flows is presented and discussed. The governing equations are the primitive three-dimensional turbulent mean flow equations where the pressure distribution in the vertical has been assumed to be hydrostatic. In the method of solution a minimal degree of implicitness has been adopted in such a fashion that the resulting algorithm is stable and gives a maximal computational efficiency at a minimal computational cost. At each time step the numerical method requires the solution of one large linear system which can be formally decomposed into a set of small three-diagonal systems coupled with one five-diagonal system. All these linear systems are symmetric and positive definite. Thus the existence and uniqueness of the numerical solution are assured. When only one vertical layer is specified, this method reduces as a special case to a semi-implicit scheme for solving the corresponding two-dimensional shallow water equations. The resulting two- and three-dimensional algorithm has been shown to be fast, accurate and mass-conservative and can also be applied to simulate flooding and drying of tidal mud-flats in conjunction with three-dimensional flows. Furthermore, the resulting algorithm is fully vectorizable for an efficient implementation on modern vector computers.

KEY WORDS Three-dimensional Semi-implicit Shallow water

1. INTRODUCTION

Several numerical methods for the time-dependent two- and three-dimensional shallow water equations are known in the current literature and are now widely used in practical applications (see References 1–3 and the numerous references cited therein). The time integration schemes of these methods range from fully explicit to fully implicit. A fully explicit finite difference method is relatively simple to implement and easily vectorizable. However, a severe limitation exists for standard explicit numerical methods owing to the propagation of surface gravity waves. This restriction, known as the Courant–Friedrich–Lewy (CFL) stability criterion,[4] usually requires a much smaller time step in the numerical integration than permitted by accuracy considerations. Several existing numerical models for two- and three-dimensional shallow water flow simulations are based on an alternating direction implicit (ADI) method. ADI methods result in computational efficiency superior to fully explicit methods because their improved stability allows larger time steps to be employed. However, a source of inaccuracy, known as the ADI effect,[3,5,6] arises

0271-2091/92/180629-20$15.00

Received September 1991
Revised January 1992

when these methods are used with large time steps in flow domains characterized by complex geometries. The ADI effect can be reduced by limiting the time step, but this also limits the efficiency of these methods.

In recent years, more robust two-dimensional shallow water flow simulators which are economically competitive with ADI methods have been developed and applied. These methods include semi-implicit[7,8] as well as fully implicit[9,10] splitting methods. In semi-implicit methods only the barotropic pressure gradient in the momentum equations and the velocity divergence in the continuity equation are taken implicitly. Computationally, at each time step a linear five-diagonal system is solved in which the new water surface elevations for the entire domain are the unknowns. The matrix coefficient for such a system is symmetric and positive definite and its solution can be determined uniquely and efficiently by using a conjugate gradient method.[11] The fully implicit time-splitting methods use two or more fractional time steps which essentially decouple the propagation operator from convection and diffusion. Each of these operators is then discretized implicitly.[10]

Several three-dimensional numerical models have been reported in the literature[12–16] and applications of three-dimensional models for solutions of practical problems are becoming a reality with the aid of modern computers. For example, the model developed by Blumberg and Mellor[17,18] has been used successfully by the authors and several other investigators in numerous practical applications and in studies of various important waterways (see e.g. References 19–23). The model is generally explicit with the exception that the vertical eddy viscosity terms are discretized implicitly. Since the model must simulate both the velocity field and the propagation of fast-moving gravity waves, a technique known as 'mode splitting'[24,25] has been used just as in several other three-dimensional models. In the model formulation the governing system of equations is split into an external mode and an internal mode. A system of two-dimensional vertically integrated equations (external mode) is solved independently from the three-dimensional equations (internal mode). Using the external mode in two dimensions permits the efficient calculation of the free surface gravity wave propagation even if a small integration time step is required. The solution of the more computationally intensive three-dimensional internal mode equations can be achieved by using a large time step, because the integration is no longer limited by a stability associated with the gravity wave propagation.

When mode splitting is used, care must be exercised to ensure the consistency of the physical quantities derived from the internal and external modes. Notably, the vertically integrated velocities in the external mode should be the same as the vertical integration of the velocity profile from the solution of the internal mode. Additionally, the representation of the bottom stress in the internal and external modes must also be consistent. Mathematically, if any of these inconsistencies exist, then the convergence of the numerical solution cannot be assured because the finite difference approximations for the respective modes are not consistent.

Because of these potential difficulties, the numerical method presented in this paper does not incorporate mode splitting but instead solves directly the primitive three-dimensional governing equations. These equations are discretized and solved by a semi-implicit technique which accomplishes the necessary objective that the stability of the scheme does not depend upon the celerity. When the convective and viscous terms are discretized using a Eulerian–Lagrangian approach, the resulting algorithm is also shown to be stable under a mild stability condition. Thus the use of large time steps is permitted along with the benefit of improvements in both computational efficiency and accuracy. Computationally, fixed staggered finite difference meshes are used on horizontal planes and the dependent variables are defined in fixed vertical layers. At each time step, only the barotropic terms and the vertical viscosity terms are finite differenced implicitly in the horizontal momentum equations. Momentum exchanges between vertical layers are expressed in a set of tridiagonal matrix equations relating the discrete horizontal velocities in each vertical level to the gradient of the water surface elevations (barotropic pressure gradient). A formal expression for the solution of these tridiagonal systems can be written in terms of the barotropic pressure gradient. Substituting the formal solutions into the vertically integrated continuity equation gives rise to a linear five-diagonal system whose only unknowns are the water surface elevation over the domain of interest. Such a system is symmetric and positive definite and can be solved uniquely and efficiently by using a conjugate gradient method. By direct substitution of the barotropic pressure gradient known at the advanced time level, the horizontal velocity for each vertical layer can be computed. Finally, the vertical velocity component can be found by integration of the continuity equation. In the particular case where only one vertical layer is used to represent the three-dimensional system, the present formulation reduces naturally to the semi-implicit finite difference method of solution for the vertically averaged shallow water equations.[7,11] In the present numerical solution scheme most of the required arithmetic operations can be made independent of each other and are thus highly vectorizable for efficient implementation on vector computers.

This paper should be considered as phase one of the development of a general three-dimensional model (TRIM-3D). The main purpose of this paper is to establish a firm mathematical foundation for the numerical scheme and computational algorithm for the numerical solutions of two- and three-dimensional geophysical flow problems. To this end, only the special case of constant density flow is considered. Obviously, numerical investigations of three-dimensional flow problems are not complete unless proper representations of vertical turbulent mixing, transport of salt, variations in density distributions and coupling of salt transport through baroclinic forcing are addressed. These subjects will be the focus of phase two of the development of the TRIM-3D model.

2. GOVERNING EQUATIONS

The governing three-dimensional, primitive variable equations describing constant density, free surface flows in estuarine embayments and coastal oceans can be derived from the Navier–Stokes equations after turbulent averaging and under the simplifying assumption that the pressure is hydrostatic.[2] Such equations have the form

$$\begin{aligned}
&\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}=-g\frac{\partial \eta}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+\frac{\partial}{\partial z}\left(\nu\frac{\partial u}{\partial z}\right)+fv,\\
&\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}=-g\frac{\partial \eta}{\partial y}+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}\right)+\frac{\partial}{\partial z}\left(\nu\frac{\partial v}{\partial z}\right)-fu,\\
&\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0,
\end{aligned}\tag{1}$$

where $u(x, y, z, t)$, $v(x, y, z, t)$ and $w(x, y, z, t)$ are the velocity components in the horizontal x, y- and in the vertical z-direction respectively, t is the time, $\eta(x, y, t)$ is the water surface elevation measured from the undisturbed water surface, g is the constant gravitational acceleration, f is the Coriolis parameter, assumed to be constant, and μ and ν are the coefficients of horizontal and vertical eddy viscosity respectively. In applications to large shallow embayments the ratio of vertical length scale to horizontal length scale is very small. As a consequence the horizontal eddy viscosity terms are typically orders of magnitude smaller than the vertical viscosity terms and their effect is normally small and obscured by numerical diffusion. Therefore most models either neglect these terms or simply use a constant horizontal eddy viscosity coefficient. A more refined treatment of these terms is usually not justified.

Integrating the continuity equation over the depth and using a kinematic condition at the free surface leads to the free surface equation

$$\frac{\partial\eta}{\partial t}+\frac{\partial}{\partial x}\left(\int_{-h}^{\eta}u\,\mathrm{d}z\right)+\frac{\partial}{\partial y}\left(\int_{-h}^{\eta}v\,\mathrm{d}z\right)=0, \tag{2}$$

where $h(x, y)$ is the water depth measured from the undisturbed water surface. $H(x, y, t)$ will be used to denote the total water depth, $H(x, y, t)=h(x, y)+\eta(x, y, t)$.

The boundary conditions at the free surface are specified by the prescribed wind stresses τ_x^w and τ_y^w:

$$v\frac{\partial u}{\partial z}=\tau_x^w, \qquad v\frac{\partial v}{\partial z}=\tau_y^w. \tag{3}$$

The boundary conditions at the bottom are given by expressing the bottom stress in terms of the velocity components taken from values of the layer adjacent to the sediment–water interface. The bottom stress can be related to the turbulent law of the wall, a drag coefficient associated with quadratic velocity or using a Manning–Chezy formula such as

$$v\frac{\partial u}{\partial z}=\frac{g\sqrt{(u^2+v^2)}}{Cz^2}u, \qquad v\frac{\partial v}{\partial z}=\frac{g\sqrt{(u^2+v^2)}}{Cz^2}v, \tag{4}$$

where Cz is the Chezy friction coefficient.

In lieu of solutions of the complete three-dimensional governing equations, the flow and circulation in a class of well-mixed estuaries and coastal embayments can be satisfactorily represented by the solutions of a set of vertically averaged shallow water equations. The system of vertically averaged momentum equations can be derived by integrating vertically equations (1) from the sea bed $z=-h$ to the free surface $z=\eta$. By using the free surface equation (2) and the boundary conditions (3) and (4), and after standard approximations on the non-linear convective terms, one gets the two-dimensional, vertically averaged shallow water equations[26]

$$\begin{aligned}
&\frac{\partial U}{\partial t}+U\frac{\partial U}{\partial x}+V\frac{\partial U}{\partial y}=-g\frac{\partial\eta}{\partial x}+\mu\left(\frac{\partial^2 U}{\partial x^2}+\frac{\partial^2 U}{\partial y^2}\right)+\frac{\tau_x^w}{H}-\frac{g\sqrt{(U^2+V^2)}}{Cz^2H}U+fV,\\
&\frac{\partial V}{\partial t}+U\frac{\partial V}{\partial x}+V\frac{\partial V}{\partial y}=-g\frac{\partial\eta}{\partial y}+\mu\left(\frac{\partial^2 V}{\partial x^2}+\frac{\partial^2 V}{\partial y^2}\right)+\frac{\tau_y^w}{H}-\frac{g\sqrt{(U^2+V^2)}}{Cz^2H}V-fU,\\
&\frac{\partial\eta}{\partial t}+\frac{\partial(HU)}{\partial x}+\frac{\partial(HV)}{\partial y}=0,
\end{aligned} \tag{5}$$

where $U=(1/H)\int_{-h}^{\eta}u\,\mathrm{d}z$ and $V=(1/H)\int_{-h}^{\eta}v\,\mathrm{d}z$ are the depth-averaged horizontal velocities.

Equations (5) constitute a system of three partial differential equations with three unknown functions $U(x, y, t)$, $V(x, y, t)$ and $\eta(x, y, t)$. It will be shown that the three-dimensional numerical model, to be described next, reduces to a two-dimensional system in the shallow part of a three-dimensional domain. Moreover, the two-dimensional system is consistent with equations (5) and contributes to a major improvement in the computational efficiency of the three-dimensional model.

