

# An implicit three-dimensional numerical model to simulate transport processes in coastal water bodies

Lale Balas<sup>1a,\*</sup> and Erdal Özhan<sup>2b</sup>

<sup>a</sup> *Gazi University, Civil Engineering Department, Maltepe, 06570, Ankara, Turkey*

<sup>b</sup> *Middle East Technical University, Civil Engineering Department, Coastal Engineering Laboratory, 06531, Ankara, Turkey*

## SUMMARY

A three-dimensional baroclinic numerical model has been developed to compute water levels and water particle velocity distributions in coastal waters. The numerical model consists of hydrodynamic, transport and turbulence model components. In the hydrodynamic model component, the Navier–Stokes equations are solved with the hydrostatic pressure distribution assumption and the Boussinesq approximation. The transport model component consists of the pollutant transport model and the water temperature and salinity transport models. In this component, the three-dimensional convective diffusion equations are solved for each of the three quantities. In the turbulence model, a two-equation  $k-\epsilon$  formulation is solved to calculate the kinetic energy of the turbulence and its rate of dissipation, which provides the variable vertical turbulent eddy viscosity. Horizontal eddy viscosities can be simulated by the Smagorinsky algebraic sub grid scale turbulence model. The solution method is a composite finite difference–finite element method. In the horizontal plane, finite difference approximations, and in the vertical plane, finite element shape functions are used. The governing equations are solved implicitly in the Cartesian co-ordinate system. The horizontal mesh sizes can be variable. To increase the vertical resolution, grid clustering can be applied. In the treatment of coastal land boundaries, the flooding and drying processes can be considered. The developed numerical model predictions are compared with the analytical solutions of the steady wind driven circulatory flow in a closed basin and of the uni-nodal standing oscillation. Furthermore, model predictions are verified by the experiments performed on the wind driven turbulent flow of an homogeneous fluid and by the hydraulic model studies conducted on the forced flushing of marinas in enclosed seas. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: circulation; finite difference; finite element; implicit; numerical model; turbulence

---

\* Correspondence to: Gazi University, Civil Engineering Department, Maltepe, 06570, Ankara, Turkey. Tel.: + 90 312 2105429; fax: + 90 312 210412.

<sup>1</sup> E-mail: [balas@mikasa.mmf.gazi.edu.tr](mailto:balas@mikasa.mmf.gazi.edu.tr)

<sup>2</sup> E-mail: [ozhan@rorqual.cc.metu.edu.tr](mailto:ozhan@rorqual.cc.metu.edu.tr)

## 1. INTRODUCTION

A number of papers have appeared in the literature reporting numerical modelling studies of turbulent transport processes in the coastal water bodies, like bays, lagoons, estuaries, etc. Three-dimensional models extended the common two-dimensional vertically averaged models to include the vertical structure. The physical processes in coastal water bodies are generally complicated, therefore, many approximations have had to be made in several of these three-dimensional models, including: advection, bottom friction, density, turbulence models, co-ordinate systems, treatment of horizontal and vertical variables and treatment of flooding and drying [1]. Different levels of accuracy in these approximations and in numerical solution schemes result in different levels of accuracy in the model predictions [1].

Most of the three-dimensional models have used finite difference grids in the horizontal plane, but mainly different methods in the vertical. In layered models, the governing differential equations are based on the layer-averaged three-dimensional equations of continuity and momentum conservation [1]. In grid box models, three-dimensional equations of motion are solved by using a fixed finite difference grid in the vertical, through which the fluid is free to move [2,3]. Davies and Stephens [4] compared the finite difference and Galerkin methods as applied to the solution of the hydrodynamic equations. The accuracy and computational efficiency, in terms of both computer time and memory requirements, of using either the Galerkin method or the finite difference method through the vertical were considered. Calculations showed that, the Galerkin method had superior accuracy over the grid box method. If the finite difference grid is fixed in the vertical, the number of vertical grid boxes increases as depth increases, but reduces in the shallower water depths. Unfortunately, this has the effect of reducing the vertical resolution in the shallow water areas, where high vertical shears occur. This problem of reduced vertical resolution in the shallow regions can be overcome by transforming the hydrodynamic equations into sigma co-ordinates, and then applying a finite difference grid in the vertical, i.e. a sigma co-ordinate box model. This way, a constant number of grid boxes is used in the vertical at each horizontal grid point [5,6]. In transforming the curved physical region to the computational domain, the governing equations are also transformed resulting in more complex equations. Stansby [7], used a three-dimensional, semi implicit finite volume scheme for tide-induced shallow water flow with the hydrostatic pressure assumption, using the sigma co-ordinate system and incorporating a standard  $k-\epsilon$  turbulence transport model. Evaluation of horizontal gradients in the sigma co-ordinate system required high-order derivatives, which caused spurious flows, and this was avoided by obtaining these gradients in real space. Since the sigma co-ordinate transformation is non-conformal, the transformed governing equations become very complex, with some of the additional terms involving cross derivatives. When large bathymetric irregularities exist, the layer thickness considerably increases in deep water regions, whereas decreases in shallow water areas. Huang and Spaulding [8] used an algebraic transformation within the  $\sigma$ -co-ordinate transformation to achieve higher resolution.