3. A THREE-DIMENSIONAL SEMI-IMPLICIT NUMERICAL METHOD

A characteristic analysis of the two-dimensional, vertically integrated shallow water equations has shown that the celerity term $\sqrt{(gH)}$ in the equation for the characteristic cone arises from the barotropic pressure gradient in the momentum equations and from the velocity derivatives in the free surface equation.[7] A rigorous stability analysis was also provided by using the von Neumann method on the corresponding linearized scheme. Results of this analysis have led to a practical semi-implicit method of solution for the two-dimensional shallow water equations which has proven to be very useful in several applications.[11] On the basis of these previous studies a new, semi-implicit numerical method of solution for the three-dimensional free surface flow equations (1) will be derived in which the gradient of surface elevation in the momentum equations and the velocity in the free surface equation (2) will be discretized implicitly. The convective, Coriolis and horizontal viscosity terms in the momentum equations, however, will be discretized explicitly, but in order to eliminate a stability condition due to the vertical eddy viscosity, the vertical mixing terms will be discretized implicitly as well.

As shown in Figure 1, a spatial mesh which consists of rectangular cells of length Δx, width Δy and height Δz_k is introduced. Each cell is numbered at its centre with indices i, j and k. The discrete u-velocity is then defined at half-integer i and integers j and k; v is defined at integers i and k and half-integer j; w is defined at integers i and j and half-integer k. Finally, η is defined at integers i and j. The water depth $h(x, y)$ is specified at the u and v horizontal points. Then a general semi-implicit discretization of the momentum equations in (1) takes the form

$$u^{n+1}_{i+1/2,j,k}=Fu^{n}_{i+1/2,j,k}-g\frac{\Delta t}{\Delta x}(\eta^{n+1}_{i+1,j}-\eta^{n+1}_{i,j})+\Delta t\frac{\nu_{k+1/2}\dfrac{u^{n+1}_{i+1/2,j,k+1}-u^{n+1}_{i+1/2,j,k}}{\Delta z_{i+1/2,j,k+1/2}}-\nu_{k-1/2}\dfrac{u^{n+1}_{i+1/2,j,k}-u^{n+1}_{i+1/2,j,k-1}}{\Delta z_{i+1/2,j,k-1/2}}}{\Delta z_{i+1/2,j,k}}, \tag{6}$$

$$v^{n+1}_{i,j+1/2,k}=Fv^{n}_{i,j+1/2,k}-g\frac{\Delta t}{\Delta y}(\eta^{n+1}_{i,j+1}-\eta^{n+1}_{i,j})+\Delta t\frac{\nu_{k+1/2}\dfrac{v^{n+1}_{i,j+1/2,k+1}-v^{n+1}_{i,j+1/2,k}}{\Delta z_{i,j+1/2,k+1/2}}-\nu_{k-1/2}\dfrac{v^{n+1}_{i,j+1/2,k}-v^{n+1}_{i,j+1/2,k-1}}{\Delta z_{i,j+1/2,k-1/2}}}{\Delta z_{i,j+1/2,k}}, \tag{7}$$

where $\Delta z_{i+1/2,j,k}$ and $\Delta z_{i,j+1/2,k}$ are in general the thickness of the kth water layer more simply denoted by Δz_k. If, however, a vertical face of the box is not fully filled (because either the bottom or the free surface crosses a vertical face of the kth box), then $\Delta z_{i+1/2,j,k}$ and/or $\Delta z_{i,j+1/2,k}$ are defined to be the wetted height of the corresponding face. If, in particular, both the bottom and the free surface cross the same vertical face of the kth box, then of course $\Delta z_{i+1/2,j,k}$ or $\Delta z_{i,j+1/2,k}$ will be equal to the total water depth $H=h+\eta$ at that point. Finally, $\Delta z_{i+1/2,j,k+1/2}$ is defined to be the average of $\Delta z_{i+1/2,j,k}$ and $\Delta z_{i+1/2,j,k+1}$. Strictly speaking, since the thickness of the surface layer will depend on the position of the free surface and since the free surface changes with time, then the surface Δz will also change with the time level n. For simplicity, however, we will omit the index n from the notation. In (6) and (7) F is an explicit, non-linear finite difference operator which includes the explicit discretization of the substantial derivatives (convective terms) $u_t+uu_x+vu_y+wu_z$ and $v_t+uv_x+vv_y+wv_z$, the horizontal eddy viscosity terms and the Coriolis terms. A particular form for F can be chosen in a variety of ways which will be discussed later.

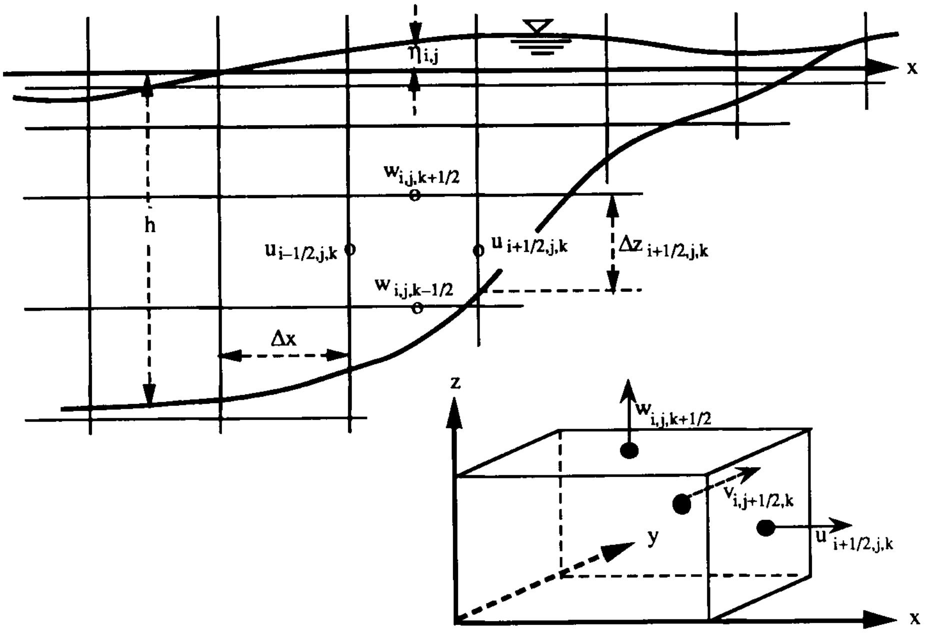


Figure 1. Schematic diagram of computational mesh and notations

Let m and M denote the k-index of the bottom and of the top finite difference stencil respectively. Since both m and M vary with the spatial position and since the value of M varies also with time, a precise index notation for m and M should be $m_{i+1/2,j}$, $m_{i,j+1/2}$, $M^n_{i+1/2,j}$, $M^n_{i,j+1/2}$. For notational simplicity, however, these indices will be omitted. Using this notation, the boundary conditions at the free surface and at the sediment–water interface, (3) and (4), are written in difference form as

$$\nu_{M+1/2}\frac{u^{n+1}_{i+1/2,j,M+1}-u^{n+1}_{i+1/2,j,M}}{\Delta z_{i+1/2,j,M+1/2}}=\tau^{w}_{x},\qquad \nu_{M+1/2}\frac{v^{n+1}_{i,j+1/2,M+1}-v^{n+1}_{i,j+1/2,M}}{\Delta z_{i,j+1/2,M+1/2}}=\tau^{w}_{y},\tag{8}$$

$$\begin{aligned}\nu_{m-1/2}\frac{u^{n+1}_{i+1/2,j,m}-u^{n+1}_{i+1/2,j,m-1}}{\Delta z_{i+1/2,j,m-1/2}}&=\frac{g\sqrt{[(u^{n}_{i+1/2,j,m})^2+(v^{n}_{i+1/2,j,m})^2]}}{Cz^2}\,u^{n+1}_{i+1/2,j,m},\\ \nu_{m-1/2}\frac{v^{n+1}_{i,j+1/2,m}-v^{n+1}_{i,j+1/2,m-1}}{\Delta z_{i,j+1/2,m-1/2}}&=\frac{g\sqrt{[(u^{n}_{i,j+1/2,m})^2+(v^{n}_{i,j+1/2,m})^2]}}{Cz^2}\,v^{n+1}_{i,j+1/2,m}.\end{aligned}\tag{9}$$

In equations (8) and (9) the values of u and v at levels $M+1$ and $m-1$ are fictitious; they also appear in the finite difference equations (6) and (7) when $k=M$ and m. By substituting the boundary conditions (8) and (9) in (6) and (7), the fictitious values will be replaced by values of u and v defined within the domain of interest. Equations (6) and (7) with the respective boundary conditions (8) and (9) can be written in the more compact matrix form

$$\mathbf{A}^{n}_{i+1/2,j}\mathbf{U}^{n+1}_{i+1/2,j}=\mathbf{G}^{n}_{i+1/2,j}-g\frac{\Delta t}{\Delta x}(\eta^{n+1}_{i+1,j}-\eta^{n+1}_{i,j})\Delta\mathbf{Z}^{n}_{i+1/2,j},\tag{10}$$

$$\mathbf{A}^{n}_{i,j+1/2}\mathbf{V}^{n+1}_{i,j+1/2}=\mathbf{G}^{n}_{i,j+1/2}-g\frac{\Delta t}{\Delta y}(\eta^{n+1}_{i,j+1}-\eta^{n+1}_{i,j})\Delta\mathbf{Z}^{n}_{i,j+1/2},\tag{11}$$

where $\mathbf{U}$, $\mathbf{V}$, $\Delta\mathbf{Z}$, $\mathbf{G}$ and $\mathbf{A}$ are defined as

$$\mathbf{U}^{n+1}_{i+1/2,j}=\begin{bmatrix} u^{n+1}_{i+1/2,j,M}\\ u^{n+1}_{i+1/2,j,M-1}\\ u^{n+1}_{i+1/2,j,M-2}\\ \vdots\\ u^{n+1}_{i+1/2,j,m}\end{bmatrix},\qquad \mathbf{V}^{n+1}_{i,j+1/2}=\begin{bmatrix} v^{n+1}_{i,j+1/2,M}\\ v^{n+1}_{i,j+1/2,M-1}\\ v^{n+1}_{i,j+1/2,M-2}\\ \vdots\\ v^{n+1}_{i,j+1/2,m}\end{bmatrix},\qquad \Delta\mathbf{Z}=\begin{bmatrix} \Delta z_M\\ \Delta z_{M-1}\\ \Delta z_{M-2}\\ \vdots\\ \Delta z_m\end{bmatrix},$$

$$\mathbf{G}^{n}_{i+1/2,j}=\begin{bmatrix} \Delta z_M F u^n_{i+1/2,j,M}+\Delta t\tau^w_x\\ \Delta z_{M-1} F u^n_{i+1/2,j,M-1}\\ \Delta z_{M-2} F u^n_{i+1/2,j,M-2}\\ \vdots\\ \Delta z_m F u^n_{i+1/2,j,m}\end{bmatrix},\qquad \mathbf{G}^{n}_{i,j+1/2}=\begin{bmatrix} \Delta z_M F v^n_{i,j+1/2,M}+\Delta t\tau^w_y\\ \Delta z_{M-1} F v^n_{i,j+1/2,M-1}\\ \Delta z_{M-2} F v^n_{i,j+1/2,M-2}\\ \vdots\\ \Delta z_m F v^n_{i,j+1/2,m}\end{bmatrix},$$

$$\mathbf{A}=$$

$$\begin{bmatrix} \Delta z_M+\dfrac{v_{M-1/2}\Delta t}{\Delta z_{M-1/2}} & \dfrac{-v_{M-1/2}\Delta t}{\Delta z_{M-1/2}} & & 0\\ \dfrac{-v_{M-1/2}\Delta t}{\Delta z_{M-1/2}} & \Delta z_{M-1}+\dfrac{v_{M-1/2}\Delta t}{\Delta z_{M-1/2}}+\dfrac{v_{M-3/2}\Delta t}{\Delta z_{M-3/2}} & \dfrac{-v_{M-3/2}\Delta t}{\Delta z_{M-3/2}} & \\ \vdots & \vdots & \vdots & \vdots\\ 0 & & \dfrac{-v_{m+1/2}\Delta t}{\Delta z_{m+1/2}} & \Delta z_m+\dfrac{v_{m+1/2}\Delta t}{\Delta z_{m+1/2}}+\dfrac{g\Delta t\sqrt{(u^2+v^2)}}{Cz^2}\end{bmatrix},$$