In the developed model, the depth-following co-ordinate system is applied [9,10]. The solution method is a composite finite difference–finite element method. The governing equations, written in the Cartesian co-ordinates, are solved by the Galerkin weighted residual method in the vertical plane and by finite difference approximations in the horizontal plane,

without any co-ordinate transformation. When the integrals of the functions are to be performed, all the horizontal gradient terms are corrected by considering the horizontal gradients of the basis functions. The water depths are divided into the same number of layers following the bottom topography. Therefore, the vertical layer thickness is proportional to the local water depth. Higher resolution can be achieved either by increasing the number of layers, or by grid clustering. Grids can be concentrated near the surface layer, near the bottom layer or around any specified water depth by grid clustering. The governing equations are solved implicitly. The finite difference approximations can handle the variations in the horizontal mesh sizes. The horizontal mesh size  $\Delta x$  in the  $x$ -co-ordinate is orthogonal to the horizontal mesh size  $\Delta y$  in the  $y$ -co-ordinate. The horizontal mesh sizes  $\Delta x$  and  $\Delta y$  can be different from each other. Furthermore,  $\Delta x$  can vary along the  $x$  co-ordinate and  $\Delta y$  can vary along the  $y$  co-ordinate. The volume of water in a typical shallow coastal water body such as a lagoon, may show seasonal variations, causing some water areas in the system to dry out or vice versa. Therefore, the boundaries of the coastal area need to be defined as movable boundaries. In the model developed, the flooding and drying processes can also be considered in the treatment of shoreline boundaries. The model predictions were verified by using several experimental and analytical results published in the literature and its successful use for a variety of real life cases was demonstrated [11].

## 2. THEORETICAL CONSIDERATIONS

Three-dimensional mathematical models are necessary to simulate wind driven circulations and density currents that occur in coastal waters, which are stratified by salinity and temperature layers resulting in significant vertical and lateral density gradients. The developed three-dimensional mathematical model is capable of computing the water levels and water particle velocity distributions in three principal directions by solving the Navier–Stokes equations. Only two simplifying approximations are used in the model. First, it is assumed that the weight of the fluid balances the pressure, i.e. the hydrostatic pressure distribution assumption; and second, the density differences are neglected unless the differences are multiplied by the gravity, i.e. the Boussinesq approximation. Since the horizontal extent of the coastal sea is much larger than the vertical extent, the vertical component of the motion is much weaker than the horizontal velocity [12]. If the flow is predominantly horizontal, and the vertical acceleration is small compared with the gravity acceleration, the equation of motion can be reduced to the simple hydrostatic law. Although the vertical acceleration is assumed to be negligible, the vertical velocity can be calculated from the equation of continuity.

The governing hydrodynamic equations in the three-dimensional Cartesian co-ordinate system are as follows:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (1)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = fv - \frac{1}{\rho_0} \frac{\partial P}{\partial x} + 2 \frac{\partial}{\partial x} \left( v_x \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( v_y \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \right) + \frac{\partial}{\partial z} \left( v_z \frac{\partial u}{\partial z} \right) \quad (2)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = -fu - \frac{1}{\rho_0} \frac{\partial p}{\partial y} + 2 \frac{\partial}{\partial y} \left( v_y \frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial x} \left( v_x \left( \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \right) + \frac{\partial}{\partial z} \left( v_z \frac{\partial v}{\partial z} \right) \quad (3)$$

$$\frac{\partial p}{\partial z} = -\rho g \quad (4)$$

where  $x$ ,  $y$  are horizontal co-ordinates;  $z$  is the vertical co-ordinate;  $t$  is time;  $u$ ,  $v$ ,  $w$  are velocity components in  $x$ -,  $y$ -,  $z$ -directions at any grid locations in space;  $v_x$ ,  $v_y$ ,  $v_z$  are eddy viscosity coefficients in  $x$ -,  $y$ - and  $z$ -directions respectively;  $f$  is the Coriolis coefficient;  $\rho(x, y, z, t)$  is the *in situ* water density;  $\rho_0$  is the reference density;  $g$  is gravitational acceleration;  $p$  is pressure.

The numerical model can simulate the flows induced by the density gradients. The density of sea water is a function of its salinity and its temperature. The temperature and salinity variations are calculated by solving the three-dimensional convection–diffusion equation which is written as:

$$\frac{\partial Q}{\partial t} + u \frac{\partial Q}{\partial x} + v \frac{\partial Q}{\partial y} + w \frac{\partial Q}{\partial z} = \frac{\partial}{\partial x} \left( D_x \frac{\partial Q}{\partial x} \right) + \frac{\partial}{\partial y} \left( D_y \frac{\partial Q}{\partial y} \right) + \frac{\partial}{\partial z} \left( D_z \frac{\partial Q}{\partial z} \right) \quad (5)$$

where  $D_x$ ,  $D_y$  and  $D_z$  are turbulent diffusion coefficients in  $x$ -,  $y$ - and  $z$ -directions respectively;  $Q$  is temperature ( $T$ ) or salinity ( $S$ ).

The conservation equation for a pollutant constituent is:

$$\frac{\partial C}{\partial t} + u \frac{\partial C}{\partial x} + v \frac{\partial C}{\partial y} + w \frac{\partial C}{\partial z} = \frac{\partial}{\partial x} \left( D_x \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left( D_y \frac{\partial C}{\partial y} \right) + \frac{\partial}{\partial z} \left( D_z \frac{\partial C}{\partial z} \right) - k_p C \quad (6)$$

where  $C$  is the pollutant concentration;  $k_p$  is the decay rate of the pollutant;  $D_x$ ,  $D_y$  and  $D_z$  are turbulent diffusion coefficients in  $x$ -,  $y$ - and  $z$ -directions respectively.