Equations (10) and (11) are linear tridiagonal systems which are coupled to the water surface elevation η^{n+1} at time t_{n+1}. In order to determine $\eta^{n+1}_{i,j}$, and for numerical stability, the new velocity field must satisfy for each i, j the finite difference analogue of the free surface equation (2),

$$\eta^{n+1}_{i,j}=\eta^n_{i,j}-\frac{\Delta t}{\Delta x}\left(\sum_{k=m}^{M}\Delta z_{i+1/2,j,k}u^{n+1}_{i+1/2,j,k}-\sum_{k=m}^{M}\Delta z_{i-1/2,j,k}u^{n+1}_{i-1/2,j,k}\right)$$
$$-\frac{\Delta t}{\Delta y}\left(\sum_{k=m}^{M}\Delta z_{i,j+1/2,k}v^{n+1}_{i,j+1/2,k}-\sum_{k=m}^{M}\Delta z_{i,j-1/2,k}v^{n+1}_{i,j-1/2,k}\right),$$

or, in matrix notation,

$$\eta^{n+1}_{i,j}=\eta^n_{i,j}-\frac{\Delta t}{\Delta x}[(\Delta\mathbf{Z}_{i+1/2,j})^{\mathrm{T}}\mathbf{U}^{n+1}_{i+1/2,j}-(\Delta\mathbf{Z}_{i-1/2,j})^{\mathrm{T}}\mathbf{U}^{n+1}_{i-1/2,j}]$$
$$-\frac{\Delta t}{\Delta y}[(\Delta\mathbf{Z}_{i,j+1/2})^{\mathrm{T}}\mathbf{V}^{n+1}_{i,j+1/2}-(\Delta\mathbf{Z}_{i,j-1/2})^{\mathrm{T}}\mathbf{V}^{n+1}_{i,j-1/2}] \qquad (12)$$

For any structure given to F, equations (10)–(12) constitute a linear system of equations with unknowns $u^{n+1}_{i+1/2,j,k}$, $v^{n+1}_{i,j+1/2,k}$ and $\eta^{n+1}_{i,j}$ over the entire computational mesh. This system has to be solved at each time step to determine recursively values of the field variables from given initial data. For computational convenience this system will first be reduced to a smaller system in which $\eta_{i,j}^{n+1}$ are the only unknowns. Specifically, formal substitution of the expressions for $\mathbf{U}_{i\pm 1/2,j}^{n+1}$ and $\mathbf{V}_{i,j\pm 1/2}^{n+1}$ from (10) and (11) into (12) yields

$$\begin{aligned}\eta_{i,j}^{n+1} &- g\frac{\Delta t^2}{\Delta x^2}\{[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\Delta\mathbf{Z}]_{i+1/2,j}^{n}(\eta_{i+1,j}^{n+1}-\eta_{i,j}^{n+1})-[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\Delta\mathbf{Z}]_{i-1/2,j}^{n}(\eta_{i,j}^{n+1}-\eta_{i-1,j}^{n+1})\}\\ &- g\frac{\Delta t^2}{\Delta y^2}\{[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\Delta\mathbf{Z}]_{i,j+1/2}^{n}(\eta_{i,j+1}^{n+1}-\eta_{i,j}^{n+1})-[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\Delta\mathbf{Z}]_{i,j-1/2}^{n}(\eta_{i,j}^{n+1}-\eta_{i,j-1}^{n+1})\}\\ &= \eta_{i,j}^{n}-\frac{\Delta t}{\Delta x}\{[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i+1/2,j}^{n}-[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i-1/2,j}^{n}\}-\frac{\Delta t}{\Delta y}\{[\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i,j+1/2}^{n}\\ &- [(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i,j-1/2}^{n}\}. \end{aligned} \tag{13}$$

Since $\mathbf{A}$ is positive definite, $\mathbf{A}^{-1}$ is also positive definite and therefore $(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\Delta\mathbf{Z}$ is a non-negative number. Hence equations (13) constitute a linear five-diagonal system of equations for $\eta_{i,j}^{n+1}$ which is symmetric and strictly diagonally dominant with positive elements on the main diagonal and negative ones elsewhere. Thus the system is positive definite and has a unique solution. In practice, this five-diagonal system can be solved very efficiently by a conjugate gradient method. Once the new free surface location has been determined, equations (10) and (11) are readily applicable to yield the new velocities u and v at time t_{n+1}.

Finally, by discretizing the continuity equation in system (1), the vertical component of the velocity w at the new time level is

$$\begin{aligned} w_{i,j,k+1/2}^{n+1} = w_{i,j,k-1/2}^{n+1} &- \frac{\Delta z_{i+1/2,j,k}^{n}u_{i+1/2,j,k}^{n+1}-\Delta z_{i-1/2,j,k}^{n}u_{i-1/2,j,k}^{n+1}}{\Delta x}\\ &- \frac{\Delta z_{i,j+1/2,k}^{n}v_{i,j+1/2,k}^{n+1}-\Delta z_{i,j-1/2,k}^{n}v_{i,j-1/2,k}^{n+1}}{\Delta y}, \quad k=m, m+1, \ldots, M, \end{aligned} \tag{14}$$

where the no-flux condition across the bottom boundary is assured by taking $w_{i,j,m-1/2}^{n+1}=0$.

4. FREE SURFACE CALCULATION BY CONJUGATE GRADIENT METHOD

Since a substantial part of the computing time will be spent on solving the linear system of equations (13), the present section will be devoted to deriving a suitable form of the conjugate gradient method which is fast and requires a minimum amount of computer memory.[27] First, equation (13) is rewritten in the more compact form

$$d_{i,j}^{n}\eta_{i,j}^{n+1}-s_{i+1/2,j}^{n}\eta_{i+1,j}^{n+1}-s_{i-1/2,j}^{n}\eta_{i-1,j}^{n+1}-s_{i,j+1/2}^{n}\eta_{i,j+1}^{n+1}-s_{i,j-1/2}^{n}\eta_{i,j-1}^{n+1}=q_{i,j}^{n}, \tag{15}$$

where

$$s_{i\pm 1/2,j}^{n}=g\frac{\Delta t^2}{\Delta x^2}[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\Delta\mathbf{Z}]_{i\pm 1/2,j}^{n}, \qquad s_{i,j\pm 1/2}^{n}=g\frac{\Delta t^2}{\Delta y^2}[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\Delta\mathbf{Z}]_{i,j\pm 1/2}^{n},$$

$$d_{i,j}^{n}=1+s_{i+1/2,j}^{n}+s_{i-1/2,j}^{n}+s_{i,j+1/2}^{n}+s_{i,j-1/2}^{n},$$

$$\begin{aligned} q_{i,j}^{n}=\eta_{i,j}^{n}&-\frac{\Delta t}{\Delta x}\{[\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i+1/2,j}^{n}-[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i-1/2,j}^{n}\}\\ &-\frac{\Delta t}{\Delta y}\{[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i,j+1/2}^{n}-[(\Delta\mathbf{Z})^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}]_{i,j-1/2}^{n}\}. \end{aligned}$$

Equation (15) can also be written in the normalized form

$$\sqrt{(d^n_{i,j})}\eta^{n+1}_{i,j}-\frac{s^n_{i+1/2,j}}{\sqrt{(d^n_{i,j}d^n_{i+1,j})}}\sqrt{(d^n_{i+1,j})}\eta^{n+1}_{i+1,j}-\frac{s^n_{i-1/2,j}}{\sqrt{(d^n_{i,j}d^n_{i-1,j})}}\sqrt{(d^n_{i-1,j})}\eta^{n+1}_{i-1,j}$$
$$-\frac{s^n_{i,j+1/2}}{\sqrt{(d^n_{i,j}d^n_{i,j+1})}}\sqrt{(d^n_{i,j+1})}\eta^{n+1}_{i,j+1}-\frac{s^n_{i,j-1/2}}{\sqrt{(d^n_{i,j}d^n_{i,j-1})}}\sqrt{(d^n_{i,j-1})}\eta^{n+1}_{i,j-1}=\frac{q^n_{i,j}}{\sqrt{(d^n_{i,j})}}, \tag{16}$$

which, by changing the variable $\sqrt{(d^n_{i,j})}\eta^{n+1}_{i,j}$ to $e_{i,j}$, is equivalent to

$$e_{ij}-a_{i+1/2,j}e_{i+1,j}-a_{i-1/2,j}e_{i-1,j}-a_{i,j+1/2}e_{i,j+1}-a_{i,j-1/2}e_{i,j-1}=b_{i,j}, \tag{17}$$

where

$$a_{i\pm 1/2,j}=\frac{s^n_{i\pm 1/2,j}}{\sqrt{(d^n_{i,j}d^n_{i\pm 1,j})}},\qquad a_{i,j\pm 1/2}=\frac{s^n_{i,j\pm 1/2}}{\sqrt{(d^n_{i,j}d^n_{i,j\pm 1})}},\qquad b_{i,j}=\frac{q^n_{i,j}}{\sqrt{(d^n_{i,j})}}.$$

Note that for notational simplicity the superscript n has been omitted; however, in equation (17) the coefficients $a_{i\pm 1/2,j}$, $a_{i,j\pm 1/2}$ and $b_{i,j}$ and the unknowns $e_{i,j}$ depend on the time step. Note also that the coefficients $a_{i\pm 1/2,j}$ and $a_{i,j\pm 1/2}$ in (17) are non-negative and their sum is strictly less than unity. Thus the system formed by these equations is normalized, symmetric and positive definite.

The conjugate gradient algorithm to solve the system of equations (17) takes the following steps.

(a) Guess $e^{(0)}_{i,j}$.

(b) Set $p^{(0)}_{i,j}=r^{(0)}_{i,j}=e^{(0)}_{i,j}-a_{i+1/2,j}e^{(0)}_{i+1,j}-a_{i-1/2,j}e^{(0)}_{i-1,j}-a_{i,j+1/2}e^{(0)}_{i,j+1}-a_{i,j-1/2}e^{(0)}_{i,j-1}-b_{i,j}$.

(c) Then for $k=0, 1, 2, \ldots$ and until $(\mathbf{r}^{(k)}, \mathbf{r}^{(k)})<\varepsilon$, calculate

$$e^{(k+1)}_{i,j}=e^{(k)}_{i,j}-\alpha^{(k)}p^{(k)}_{i,j},\qquad \text{where}\quad \alpha^{(k)}=\frac{(\mathbf{r}^{(k)},\mathbf{r}^{(k)})}{(\mathbf{p}^{(k)},\mathbf{Mp}^{(k)})}, \tag{18}$$

$$r^{(k+1)}_{i,j}=r^{(k)}_{i,j}-\alpha^{(k)}(Mp^{(k)})_{i,j}, \tag{19}$$

$$p^{(k+1)}_{i,j}=r^{(k+1)}_{i,j}+\beta^{(k)}p^{(k)}_{i,j},\qquad \text{where}\quad \beta^{(k)}=\frac{(\mathbf{r}^{(k+1)},\mathbf{r}^{(k+1)})}{(\mathbf{r}^{(k)},\mathbf{r}^{(k)})}. \tag{20}$$

In equation (19) each element of the vector **Mp** is simply given by

$$(Mp^{(k)})_{i,j}=p^{(k)}_{i,j}-a_{i+1/2,j}p^{(k)}_{i+1,j}-a_{i-1/2,j}p^{(k)}_{i-1,j}-a_{i,j+1/2}p^{(k)}_{i,j+1}-a_{i,j-1/2}p^{(k)}_{i,j-1}. \tag{21}$$

Thus at each iteration the essential calculations consist of a matrix–vector multiplication **Mp** as specified in equation (21), two scalar products between vectors, namely $(\mathbf{r}, \mathbf{r})$ and $(\mathbf{p}, \mathbf{Mp})$, and three sums between vectors, namely $\mathbf{e}-\alpha\mathbf{p}$, $\mathbf{r}-\alpha\mathbf{Mp}$ and $\mathbf{r}+\beta\mathbf{p}$. Obviously, all these operations are relatively simple and fully vectorizable.