Integrating the continuity equation over the depth and using the kinematic condition at the free surface leads to the following depth integrated continuity equation:

$$\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x} \left( \int_{-h}^{\eta} u \, dz \right) + \frac{\partial}{\partial y} \left( \int_{-h}^{\eta} v \, dz \right) = 0 \quad (7)$$

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

The horizontal pressure gradient terms are:

$$\frac{\partial p}{\partial x} = \frac{\partial}{\partial x} \int_z^{\eta} g \rho \, dz = \int_z^{\eta} g \frac{\partial \rho}{\partial x} \, dz + g \rho_s \frac{\partial \eta}{\partial x} \quad (8)$$

where  $\rho_s$  is the density at the surface. A similar expression can be written for  $\partial p / \partial y$ .

The two-equation  $k$ - $\epsilon$  turbulence model is used for the turbulence modelling. The model equations for the kinetic energy and its dissipation rate are:

$$\frac{\partial k}{\partial t} + u \frac{\partial k}{\partial x} + v \frac{\partial k}{\partial y} + w \frac{\partial k}{\partial z} = \frac{\partial}{\partial z} \left( \frac{v_z}{\sigma_k} \frac{\partial k}{\partial z} \right) + P + B - \epsilon + \frac{\partial}{\partial x} \left( D_x \frac{\partial k}{\partial x} \right) + \frac{\partial}{\partial y} \left( D_y \frac{\partial k}{\partial y} \right) \quad (9)$$

$$\begin{aligned} \frac{\partial \epsilon}{\partial t} + u \frac{\partial \epsilon}{\partial x} + v \frac{\partial \epsilon}{\partial y} + w \frac{\partial \epsilon}{\partial z} = \frac{\partial}{\partial z} \left( \frac{v_z}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial z} \right) + C_{1\epsilon} \frac{\epsilon}{k} (P + C_{3\epsilon} B) - C_{2\epsilon} \frac{\epsilon^2}{k} + \frac{\partial}{\partial x} \left( D_x \frac{\partial \epsilon}{\partial x} \right) \\ + \frac{\partial}{\partial y} \left( D_y \frac{\partial \epsilon}{\partial y} \right) \end{aligned} \quad (10)$$

where  $k$  is kinetic energy;  $\epsilon$  is the rate of dissipation of kinetic energy;  $v_z$  is the vertical eddy viscosity;  $D_x$ ,  $D_y$  are horizontal diffusion coefficients in  $x$ - and  $y$ -directions respectively;  $P$  is the stress production of the kinetic energy;  $B$  is the buoyancy production of the kinetic energy which is defined by:

$$B = \frac{g}{\rho_0} \frac{v_z}{Pr} \frac{\partial p}{\partial z} \quad (11)$$

where  $Pr$  is the turbulent Prandtl or Schmidt number. Experiments have shown that, the turbulent Prandtl or Schmidt number, varies slightly in a flow and from one flow to the other [13]. Therefore, it is considered as a constant,  $Pr = 0.7$ .

The stress production of the kinetic energy is defined by:

$$P = v_h \left[ 2 \left( \frac{\partial u}{\partial x} \right)^2 + 2 \left( \frac{\partial v}{\partial y} \right)^2 + \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 \right] + v_z \left[ \left( \frac{\partial u}{\partial z} \right)^2 + \left( \frac{\partial v}{\partial z} \right)^2 \right] \quad (12)$$

where  $v_h$  is the horizontal eddy viscosity and  $u$  and  $v$  are the horizontal water particle velocities in  $x$ - and  $y$ -directions respectively.

The vertical eddy viscosity is calculated by:

$$v_z = C_\mu \frac{k^2}{\epsilon} \quad (13)$$

The following universal empirical constants are used [13]:  $C_\mu = 0.09$ ,  $\sigma_\epsilon = 1.3$ ,  $C_{1\epsilon} = 1.44$ ,  $C_{2\epsilon} = 1.92$ ,  $C_{3\epsilon} = 1$  if  $B > 0$  (unstable stratification) and  $C_{3\epsilon} = 0.2$  if  $B < 0$  (stable stratification).

The standard  $k$ - $\epsilon$  model assumes the local isotropy of the turbulence, where horizontal eddy viscosity is equal to the vertical eddy viscosity. If the horizontal motion has an intensity and length scale greater than the vertical motion, as in shallow water bodies, the standard single length scale,  $k$ - $\epsilon$  turbulence model underestimates the effective horizontal eddy viscosities. To account for large scale turbulence generated by the horizontal shear, horizontal eddy viscosity can be simulated by the Smagorinsky algebraic subgrid scale turbulence model [14]:

$$v_h = 0.01 \Delta x \Delta y \left( \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial y} \right)^2 + \frac{1}{2} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 \right)^{1/2} \quad (14)$$

In the case of stratified flows, as observed from the experimental results of Huq and Stretch [15], the influence of stratification on turbulence in the horizontal direction is negligible. Hence, the horizontal eddy diffusivities are approximately equal to the horizontal eddy viscosities. Whereas, the vertical diffusivity,  $D_z$ , is expressed as:

$$D_z = \frac{v_z}{Pr} \quad (15)$$

where  $v_z$  is the vertical eddy viscosity coefficient.

There are four types of boundaries, namely; free surface, sea bed, open sea and coastal land boundaries.

### 2.1. Free surface

The wind induced shear stress at the free surface is expressed as:

$$[\tau_{wx}, \tau_{wy}] = \rho_a C_d [u_w, v_w] \sqrt{u_w^2 + v_w^2} \quad (16)$$

where  $\tau_{wx}$ ,  $\tau_{wy}$  are the wind stress components and  $u_w$  and  $v_w$  are the wind velocity components (m/s) in  $x$ - and  $y$ -directions respectively;  $\rho_a$  is the air density and  $C_d$  is the drag coefficient of air.

In the literature, several formulations exist for the wind drag coefficient, ranging from a constant value for all wind speeds to complicated formulae that take into account the wind speed and direction and the roughness of the sea surface (wave height). The following formulation of drag coefficient proposed by Large and Pond [16] and mostly applied in the three-dimensional models, is used in the present model:

$$C_d = \begin{cases} 1.2 * 10^{-3} & W < 11 \text{ m/s} \\ (0.49 + 0.065W) * 10^{-3} & 11 \text{ m/s} \leq W < 25 \text{ m/s} \end{cases} \quad (17)$$

where  $W$  is the wind velocity in m/s.

The wind induced shear stress at the surface results in a water velocity gradient below the surface:

$$\tau_{wx} = \rho v_z \frac{\partial u}{\partial z}; \quad \tau_{wy} = \rho v_z \frac{\partial v}{\partial z} \quad (18)$$

At the free surface, the pollutant and salt fluxes are taken as zero (no atmospheric deposition), whereas the heat flux is:

$$D_z \frac{\partial T}{\partial z} = \frac{K}{\rho C_p} (T_s - T_e) \tag{19}$$

where  $K$  is the surface heat transfer coefficient;  $\rho$  is the water density;  $C_p$  is the specific heat of water;  $T_s$  is the surface water temperature;  $T_e$  is the equilibrium temperature.

The boundary conditions for the kinetic energy and its rate of dissipation also depend on the wind shear. When there exists a wind shear [13]:

$$k_s = \frac{u_{*s}^2}{\sqrt{c_\mu}}; \quad \epsilon_s = \frac{|u_{*s}|^3}{\kappa \Delta z_s} \tag{20}$$

Otherwise,

$$\frac{\partial k_s}{\partial z} = 0; \quad \epsilon_s = \frac{(k_s \sqrt{C_\mu})^{3/2}}{0.07 \kappa H} \tag{21}$$

where  $u_{*s}$  is the surface shear velocity;  $c_\mu$  is the universal empirical constant, 0.09;  $\Delta z_s$  is the distance from the surface to the first grid point below;  $\kappa$  is the Karman constant, which is  $\kappa = 0.42$ ;  $H$  is the total water depth.

### 2.2. Sea bed

At the sea bed, the bottom shear stress is determined by matching velocities with the logarithmic law of the wall:

$$\tau_{bx} = \left[ v_z \frac{\partial u}{\partial z} \right]_b = \rho_0 C_f u_b \sqrt{u_b^2 + v_b^2}; \quad \tau_{by} = \left( v_z \frac{\partial v}{\partial z} \right)_b = \rho_0 C_f v_b \sqrt{u_b^2 + v_b^2} \tag{22}$$

where  $\tau_{bx}$ ,  $\tau_{by}$  are the bottom shear stress components and  $u_b$ ,  $v_b$  are the horizontal velocity components at the grid point nearest the bottom;  $\rho_0$  is the mean water density, and  $C_f$  is the empirical coefficient for bottom friction. If sufficient resolution near the bottom boundary is provided,  $C_f$  can be estimated from the logarithmic law of the wall:

$$C_f = \left( \frac{1}{\kappa} \ln \left( \frac{\Delta z_b}{z_0} \right) \right)^{-2} \tag{23}$$

where  $\Delta z_b$  is the distance from the bottom to the first grid point above;  $z_0$  is a parameter which is dependent on the local bottom roughness and can be taken as 1 cm.

When the bottom boundary layer is not sufficiently resolved,  $C_f$  is specified as a constant (typically in the range 0.002–0.003). The wall region occupies the range of  $30 < z^+ < 100$ , where  $z^+$  is calculated by using the following expression:  $z^+ = \Delta z_b u_{*b} / \nu$ , in which  $u_{*b}$  is the bed friction velocity. In the vertical discretization, the first vertical mesh point should lie within this region.

The kinematic boundary condition at the sea bed is:

$$w_b = -u_b \frac{\partial h}{\partial x} - v_b \frac{\partial h}{\partial y} \quad (24)$$

The bottom turbulent kinetic energy  $k_b$  and dissipation  $\epsilon_b$ , are determined from the following relations:

$$k_b = \frac{u_{*b}^2}{\sqrt{C_\mu}}; \quad \epsilon_b = \frac{u_{*b}^3}{\kappa \Delta z_b} \quad (25)$$

At the sea bottom, the gradients of temperature, salinity and pollutant are taken as zero, indicating that there are no advective and diffusive fluxes across the sea bed.