5. FLOODING AND DRYING OF COMPUTATIONAL CELLS

Once the free surface (and hence the new water velocity) has been computed throughout the computational domain, before proceeding to the next time step, some of the vertical grid spacings $\Delta z_{i+1/2,j,k}$ and $\Delta z_{i,j+1/2,k}$ have to be updated to account for the new free surface location. The new total depth $H^{n+1}_{i+1/2,j}$ and $H^{n+1}_{i,j+1/2}$ at the u and v horizontal locations have to be updated. Thus, since the bathymetry $h_{i+1/2,j}$ and $h_{i,j+1/2}$ is specified at the u and v horizontal points and since a negative value for the total depth H is physically meaningless, the discrete total depths

$H^{n+1}_{i+1/2,j}$ and $H^{n+1}_{i,j+1/2}$ are defined as

$$H^{n+1}_{i+1/2,j}=\max(0,\, h_{i+1/2,j}+\eta^{n+1}_{i,j},\, h_{i+1/2,j}+\eta^{n+1}_{i+1,j}), \tag{22}$$

$$H^{n+1}_{i,j+1/2}=\max(0,\, h_{i,j+1/2}+\eta^{n+1}_{i,j},\, h_{i,j+1/2}+\eta^{n+1}_{i,j+1}). \tag{23}$$

A resulting zero value of the total depth $H=h(x, y)+\eta(x, y, t)$ simply means a dry point which may be flooded when the total water depth H becomes positive. If the total depth is positive, then the side is a wet side and along that side some of the vertical increments Δz will be non-zero. If the total water depth is zero, then all the vertical increments Δz will be zero at the corresponding side. Thus the regions representing low land can be initialized by a small negative value of h and the regions of permanent dry land can be initialized by a large negative value of h, so that the total depth will never become positive even at high tides. Moreover, when H is zero, the respective friction factor will be assumed to be infinity and, accordingly, the corresponding velocity u or v across the side of the cell is forced to vanish. It is important to point out that the resulting finite difference equation for the water surface elevation, equation (12), correctly accounts for positive and zero values of the total depth on each side of a computational stencil. The treatment of flooding and drying is borne out naturally without special treatment and guarantees mass conservation while accounting for the flooding and drying of tidal mud-flats. The occurrence of a zero value for the total depth H on one side of a cell implies zero velocity or zero mass flux until, at a later time, H becomes positive. Of course, a cell is considered a dry cell only if the total water depths at all sides are zero. Accordingly, in a dry cell equation (12) is reduced to $\eta^{n+1}_{i,j}=\eta^{n}_{i,j}$, i.e. there is no variation of the water surface elevation in a dry cell. The η-value for a dry cell will be solved in the general procedure; an artificial η-value for a land point is never required by equation (12). Similarly, on a dry side of a cell equations (10) or (11) will be replaced by $\mathbf{U}^{n+1}_{i+1/2,j}=\mathbf{0}$ or $\mathbf{V}^{n+1}_{i,j+1/2}=\mathbf{0}$. Hence, when the flooding and drying of cells takes place, the application of the present algorithm in those cells produces the same finite difference equations, equation (12). This simplifies the computer algorithm in that equation (12) can be applied to all points throughout the domain, resulting in an algorithm that can be vectorized for efficient computations. The presence of islands and other permanently dry areas as well as tidal flats will be accounted for appropriately and automatically. The boundary shorelines, which are varying with time, are defined by the condition of no mass flux. This condition is automatically satisfied as discussed above without any special treatment.

6. EULERIAN–LAGRANGIAN DISCRETIZATION OF CONVECTIVE AND VISCOUS TERMS

One of the major difficulties in the numerical treatment of the shallow water equations arises from the discretization of the convective and viscous terms. Consider then the following convection–diffusion equation in three space dimensions:

$$\frac{\partial c}{\partial t}+u\frac{\partial c}{\partial x}+v\frac{\partial c}{\partial y}+w\frac{\partial c}{\partial z}=\mu\left(\frac{\partial^2 c}{\partial x^2}+\frac{\partial^2 c}{\partial y^2}\right)+\frac{\partial}{\partial z}\left(v\frac{\partial c}{\partial z}\right), \tag{24}$$

where μ and v are non-negative diffusion coefficients and, for the time being, the convective velocities u, v and w are assumed to be constants.

Equation (24) can be solved numerically in a variety of ways. A convenient semi-implicit finite difference method, whose stability does not depend upon the vertical eddy diffusivity, is obtained by discretizing the convective terms by explicit upwind finite differences, the horizontal eddy diffusivity by explicit central differences and the vertical eddy diffusivity term by an implicit finite difference. For non-negative u, v and w the resulting finite difference equation is

$$\frac{c^{n+1}_{i,j,k}-c^{n}_{i,j,k}}{\Delta t}+u\frac{c^{n}_{i,j,k}-c^{n}_{i-1,j,k}}{\Delta x}+v\frac{c^{n}_{i,j,k}-c^{n}_{i,j-1,k}}{\Delta y}+w\frac{c^{n}_{i,j,k}-c^{n}_{i,j,k-1}}{\Delta z}$$

$$=\mu\left(\frac{c^{n}_{i+1,j,k}-2c^{n}_{i,j,k}+c^{n}_{i-1,j,k}}{\Delta x^2}+\frac{c^{n}_{i,j+1,k}-2c^{n}_{i,j,k}+c^{n}_{i,j-1,k}}{\Delta y^2}\right)$$

$$+\frac{v_{k+1/2}\dfrac{c^{n+1}_{i,j,k+1}-c^{n+1}_{i,j,k}}{\Delta z_{i,j,k+1/2}}-v_{k-1/2}\dfrac{c^{n+1}_{i,j,k}-c^{n+1}_{i,j,k-1}}{\Delta z_{i,j,k-1/2}}}{\Delta z_{i,j,k}}. \tag{25}$$

For every i and j this method requires the solution of a symmetric, positive definite, tridiagonal system. The necessary and sufficient stability condition of scheme (25) is

$$\Delta t \leqslant \left[\frac{|u|}{\Delta x}+\frac{|v|}{\Delta y}+\frac{|w|}{\Delta z}+2\mu\left(\frac{1}{\Delta x^2}+\frac{1}{\Delta y^2}\right)\right]^{-1} \tag{26}$$

In convection-dominated problems the stability condition (26) is not very restrictive. This method, however, is only first-order-accurate in space and the truncation error is in the form of a diffusion term. This artificial viscosity is directionally dependent. Hence, in convection-dominated problems, not only the artificial viscosity will prevail over the physical viscosity, but drastically different numerical predictions can be obtained simply as a result of different spatial orientations of the computational grid.[28,29]

In order to improve the stability and accuracy of an explicit finite difference method, consider again equation (24) in the Lagrangian form

$$\frac{\mathrm{d}c}{\mathrm{d}t}=\mu\left(\frac{\partial^2 c}{\partial x^2}+\frac{\partial^2 c}{\partial y^2}\right)+\frac{\partial}{\partial z}\left(v\frac{\partial c}{\partial z}\right), \tag{27}$$

where the substantial derivative $\mathrm{d}/\mathrm{d}t$ indicates that the time rate of change is calculated along the streak line defined by

$$\frac{\mathrm{d}x}{\mathrm{d}t}=u, \qquad \frac{\mathrm{d}y}{\mathrm{d}t}=v, \qquad \frac{\mathrm{d}z}{\mathrm{d}t}=w. \tag{28}$$

A natural semi-implicit discretization of equation (27) is simply given by

$$\frac{c^{n+1}_{i,j,k}-c^{n}_{i-a,j-b,k-d}}{\Delta t}=\frac{v_{k+1/2}\dfrac{c^{n+1}_{i,j,k+1}-c^{n+1}_{i,j,k}}{\Delta z_{i,j,k+1/2}}-v_{k-1/2}\dfrac{c^{n+1}_{i,j,k}-c^{n+1}_{i,j,k-1}}{\Delta z_{i,j,k-1/2}}}{\Delta z_{i,j,k}}$$

$$+\mu\left(\frac{c^{n}_{i-a+1,j-b,k-d}-2c^{n}_{i-a,j-b,k-d}+c^{n}_{i-a-1,j-b,k-d}}{\Delta x^2}+\frac{c^{n}_{i-a,j-b+1,k-d}-2c^{n}_{i-a,j-b,k-d}+c^{n}_{i-a,j-b-1,k-d}}{\Delta y^2}\right), \tag{29}$$

where $a=u\Delta t/\Delta x$, $b=v\Delta t/\Delta y$ and $d=w\Delta t/\Delta z$ are the grid Courant numbers.

It is important to interpret the physical significance of (29). The values of c at and around (i,j,k) at time t_{n+1} are related to the c-values at and around $(i-a,j-b,k-d)$ at time t_n. Moreover, $(i-a,j-b,k-d)$ denotes a point on the same streak line which passes through (i,j,k) at time t_{n+1}. Thus (29) is not only a simple algorithm but also accounts correctly for both convection and diffusion. In general, however, a, b and d are not integers; therefore $(i-a,j-b,k-d)$ is not a grid point and an interpolation formula must be used to define

$c^n_{i-a,j-b,k-d}$. The accuracy, stability, numerical diffusion and spurious oscillations of (29) depend on the interpolation formula chosen. The simplest interpolation that can be taken for calculating $c^n_{i-a,j-b,k-d}$ is the trilinear interpolation over the eight surrounding mesh points. For positive a, b and d, let l, m and n be the integer parts of a, b and d respectively and p, q and r be their corresponding decimal parts, so that $a=l+p$, $b=m+q$ and $d=n+r$. Then $c^n_{i-a,j-b,k-d}$ is approximated by

$$\begin{aligned} c^n_{i-a,j-b,k-d} = {} & (1-r)\{(1-p)[(1-q)c^n_{i-l,j-m,k-n} + qc^n_{i-l,j-m-1,k-n}] \\ & + p[(1-q)c^n_{i-l-1,j-m,k-n} + qc^n_{i-l-1,j-m-1,k-n}]\} \\ & + r\{(1-p)[(1-q)c^n_{i-l,j-m,k-n-1} + qc^n_{i-l,j-m-1,k-n-1}] \\ & + p[(1-q)c^n_{i-l-1,j-m,k-n-1} + qc^n_{i-l-1,j-m-1,k-n-1}]\}. \end{aligned} \tag{30}$$

It can be shown[28] that when a bilinear interpolation (30) is used, the Eulerian–Lagrangian scheme (29) is free from spurious oscillations. Moreover, the artificial diffusion, which can be regarded as the interpolation error, is reduced when compared with the artificial diffusion induced by the upwind method. Further reduction can be obtained by increasing a, b and d, i.e. by reducing Δx, Δy and Δz. Complete elimination of the numerical diffusion can be achieved by using a higher-order interpolation formula, but the resulting method may introduce some spurious oscillations. Applications of this scheme to problems with large vertical diffusion v or small vertical spacings Δz have suggested the use of an implicit discretization only for the vertical diffusion term. Indeed, it can be shown[30] that the stability condition for the scheme (29) is simply given by

$$\Delta t \leqslant \left[2\mu \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right) \right]^{-1}, \tag{31}$$

which is much less restrictive than (26). Clearly, when $\mu=0$, this scheme becomes unconditionally stable.

The Eulerian–Lagrangian method described above is also applicable to the case when equation (24) is non-linear. In this case the determination of a, b and d requires the integration of equations (28), in which the right-hand sides are known only at time level t_n. Therefore u, v and w are assumed to be invariant over a time step and equations (28) will be integrated numerically backwards from time level t_{n+1} to t_n by using, for instance, the Euler method.[7] The streak lines, which in general are not straight lines, are better approximated. This integration process is relatively fast, especially if performed on a vector machine.

Use of the Eulerian–Lagrangian method to discretize the convective and viscous terms in the momentum equations of system (1) now appears to be quite straightforward. Specifically, the finite difference operator F in (6) and (7) can be defined as

$$\begin{aligned} Fu^{n+1}_{i+1/2,j,k} = {} & u^n_{i+1/2-a,j-b,k-d} \\ & + \mu\Delta t \left(\frac{u^n_{i+1/2-a+1,j-b,k-d} - 2u^n_{i+1/2-a,j-b,k-d} + u^n_{i+1/2-a-1,j-b,k-d}}{\Delta x^2} \right. \\ & \left. + \frac{u^n_{i+1/2-a,j-b+1,k-d} - 2u^n_{i+1/2-a,j-b,k-d} + u^n_{i+1/2-a,j-b-1,k-d}}{\Delta y^2} \right) \\ & + f\Delta t\, v^n_{i+1/2-a,j-b,k-d}, \end{aligned} \tag{32}$$

$$\begin{aligned} Fv^{n+1}_{i,j+1/2,k} &= v^{n}_{i-a,j+1/2-b,k-d} \\ &+\mu\Delta t\left(\frac{v^{n}_{i-a+1,j+1/2-b,k-d}-2v^{n}_{i-a,j+1/2-b,k-d}+v^{n}_{i-a-1,j+1/2-b,k-d}}{\Delta x^{2}}\right. \\ &\left.+\frac{v^{n}_{i-a,j+1/2-b+1,k-d}-2v^{n}_{i-a,j+1/2-b,k-d}+v^{n}_{i-a,j+1/2-b-1,k-d}}{\Delta y^{2}}\right) \\ &-f\Delta t\, u^{n}_{i-a,j+1/2-b,k-d}. \end{aligned} \tag{33}$$

The inequality (31) then constitutes the stability condition for the semi-implicit flow algorithm.

7. A PARTICULAR CASE: THE TWO-DIMENSIONAL MODEL

Interestingly enough, if the vertical spacing Δz is taken to be large enough so that both the bottom and the free surface fall within one vertical layer, then this algorithm reduces to a two-dimensional semi-implicit numerical method as described by Casulli.[7] In this case the third index can be omitted from the notation and $\Delta\mathbf{Z}_{i+1/2,j}$ and $\Delta\mathbf{Z}_{i,j+1/2}$ reduce to $H^{n}_{i+1/2,j}$ and $H^{n}_{i,j+1/2}$, respectively. Since $m=M=1$, by substituting the boundary conditions (8) and (9) in equations (6) and (7), the following two-dimensional discretized momentum equations result:

$$\begin{aligned} U^{n+1}_{i+1/2,j} &= FU^{n}_{i+1/2,j}-g\frac{\Delta t}{\Delta x}(\eta^{n+1}_{i+1,j}-\eta^{n+1}_{i,j})+\Delta t\frac{\tau^{w}_{x}}{H^{n}_{i+1/2,j}} \\ &-\Delta t\frac{g\sqrt{[(U^{n}_{i+1/2,j})^{2}+(V^{n}_{i+1/2,j})^{2}]}}{Cz^{2}H^{n}_{i+1/2,j}}U^{n+1}_{i+1/2,j}, \end{aligned} \tag{34}$$

$$\begin{aligned} V^{n+1}_{i,j+1/2} &= FV^{n}_{i,j+1/2}-g\frac{\Delta t}{\Delta y}(\eta^{n+1}_{i,j+1}-\eta^{n+1}_{i,j})+\Delta t\frac{\tau^{w}_{y}}{H^{n}_{i,j+1/2}} \\ &-\Delta t\frac{g\sqrt{[(U^{n}_{i,j+1/2})^{2}+(V^{n}_{i,j+1/2})^{2}]}}{Cz^{2}H^{n}_{i,j+1/2}}V^{n+1}_{i,j+1/2}, \end{aligned} \tag{35}$$

where F is again an explicit, non-linear finite difference operator corresponding to the spatial discretization of the convective terms, the horizontal eddy viscosity terms and the Coriolis terms. Again, equations (34) and (35) can be written in the form (10) and (11), where the matrix $\mathbf{A}$ and the vectors $\mathbf{G}$ and $\Delta\mathbf{Z}$ now contain only one element. Specifically,

$$\mathbf{A}^{n}_{i+1/2,j}=H^{n}_{i+1/2,j}+\Delta t\frac{g\sqrt{[(U^{n}_{i+1/2,j})^{2}+(V^{n}_{i+1/2,j})^{2}]}}{Cz^{2}},$$

$$\mathbf{A}^{n}_{i,j+1/2}=H^{n}_{i,j+1/2}+\Delta t\frac{g\sqrt{[(U^{n}_{i,j+1/2})^{2}+(V^{n}_{i,j+1/2})^{2}]}}{Cz^{2}},$$

$$\mathbf{G}^{n}_{i+1/2,j}=H^{n}_{i+1/2,j}Fu^{n}_{i+1/2,j}+\Delta t\tau^{w}_{x},$$

$$\mathbf{G}^{n}_{i,j+1/2}=H^{n}_{i,j+1/2}Fv^{n}_{i,j+1/2}+\Delta t\tau^{w}_{y},$$

$$\Delta\mathbf{Z}^{n}_{i+1/2,j}=H^{n}_{i+1/2,j}$$

$$\Delta\mathbf{Z}^{n}_{i+1/2,j}=H^{n}_{i+1/2,j}.$$

Thus the finite difference free surface equation (12) in this case then reduces to

$$\eta^{n+1}_{i,j}=\eta^{n}_{i,j}-\frac{\Delta t}{\Delta x}(H^{n}_{i+1/2,j}U^{n+1}_{i+1/2,j}-H^{n}_{i-1/2,j}U^{n+1}_{i-1/2,j})$$

$$-\frac{\Delta t}{\Delta y}(H^{n}_{i,j+1/2}V^{n+1}_{i,j+1/2}-H^{n}_{i,j-1/2}V^{n+1}_{i,j-1/2}). \qquad (36)$$

Equations (34)–(36) can also be regarded as obtained directly from a semi-implicit discretization of two-dimensional shallow water equations (5). Formal substitution of the expressions for $U^{n+1}_{i\pm 1/2,j}$ and $V^{n+1}_{i,j\pm 1/2}$ from (34) and (35) into (36) yields again an equation of the form (13). Thus the corresponding linear system is symmetric and positive definite. In practice, this five-diagonal system can be solved very efficiently by a conjugate gradient method as described above. Inequality (31) can be shown to be the stability condition which applies when a Eulerian–Lagrangian discretization is used to discretize the convective terms.[1,7] The corresponding tidal, residual, inter-tidal mud-flat (TRIM) algorithm has been presented and applied to several estuaries and tidal embayments.[11,31]

The fact that a consistent two-dimensional shallow water model can be derived from the three-dimensional model as a particular case is a very important feature of this formulation. This property of the algorithm leads to a computer code that can be used for both three-dimensional problems as well as two-dimensional problems as a particular case. More importantly, when the three-dimensional model is applied to a typical coastal plain tidal embayment characterized by deep channels connected to large and flat shallow areas, a great saving in computing time is achieved because the deep channels are correctly represented in three dimensions while the flat shallow areas are represented only in two dimensions without any special treatment. Since the shallow areas of tidal embayments are almost always well mixed, a two-dimensional representation is appropriate.

8. APPLICATIONS

In the following, two distinctly different tidal flow simulations are reported to demonstrate properties of this model. Application of the model in San Francisco Bay, California is to show the robustness of the model. The second example is the extremely complex flows in the Lagoon of Venice, Italy. The model results reported here are mainly given in support of the mathematical claims for the present model. Detailed model calibration and verification for each of these studies will be reported separately.

8.1. San Francisco Bay, California

San Francisco Bay is a complex estuary consisting of interconnected embayments, sloughs, marshes and channels (Plate 1). Among the many factors that affect the flow properties in the Bay, the water depth distribution is one of the most important factors controlling the spatial variability of both the magnitude and direction of the tidal current.[32,33] The tides entering the Bay through a narrow opening at the Golden Gate have a range of roughly 2 m composed primarily of the M_2 and K_1 tides. Within Central Bay the tides bifurcate and propagate concurrently into South Bay and the northern reach, which includes San Pablo Bay, Carquinez Strait and Suisun Bay. Central Bay is not only geometrically complex, but the variations in depth are quite large. The deepest region in Central Bay is located near the Golden Gate, where a depth

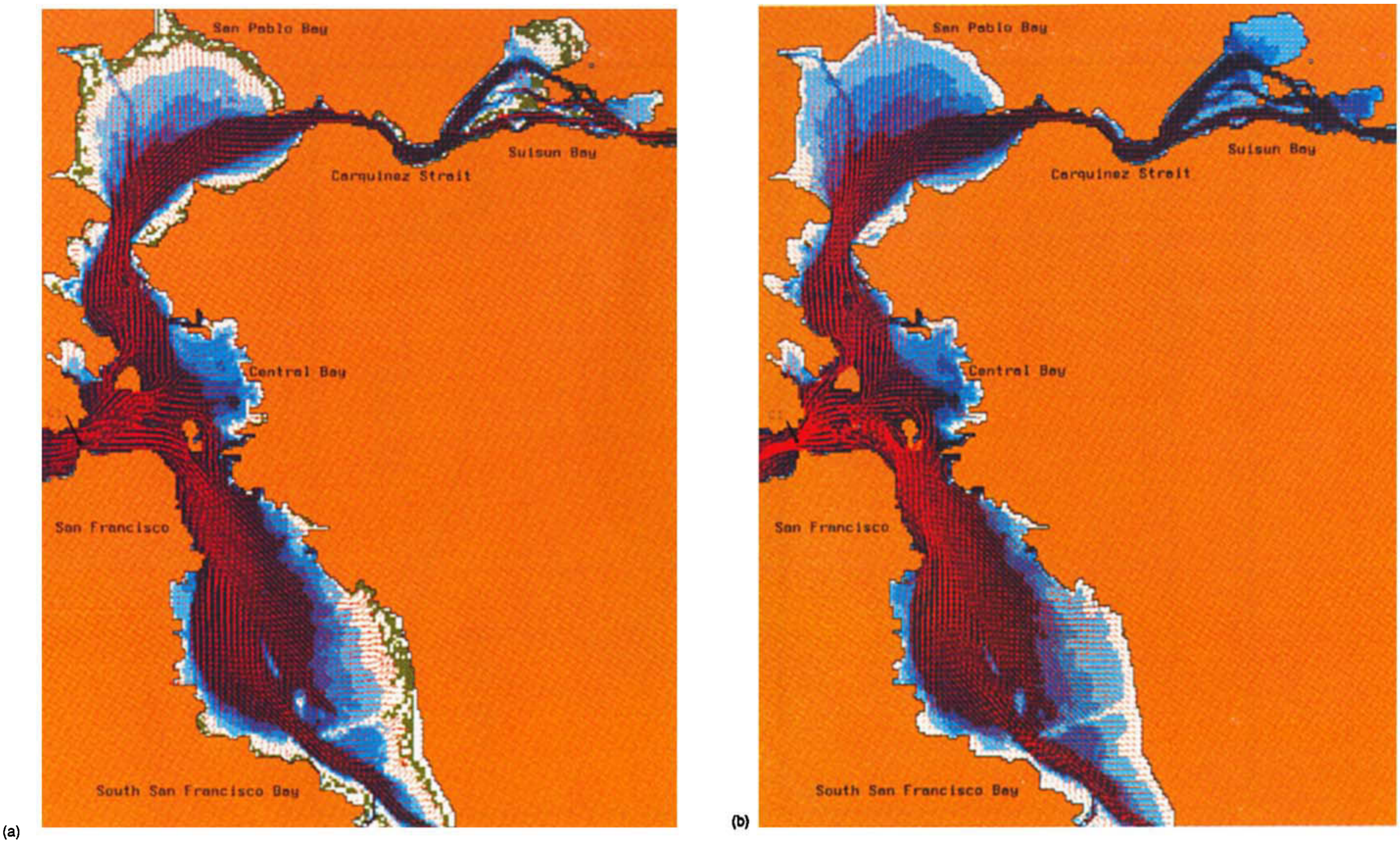


Plate 1. San Francisco Bay discretized with a 108 × 144 finite difference mesh with $\Delta x = \Delta y = 500$ m. (a) Water depth and velocity distributions 1 h after flooding at the Golden Gate. Some tidal mud-flats in South Bay, San Pablo Bay and Suisin Bay are exposed. (b) The tidal mud-flats in South Bay, San Pablo Bay and Suisin Bay are submerged 1 h after ebb