### 2.3. Coastal land boundary

The volume of water in a coastal system, like lagoons, may show seasonal variations, causing some water areas in the system to dry out or land areas to be flooded. Therefore, the water boundaries (e.g. the shoreline) should be handled in a way to simulate the flooding and drying processes. Once the free surface and the new water velocities are computed over the whole computational domain, the total water depth and vertical grid spacings have to be updated before proceeding to the next time step. The water surface slope is calculated at each time step. If the water surface slope is positive, using this slope the water surface is extended until it intersects the coastal land boundary. The horizontal distance travelled on the land should be a value which is at least one-fifth of the horizontal space increment to be accepted as a movement. Then, the horizontal space increment is modified and the water depth in the middle of the grid is calculated. A negative value for the total water depth  $H$ , has no physical meaning, therefore the discrete total depths  $H_{i,j}$  are expressed as:

$$H_{i,j} = \max(0, h_{i,j} + \eta_{i,j}) \quad (26)$$

where  $h(x, y)$  is the water depth measured from the undisturbed water surface;  $\eta(x, y, t)$  is water surface elevation.

The water depth at the middle of each grid cell is compared with  $L_b$ , which is a length scale of bed roughness. When the calculated water depth is less than  $L_b$  or zero, that cell is a dry cell which may be flooded again if the total water depth becomes positive at a later time. In a dry cell, the velocity components  $u$  or  $v$  across the side of the grid cell are forced to vanish in order not to permit flow across the grid cell sides. The shoreline boundary, which may vary with time as the consequence of flooding and drying processes, is defined by the condition of no mass flux in the normal direction. Across the shoreline, the normal gradients of temperature, salinity and pollutant are also taken as zero (i.e. no advective and diffusive fluxes across the shoreline boundary). At the inflow/outflow locations along the land boundary (shoreline),  $k$  and  $\epsilon$  can be described from fully developed channel flow data [17] as:

$$k = 0.004u_d^2; \quad \epsilon = c_\mu^{3/4} \frac{k^{3/2}}{0.09b} \quad (27)$$

where  $u_d$  is the inflow velocity and  $b$  is inflow inlet width.



#### 2.4. Open sea boundary

The open sea boundary is the lateral boundary where the open sea can flow in or out of the coastal water body. At the open sea boundary, when there is no tidal motion, velocities normal to the boundary are computed in the middle of the cell where the water depth is known and the horizontal gradient terms are treated accordingly (Appendix A). For tidal flows, the following equations are used at the open sea boundary:

$$\eta = a \sin\left(2\pi \frac{t}{T_w}\right) \quad (28)$$

$$V_n = (gH)^{1/2} a H^{-1} \cos\left(\frac{2\pi}{L_w} \frac{\Delta n}{2} + \frac{2\pi}{T_w} t\right) \quad (29)$$

where  $T_w$  and  $L_w$  are tidal period and wave length respectively;  $H$  is the total water depth;  $V_n$  is the depth averaged velocity normal to the boundary;  $\Delta n$  is the horizontal grid distance normal to the lateral boundary;  $a$  is the tidal amplitude.

### 3. NUMERICAL SOLUTION SCHEME

Equations are solved numerically by approximating the horizontal gradient terms using a staggered finite difference scheme. In the vertical plane however, the Galerkin method of finite elements is utilized. Water depths are divided into the same number of layers following the bottom topography. At all nodal points, the ratio of the length (thickness) of each element (layer) to the total depth is constant. By following the finite element approach, the values of velocities  $u$ ,  $v$ ,  $w$ ; the turbulent eddy viscosities  $\nu_x$ ,  $\nu_y$ ,  $\nu_z$ ; the temperature,  $T$ ; the salinity  $S$ ; the pollutant concentration  $C$ ; the turbulent diffusivities  $D_x$ ,  $D_y$ ,  $D_z$ ; the kinetic energy  $k$ ; the rate of dissipation of kinetic energy  $\epsilon$ ; the pressure  $p$ , at any point over the flow depth are written in terms of the discrete values of these variables at the vertical nodal points by using linear shape functions.

$$\tilde{G} = N_1 G_1^k + N_2 G_2^k \quad (30)$$

$$N_1 = \frac{z_2 - z}{l_k}; \quad N_2 = \frac{z - z_1}{l_k}; \quad l_k = z_2 - z_1 \quad (31)$$

where  $\tilde{G}$  is the approximation or shape function,  $G$  is any of the variables,  $k$  is the element number,  $N_1$  and  $N_2$  are the interpolation functions,  $l_k$  is the length of the  $k$ th element,  $z_1$  and  $z_2$  are the beginning and end elevations of the element  $k$ , and  $z$  is the transformed variable that changes from  $z_1$  to  $z_2$  in an element.

The approximate expressions for variables are substituted into the governing equations and the residual errors ( $R$ ), are minimized by using the Galerkin procedure [18,19]. To increase the vertical resolution, wherever necessary, grid clustering can be applied in the vertical plane [20]. Grids can be concentrated near the bottom, surface, or intermediate layers.

After the application of the Galerkin method, any derivative terms with respect to horizontal co-ordinates appearing in the equations are replaced by their central finite difference approximations. The mesh size may be varied in the horizontal plane. The local element matrices for all elements on a vertical line are grouped together to form the global matrix equation for the unknown nodal time derivatives of the variables at a grid point on the horizontal plane. The sea bed and sea surface boundary conditions are taken into account, while building up the global matrices over the water depth [11]. The derivation of matrix elements are given in Appendix B.

The system of non-linear equations are solved by the Crank–Nicholson method, which has second-order accuracy in time. To provide this accuracy, difference approximations are developed at the midpoint of the time step. The temporal first derivative is approximated at  $(t + 1/2)$  and all other variables and derivatives at this time are determined by averaging the difference approximations at the beginning ( $t$ ) and at the end ( $t + 1$ ) of the time increment. Resultant set of implicit equations are solved by an iterative method, which is controlled by underrelaxation. Underrelaxation is typically employed to make a non-convergent system convergent or to hasten convergence by dampening out the oscillations.

### 3.1. Handling of the continuity equation

After the estimation of horizontal velocities, the vertical velocities,  $w$ , are calculated from the continuity equation, at each time step. The approximate expressions for horizontal velocities  $u$  and  $v$ , are substituted into the continuity equation:

$$\frac{\partial \tilde{u}}{\partial x} + \frac{\partial \tilde{v}}{\partial y} + \frac{\partial \tilde{w}}{\partial z} = 0 \quad (32)$$

Then, the residual error is minimized by the Galerkin method. Since the horizontal velocities are known, the equation to calculate the vertical velocity becomes as:

$$\begin{aligned} w_2^k = w_1^k - \frac{1}{2} \left[ \frac{\partial l_k}{\partial x} (u_1^k - u_2^k) + l_k \left( \frac{\partial u_1^k}{\partial x} + \frac{\partial u_2^k}{\partial x} \right) + \frac{\partial l_k}{\partial y} (v_1^k - v_2^k) + l_k \left( \frac{\partial v_1^k}{\partial y} + \frac{\partial v_2^k}{\partial y} \right) \right] \\ - l_k \left( (u_1^k - u_2^k) \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial x} \right) + (v_1^k - v_2^k) \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial y} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial y} \right) \right) \end{aligned} \quad (33)$$

where  $k = 1, 2, \dots, m$  represents the layer number and  $l_k$  is the length of the element  $k$ .

Therefore, at each point, vertical velocities are computed, starting from the bottom layer to the surface layer, in order to satisfy the continuity equation.

## 4. APPLICATION OF THE MODEL

## 4.1. Wind driven flow

The comparison of model predictions with an analytical solution is performed for a steady wind driven circulatory flow in a closed basin with a flat bed and with linearized bottom friction. The advection, coriolis and the horizontal diffusion terms are neglected. The momentum equation requires a balance among the gravitational force due to the surface elevation gradient, the vertical diffusion of momentum, the surface wind drag and the bottom friction force. The analytic solution for a constant vertical eddy viscosity and linearized bottom friction is given as [8]:

$$u = \frac{g}{6v_z} \frac{\partial \eta}{\partial x} (3z^2 - H^2) + \frac{\tau_w}{2\rho v_z} (H + 2z) \quad (34)$$

$$\frac{\partial \eta}{\partial x} = \frac{3}{2} \frac{\tau_w}{\rho g H} \frac{(2v_z + u_f H)}{(3v_z + u_f H)} \quad (35)$$

where  $u$  is the horizontal velocity,  $u_f$  is the linearized bottom friction coefficient,  $\eta$  is the surface elevation,  $v_z$  is the vertical eddy viscosity,  $g$  is the gravitational acceleration,  $\tau_w$  is the wind stress,  $H$  is the total water depth,  $\rho$  is the water density,  $z$  is the vertical co-ordinate which is equal to zero at the sea surface, whereas it is equal to  $(-H)$  at the sea bottom.

In the numerical simulation, the following parameters are used;  $\tau_w = 1$  dyne/cm<sup>2</sup>,  $H = 40$  m,  $g = 9.8$  m/s<sup>2</sup>,  $\rho = 1025$  kg/m<sup>3</sup>,  $v_z = 0.03$  m<sup>2</sup>/s,  $u_f = 0.5$  cm/s. The basin area is  $12 \times 12$  km with grid size of  $1 \times 1$  km. Simulations were performed by using 6, 10 and 20 layers. Numerical calculations were carried out first with an equal layer thickness along the water depth, and later with the grid clustering near surface and bottom layers. Velocity profiles for both cases and the analytical solution are shown in Figure 1. For comparing numerical model results with the analytical solution, the root mean square errors and bias are calculated and listed in Table I. These results show that, model predictions approach the analytical solutions, as the number of layers is increased or the layers are concentrated near the boundaries, i.e. by increasing the resolution near the surface and bottom boundaries. The computer program is written in the FORTRAN 77 programming language and executed by using the Scalable Power Parallel System-2 (SP/2), which has an AIX/6000 operating system. The corresponding central processing unit times (CPU) of the computer for the implicit solution are presented in Table II. The CPU time increases only approximately 1.8 per cent for simulations when grid clustering is used. To obtain an accuracy with root mean square error under 0.02 cm/s, either 20 equal layers or 10 layers with grid clustering near the bottom and the surface can be used. The CPU time for the solution with 20 equal layers is 1.5 times that of the solution with 10 clustered layers near the bottom and the surface.

The predictions of the numerical model are then compared with the wind driven turbulent flow of an homogeneous fluid, as shown in Figure 2. The numerical model is executed with the two-equation  $k-\epsilon$  turbulence model. The experiment results are due to Tsanis and Leutheusserr [21] who used a laboratory basin with a length of 2.4 m, a width of 0.72 m, and