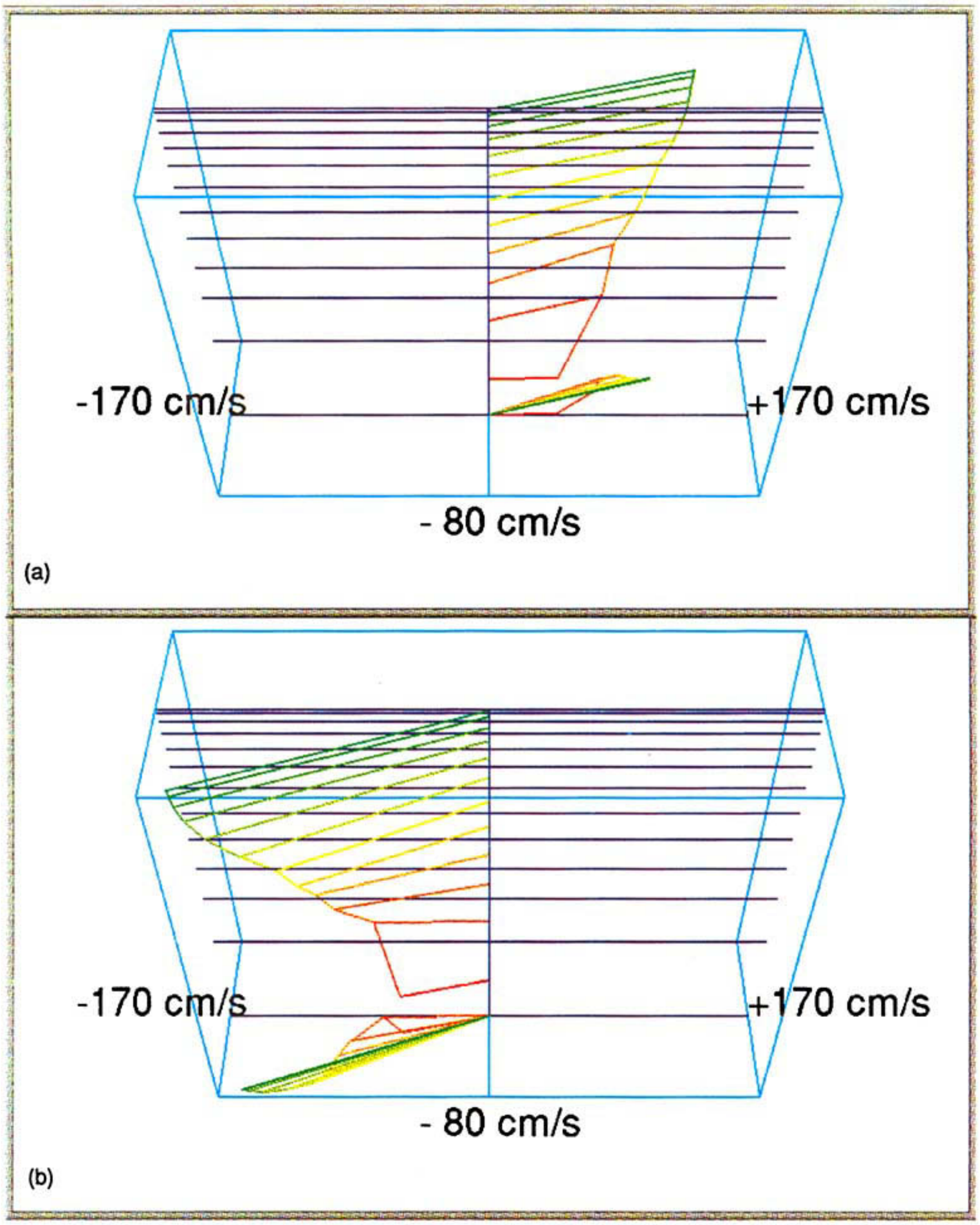


Plate 2. Vertical distribution of tidal current at C1 (a) for typical flooding and (b) for typical ebbing. Shown in the bottom panel is the projection of all velocity vectors indicating phase or direction differences

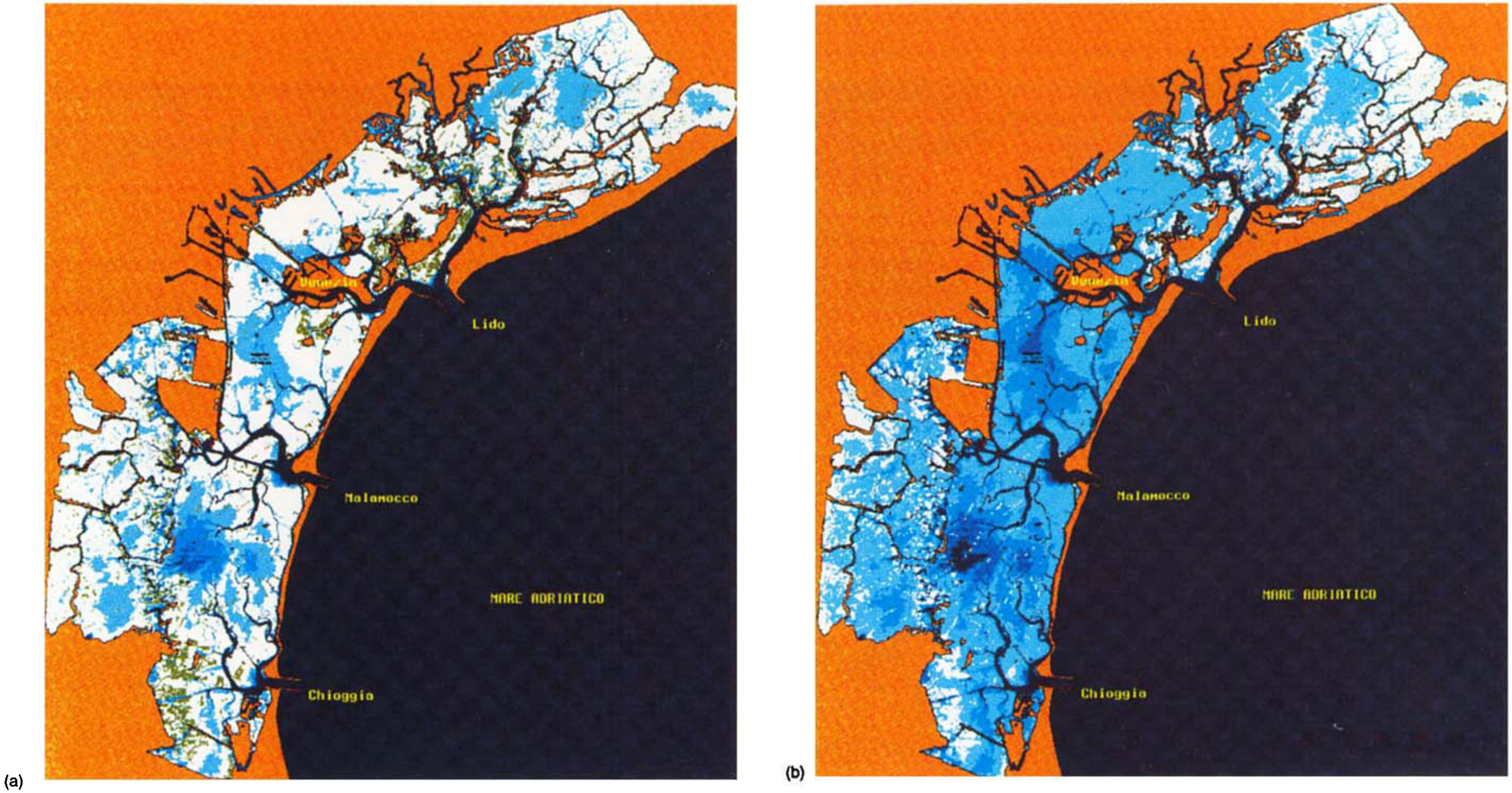


Plate 3. The Lagoon of Venice discretized with a 384 × 426 finite difference mesh with $\Delta x = \Delta y = 100$ m. There are three inlets connecting the Lagoon to the Adriatic Sea, namely Lido, Malamocco and Chioggia. The city of Venice is located near the Lido inlet. The water depth distributions show large and complex areas becoming (a) exposed at low tide and (b) submerged at high tide

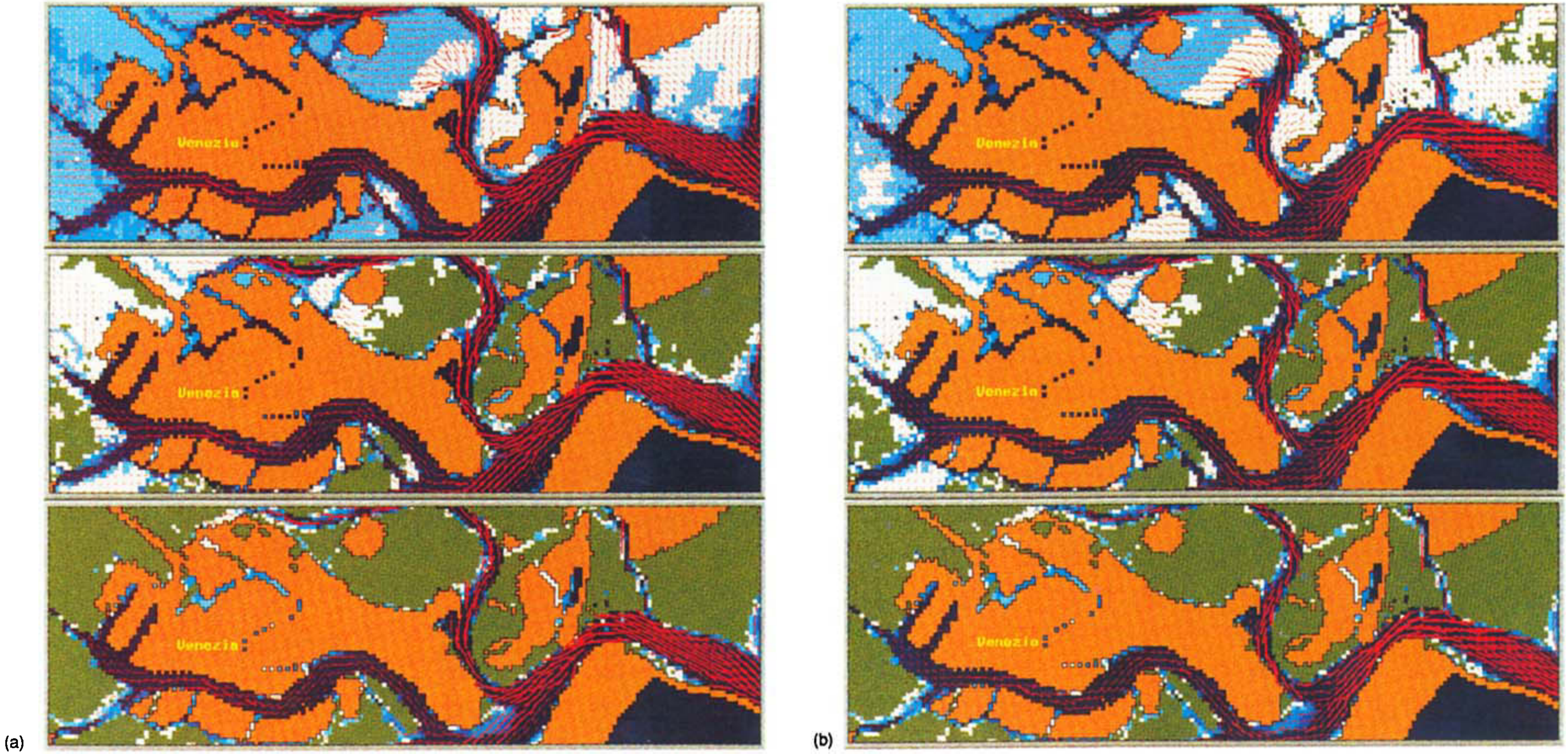


Plate 4. Tidal current distributions in top three consecutive layers at Lido inlet and around Venice during (a) maximum flood and (b) maximum ebb

close to 100 m is found. At the eastern boundary of Central Bay is a broad region of inter-tidal mud-flats. The geometry of Central Bay is further complicated by the presence of several islands.