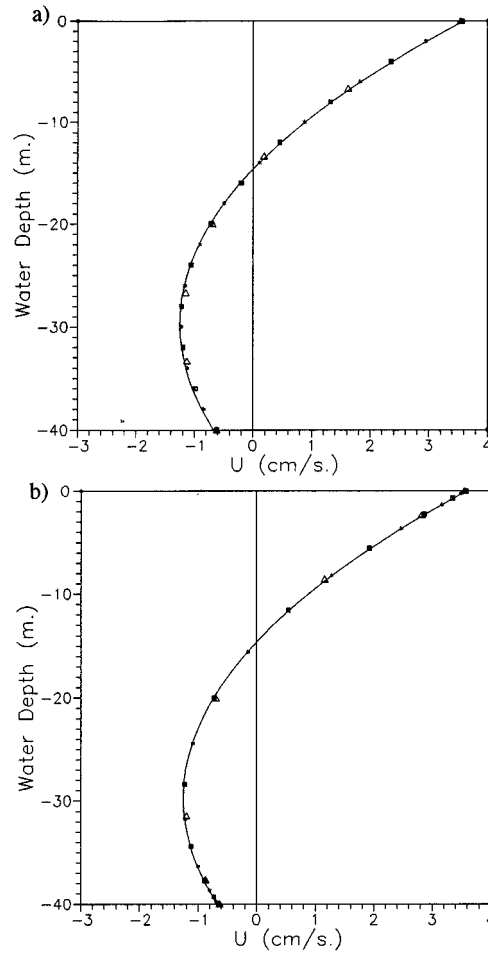


Figure 1. Wind driven flow with bottom friction. (a) Equal vertical layer thickness. (b) Grid clustering near surface and bottom layers (solid line: analytical solution; symbols: model predictions where, \*: 20 layers; □: 10 layers; △: 6 layers).

a depth of  $H = 0.05$  m. The Reynolds Number,  $R_s = u_s H \rho / \mu$  was 3000 ( $u_s$  is the surface velocity,  $H$  is the depth of the flow,  $\rho$  is the density of water and  $\mu$  is the dynamic viscosity). The root mean square error is 0.08 and bias is  $-0.027$ .

#### 4.2. Seiche motion

A particular seiche motion, the uni-nodal standing oscillation, is the next case for which the present numerical model is verified. Wind blowing over an enclosed water body piles up water

Table I. The root mean square errors and bias of horizontal velocity for wind driven flow in a closed basin with linearized bottom friction.

	Equal layer thickness		Grid clustering	
	RMS (mm/s)	Bias (mm/s)	RMS (mm/s)	Bias (mm/s)
6 layers	0.43	0.093	0.35	0.0599
10 layers	0.27	0.084	0.17	0.0335
20 layers	0.16	0.045	0.05	0.0084

Table II. Comparison of the CPU times (in min).

	Equal layer thickness	Grid clustering
6 layers	3:15	3:18
10 layers	4:50	4:55
20 layers	7:23	7:31

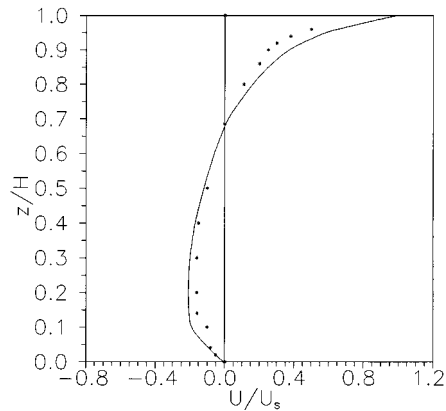


Figure 2. Horizontal velocity profile of the wind driven turbulent flow of an homogenous fluid where, \*: experimental results; —: numerical model results.

on one side and raises the water. If the wind suddenly dies away, then the piled up water is released and the water level starts to oscillate. A rectangular closed water body of constant water depth is considered. The depth of the water is sufficiently small to allow the shallow water theory to be employed. The fluid is assumed to be ideal. The analytical solution for this flow situation is expressed by the following equations [2]:

$$\eta = a \cos(k_w x) \cos(\sigma_w t) \quad (36)$$

$$u = \frac{aC_w \sin(k_w x) \sin(\sigma_w t)}{h} \quad (37)$$

$$w = -\frac{a\sigma_w(h + \eta) \cos(k_w x) \sin(\sigma_w t)}{h} \quad (38)$$

where  $a$  is the wave amplitude,  $k_w$  is the wave number ( $2\pi/L_w$ ),  $L_w$  is the wave length ( $C_w T_w$ ),  $C_w$  is the wave celerity ( $gh$ )<sup>0.5</sup>,  $h$  is the water depth,  $\sigma_w$  is the angular frequency ( $2\pi/T_w$ ),  $x$  is the distance from the left boundary.

For this test study, a closed rectangular basin is considered with a depth of  $h = 7$  m, a length of  $L_b = 3.4$  km, and a width of  $W_b = 1.6$  km. The period of oscillation for the wave is  $T_w = 820$  s and the wave length is  $L_w = 6.8$  km, which is twice the length of the basin. The standing wave is introduced as the water surface elevation data at time  $t = 0$  in the form of a half cosine wave with a maximum amplitude of 0.4 m at the left-hand closed boundary ( $x = 0$ ), and the minimum on the right-hand closed boundary ( $x = 3.4$  km). The velocities in the water are zero everywhere, i.e. the water is instantaneously at rest. At this time, the water elevation starts to fall on the left and rise on the right, and a system of currents is set up. Horizontal grid dimensions are  $200 \times 200$  m and the total simulation time is 1640 s ( $t = 2T_w$ ). The water depth is divided into seven equal layers.