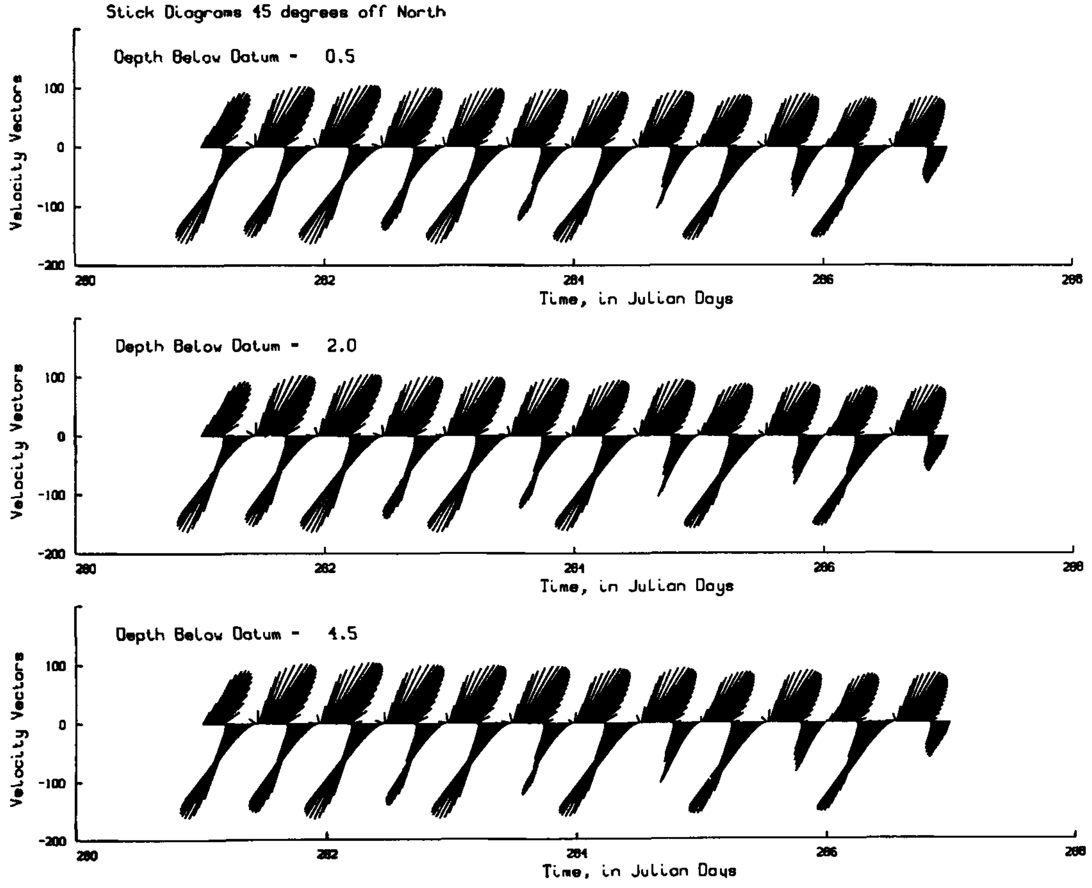
Both San Pablo Bay and South Bay are characterized by a deep channel surrounded by broad shoals. The tidal current distribution in these basins is less complex. High-intensity tidal currents are distributed within the deep channels, while lower-magnitude tidal velocities are observed over the shoal regions owing to increased friction. The spatial variability of velocity gradients over the shoals and within the deep channels are not particularly large. However, in the transition zone between the channel and the shoal the velocity gradients can be quite large. Carquinez Strait is a deep channel with a relatively simple geometry which connects San Pablo Bay to its west and Suisun Bay to its east. Suisun Bay is a shallow and complex basin consisting of several interconnected channels and shoals. For a discussion of the tides and tidal circulation in San Francisco Bay see Reference 33.

The three-dimensional tidal circulation in San Francisco Bay is simulated using the present model with a horizontal finite difference mesh of $\Delta x = \Delta y = 500$ m. At open boundaries, 16 tidal harmonic constituents are used to reproduce the tides as boundary conditions. Different numbers of vertical layers and variable layer depths have been tested. The numerical solutions have been achieved using an integration time step $\Delta t = 15$ min. In this application the maximum grid Courant number based on celerity exceeds 50, with the average grid Courant number being about 20. For a simulation of 6 days (144 h) the model with one vertical layer (i.e. the depth-averaged two-dimensional model) requires 47 CPU seconds on the Cray Y-MP8/432. Using the same horizontal discretization and integration time step, the three-dimensional model with two vertical layers requires 59 CPU seconds and with 12 vertical layers requires only 120 CPU seconds to complete the simulation. The computations have exceeded 117 Mflops (million floating point operations per second) on the Cray Y-MP supercomputer, which is an indication that the model code is highly vectorized. Separations between consecutive vertical layers in the 12-layer model are set at 1, 3, 6, 10, 15, 21, 28, 36, 45, 55 and 70 m below mean sea level. The vicinity of the maximum water depth of 98 m, just east of the Golden Gate, is the only area where all 12 layers are needed. The typical water depth and tidal current distributions in the top layer about 1 h after the beginning of flood at the Golden Gate are depicted in Plate 1(a). Similarly, the water depth and tidal current distributions 1 h after the beginning of ebb at the Golden Gate are depicted in Plate 1(b). Some shallow regions, i.e. South Bay, San Pablo Bay and Suisun Bay, are emerged at low water (Plate 1(a)) but submerged at high tide (Plate 1(b)).

The three-dimensional velocity structures recorded at C1 station near the entrance to the Bay (Plate 1) are shown in Plates 2(a) and 2(b), representing typical flooding and ebbing. The velocity vectors are not in a vertical plane. A projection of the velocity vectors onto the bottom panel is also shown in Plates 2(a) and 2(b). By examining a series of similar plots, the time evolution of flooding and ebbing cycles can be studied. These results show a noticeable phase lag in the velocity distribution in the vertical. Generally the velocity shear in the surface layers in weaker than the shear in the bottom layers. Some variations in the velocity shear in the vertical are caused by local basin topography. To exhibit the time variation of tidal velocities of a three-dimensional flow field, a stick diagram is used to depict the tidal velocity time series for each vertical layer. At this station, because the velocity vector are mostly east–west, the stick diagrams depict velocity vectors with a 45° counterclockwise rotation. Shown in Figures 2(a) and 2(b) are the stick diagrams of tidal currents at C1 for the top three layers and the bottom three layers respectively. Near the surface the velocity shear is weak and the variations between the vertical layers are generally small. Conversely, the velocity shear is considerably larger as evidenced by large variations in the velocity profile in the vertical. The tidal currents in the surface layers rotate clockwise in tidal cycles, while the bottom currents incline more towards bidirectional oscillations. Detailed model calibration and verification will be reported separately.

8.2. *Lagoon of Venice, Italy*

One of the most challenging embayments modelled with the present algorithm has been the Lagoon of Venice, Italy. The Lagoon of Venice is a very complex system whose area is about 50 km^2, consisting of several interconnected narrow channels, with a maximum width of 1 km and up to 50 m deep, encircling large and flat shallow areas. The Lagoon is connected to the Adriatic Sea through three narrow inlets, namely Lido, Malamocco and Chioggia. The city of Venice is located in the Lagoon near the Lido inlet and is threatened by at least two problems related to water circulation. During stormy seasons, storm surge of tides may inundate the ground floors of many buildings, and normally the water circulation in the Lagoon of Venice is clearly insufficient to maintain an acceptable level of water quality. The Lagoon is contaminated by the waste from the inadequate sewage system of Venice and from heavy industrial pollution sources. An accurate



(a)

Figure 2. Stick diagrams for tidal current in vertical layers. The velocities are plotted with a 45° counterclockwise rotation. (a) Stick diagrams for tidal currents in top three layers. (b) Stick diagrams for bottom three layers. The mean depth of each layer is indicated

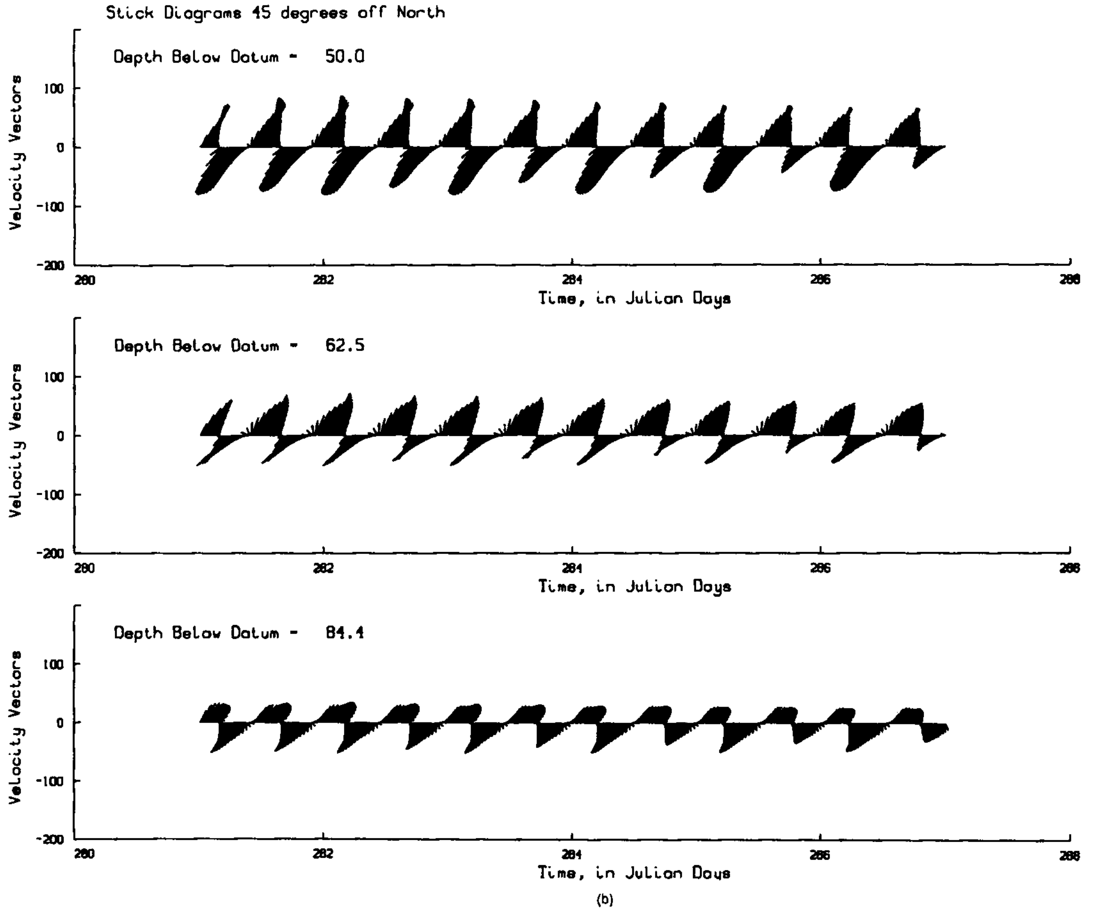


Figure 2. (Continued)

numerical model is then needed to forecast the hydrodynamic and water quality effect of any new development in the Lagoon. For this purpose a pioneer numerical study was published by Volpi and Sguazzaro.[34] A two-dimensional finite element model has recently been presented by Umgiesser *et al.*[35] Considerable portions of the Lagoon of Venice consist of areas which at low tides are emerged and at high tides are flooded (Plate 3). Therefore the capability of the present model to properly treat the flooding and drying of tidal flats becomes an essential issue. The tidal amplitudes in the Adriatic Sea are about 0·5 m. Tides propagate from the Adriatic Sea into the Lagoon of Venice through the three inlets. The Lagoon has been covered with a 384 × 426 finite difference mesh of equal $\Delta x = \Delta y = 100$ m. This fine computational mesh allows for an accurate description of the tree-like structure of the main channels. At the three inlets an M_2 tide of 0·5 m amplitude and 12 lunar hour period has been specified. The integration time step is chosen to be $\Delta t = 15$ min and the computations have been carried out by solving at each time step a corresponding linear, five-diagonal system of 163 584 equations. Using only one vertical layer (i.e. the two-dimensional model), the time required to simulate a tide of 12 h period is 62 CPU seconds on the Cray Y-MP8/432. Using the same horizontal discretization, the three-dimensional model with two vertical layers requires 83 CPU seconds. The three-dimensional model with 10 vertical layers requires only 134 CPU seconds on the Cray Y-MP8/432. Plates 3(a) and 3(b) show the typical water surface elevations computed at low and high tide respectively. Plate 4(a) shows a typical situation for inflow from the Lido inlet and around Venice in three consecutive layers from the top. Plate 4(b) shows a typical situation for outflow in the same region. The computed results clearly indicate how the currents follow the waterway of the deep channels and the ability of the present model to reproduce flooding and emergence of large areas in the Lagoon of Venice.

9. CONCLUSIONS

A semi-implicit finite-difference method for solving the two- and three-dimensional shallow water equations has been presented. The combination of judicious selection of terms that are finite differenced implicitly and use of a Eulerian–Lagrangian method for treating the convective terms makes this formulation fast, accurate and stable. A conservative form of the continuity equation is used; the resulting finite difference method is locally and globally mass-conserving. The mass conservation properties have led to an extremely simple and general algorithm for treating watering and dewatering of computational stencils. This solution scheme solves a set of tridiagonal systems along the vertical layers and one five-diagonal linear system defined throughout the horizontal flow field. All these systems are symmetric and positive definite. These matrix properties assure the existence and uniqueness of the numerical solution. Computationally, each tridiagonal system is solved by a direct method, while the numerical solution of the large five-diagonal system can be conveniently obtained by a conjugate gradient method. Furthermore, the structure of the solution algorithm leads to a computer code which is completely vectorizable. The overall accuracy of the numerical scheme is first-order in both space and time. A slight amount of numerical artificial viscosity is introduced which can be brought under control by reducing the spatial grid size.

To test the present algorithm, two applications representing a broad range of tidal characteristics have been presented and discussed. These applications, San Francisco Bay, California and the Lagoon of Venice, Italy, are unique in their own rights. The high computational efficiency of this method has made it possible to provide fine details of the circulation structures for San Francisco Bay and the Lagoon of Venice that previous studies were unable to obtain. San Francisco Bay and the Lagoon of Venice are also unique ecosystems which have received much research attention. This three-dimensional semi-implicit model will be used as one of the elements in interdisciplinary ecological investigations, providing the basic hydrodynamic information which is the backbone of transport processes on a tidal time scale. These studies are continuing; detailed calibrations and verifications will be reported separately. The computer code for this algorithm is fully vectorized and has been implemented on a variety of computers. Its superior computational efficiency has made it possible to simulate sufficiently fine details of the tidal circulation in embayments for time periods spanning the spring–neap tidal cycle or longer without breaking the computer budget of a research project.

ACKNOWLEDGEMENT

All model applications using supercomputers were performed on the Cray Y-MP8/432 at CINECA, Bologna, Italy.

REFERENCES

1. V. Casulli, 'Numerical simulation of shallow water flow', in G. Gambolati, A. Rinaldo, C. A. Brebbia, W. G. Gray and G. F. Pinder (eds), *Computational Methods in Surface Hydrology*, Springer, Berlin, 1990, pp. 13–22.

2. R. T. Cheng and P. E. Smith, 'A survey of three-dimensional numerical estuarine models', in M. L. Spaulding (ed.), *Estuarine and Coastal Modeling*, ASCE, New York, 1990, pp. 1–15.
3. G. S. Stelling, 'On the construction of computational methods for shallow water flow problems', *Rijkswaterstaat Communications No. 35*, The Hague, 1984.
4. P. Roache, *Computational Fluid Dynamics*, Revised printing, Hermosa, Albuquerque, NM, 1982.
5. T. J. Weare, 'Errors arising from irregular boundaries in ADI solutions of the shallow-water equations', *Int. j. numer. methods eng.*, **14**, 921–931 (1979).
6. J. P. Benque, J. A. Cunge, J. Feuillet, A. Hauguel and F. M. Holly, 'New method for tidal current computation', *J. Waterways, Port, Coastal, Ocean Div., ASCE*, **108**, 396–417 (1982).
7. V. Casulli, 'Semi-implicit finite difference methods for the two-dimensional shallow water equations', *J. Comput. Phys.*, **86**, 56–74 (1990).
8. J. O. Backhaus, 'A semi-implicit scheme for the shallow water equations for application to shelf sea modelling, *Continental Shelf Res.*, **2**, 243–254 (1983).
9. G. S. Stelling, A. K. Wiersma and J. B. T. Willemse, 'Practical aspects of accurate tidal computations', *J. Hydraul. Div., ASCE*, **112**, 802–817 (1986).
10. P. Wilders, Th. L. van Stijn, G. S. Stelling and G. A. Fokkema, 'A fully implicit splitting method for accurate tidal computations', *Int. j. numer. methods eng.*, **26**, 2707–2721 (1988).
11. R. T. Cheng and V. Casulli, 'Tidal, residual, inter-tidal mud-flat (TRIM) model, using semi-implicit Eulerian–Lagrangian method', *USGS Open-File Rep. 92-62*, 1992.
12. N. S. Heaps (ed.), *Three Dimensional Coastal Ocean Circulation Models, Coastal and Estuarine Sciences*, Vol. 4, AGU, Washington, DC, 1987.
13. A. F. Blumberg and H. J. Herring, 'Circulation modeling using orthogonal curvilinear coordinates', in J. C. J. Nihoul and B. M. Jamart (eds), *Three Dimensional Models of Marine and Estuarine Dynamics, Elsevier Oceanography Series*, Vol. 45, Elsevier, Amsterdam, 1987, pp. 203–214.
14. J. J. Leendertse, R. C. Alexander and S. K. Liu, 'A three-dimensional model for estuaries and coastal seas: Vol. I, Principles of computations', *Rep. R-1417-OWRR*, Rand Corporation, Santa Monica, CA, 1973.
15. J. J. Leendertse, 'A new approach to three-dimensional free-surface flow modeling', *Rep. R-3712-NETH/RC*, Rand Corporation, Santa Monica, CA, 1989.
16. A. M. Davies, 'Numerical modeling of stratified flow: a spectral approach', *Continental Shelf Res.*, **2**, 275–300 (1983).
17. A. F. Blumberg and G. L. Mellor, 'A coastal ocean numerical model', in J. Sundermann and K.-P. Holz (eds), *Mathematical Modeling of Estuarine Physics, Proc. Int. Symp.*, Hamburg, August 1978, Springer, Berlin, 1980, pp. 102–132.
18. A. F. Blumberg and G. L. Mellor, 'A description of a three dimensional coastal ocean circulation model', in N. S. Heaps (ed.), *Three Dimensional Coastal Ocean Circulation Models, Coastal and Estuarine Sciences*, Vol. 4, AGU, Washington, DC, 1987, pp. 1–16.
19. L. Y. Oey, 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 (1985).
20. L. Y. Oey, G. L. Mellor and R. I. Hires, 'A three-dimensional simulation of the Hudson–Raritan estuary. Part II: Comparison with observations', *J. Phys. Oceanogr.*, **15**, 1693–1709 (1985).
21. L. Y. Oey, 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 (1985).
22. B. Galperin and G. L. Mellor, 'A time-dependent, three-dimensional model of the Delaware Bay and estuarine system. Part 1: Description of the model and tidal analysis', *Estuarine, Coastal, Shelf Sci.*, **31**, 231–253 (1990).
23. B. Galperin and G. L. Mellor, 'A time-dependent, three-dimensional model of the Delaware Bay and estuarine system. Part 2: Three-dimensional flow field and residual circulation', *Estuarine, Coastal, Shelf Sci.*, **31**, 255–281 (1990).
24. T. J. Simons, 'Development of three-dimensional numerical models of the Great Lakes', *Canada Center for Inland Waters, Scientific Series No. 12*, Burlington, Ontario, 1973.
25. T. J. Simons, 'Verification of numerical models of Lake Ontario, Part I. Circulation in spring and early summer', *J. Phys. Oceanogr.*, **4**, 507–523 (1974).
26. J. J. Dronker, 'Tidal computations for rivers, coastal areas and seas', *J. Hydraul. Div., ASCE*, **95**, 44–77 (1969).
27. E. Bertolazzi, 'Metodo PCG ed applicazione ad un modello di acque basse', *Thesis*, Department of Mathematics, University of Trento, 1990.
28. V. Casulli, 'Eulerian–Lagrangian methods for hyperbolic and convection dominated parabolic problems', in C. Taylor, D. R. J. Owen and E. Hinton (eds), *Computational Methods for Nonlinear problems*, Pineridge, Swansea, 1987, Chap. 8, pp. 239–269.
29. R. T. Cheng, V. Casulli and S. N. Milford, 'Eulerian–Lagrangian solution of the convection–dispersion equation in natural coordinates', *Water Resources Res.*, **20**, 944–952 (1984).
30. D. Greenspan and V. Casulli, *Numerical Analysis for Applied Mathematics, Science, and Engineering*, Addison-Wesley, Reading, MA, 1988.
31. V. Casulli and F. Notarnicola, 'An Eulerian–Lagrangian method for tidal current computation', in B. A. Schrefler and O. C. Zienkiewicz (eds), *Computer Modelling in Ocean Engineering*, Balkema, Rotterdam, 1988, pp. 237–244.
32. R. T. Cheng and J. W. Gartner, 'Harmonic analysis of tides and tidal currents in south San Francisco Bay, California', *Estuarine, Coastal, Shelf Sci.*, **21**, 57–74 (1985).

33. R. A. Walters, R. T. Cheng and T. J. Conomos, 'Time scales of circulation and mixing processes of San Francisco Bay waters', *Hydrobiologia*, **129,** 13–36 (1985).
34. G. Volpi and P. Sguazzero, 'La propagazione della marea nella Laguna di Venezia: un modello di simulazione e il suo impiego nella regolazione delle bocche di porto', *Riv. Ital. Geofis. Sci. Affini*, **IV,** 67–74 (1977).
35. G. Umgiesser, J. Sundermann and E. Runca, 'A semi-implicit finite element model for the Lagoon of Venice', in B. A. Schrefler and O. C. Zienkiewicz (eds), *Computer Modelling in Ocean Engineering*, Balkema, Rotterdam, 1988, pp. 71–79.