At the end of the simulation time, the numerical and analytical results of the surface elevation are almost identical as compared in Figure 3. The root mean square error in the computations of water surface elevation is 1.2 mm and the bias is 0.08 mm. Figure 4 shows the computed velocity vectors after 1435 s ( $t = 7T_w/4$ ), when the velocities are maximal and the water surface is horizontal. At this stage of the oscillation, comparisons of the numerical results with the analytical results show that, the root mean square error in horizontal velocity

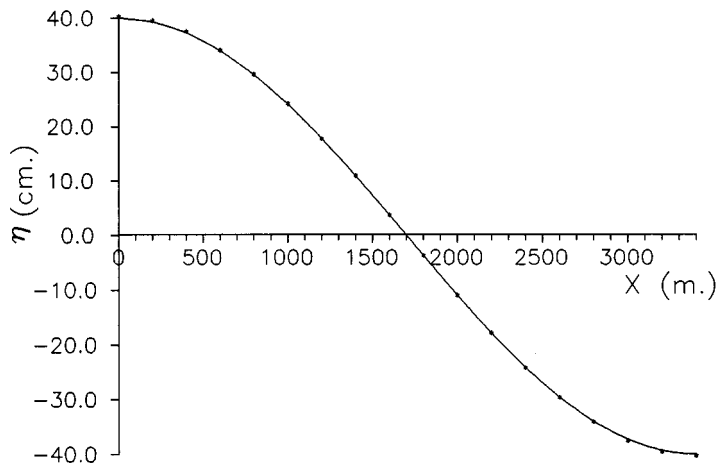


Figure 3. The water surface elevation in a closed rectangular basin after 1640 s where, —: analytical solution; \*: numerical solution.

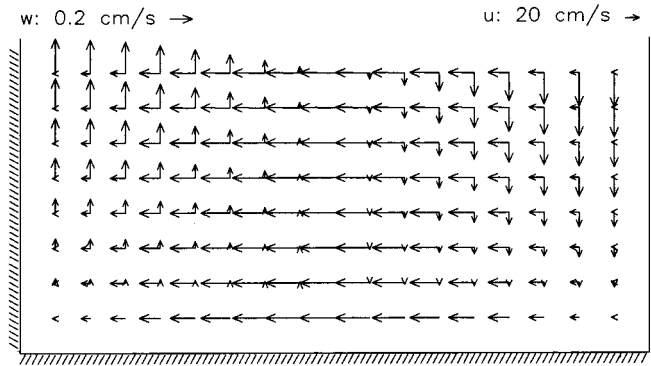


Figure 4. The velocity field due to a uni-nodal standing oscillation in a closed rectangular basin after 1435 s as computed by the numerical model.

components is 0.82 mm/s and the bias is 0.64 mm/s. The root mean square error in vertical velocity components is 0.043 mm/s and the bias is 0.032 mm/s.

#### 4.3. Forced flushing of marinas in enclosed seas

The next test study is on the forced flushing of marinas in enclosed seas. Construction of a marina disturbs the natural flow patterns and normally deteriorates the water quality in and around the project site. The breakwater, the purpose of which is to provide a physical barrier against waves, also becomes a barrier against the movement of water and sand. Water enclosed in a marina basin is normally tranquil and has a restricted contact with the outside sea. Water exchange takes place only through the entrance. The cross sectional area of the entrance is often small and the exchange is low especially in areas where the tidal range is small. Water that enters into the basin cannot freely circulate. Limited water circulation may result in poor water quality levels in the marina. It is often necessary to apply some special design features to enhance flushing of marinas. The use of a 'morning-glory spillway' like structure to pump out water from the marina for improving the flushing performance is investigated [22]. A physical model study was performed in the Coastal Engineering Laboratory of the Middle East Technical University. The measured velocities in the physical model at the grid points neighbouring the intake are used as the boundary conditions in the mathematical model. The length scale of the model was 1/50. The length and width of the rectangular model marina basin were 5.80 and 2.8 m respectively. The average water depth was 0.2 m. Using the length scale of 1/50, these dimensions correspond to a prototype marina of  $290 \times 140$  m, with a water depth of 10 m. Surface water was withdrawn from the marina by installing a structure similar to the morning-glory spillway. The structure consists of a vertically placed conical shaft connected to a horizontal discharge pipe placed on the sea bed. The water taken from the basin was pumped into the open sea. The discharge point was selected so that, the motion resulting from the pumped water does not affect the circulation in the basin. Velocity

measurements were taken with a 'Minilab SD-12' microscale 3-axis ultrasonic current meter. The morning-glory shaped intake structure was placed at various locations in the basin. Two of them, which are at the corners, are presented in this paper. The choice of the corner locations is due to the concern for affecting the manoeuvring space of the crafts inside the marina. The grid system used has a square mesh size of  $10 \times 10$  m. The vertical eddy viscosity is calculated by the  $k-\epsilon$  model and horizontal eddy viscosities are predicted by the sub-grid scale turbulence model. The water depth is divided into ten layers of equal thickness. The density of water is assumed to be constant and equal to  $1025 \text{ kg/m}^3$ . At  $t = 0$ , the pump is started, so that the water begins to flow in the intake, whereas the remaining part of the water body is assumed to be at rest and the water surface is assumed to be horizontal. The land boundaries are taken as fixed boundaries. Velocities measured in the physical model at the grid points neighbouring the intake are used as the boundary conditions of the intake location in the mathematical model. Steady state conditions are reached approximately 1.5 h after the start of pumping. The CPU time to reach the steady state by implicit solution is approximately 30 min.

The paths followed by the floats in the physical model are compared with the results obtained from the mathematical model in Figures 5 and 6. The velocity distributions at the surface layer as obtained from the mathematical model are also shown. The average velocities

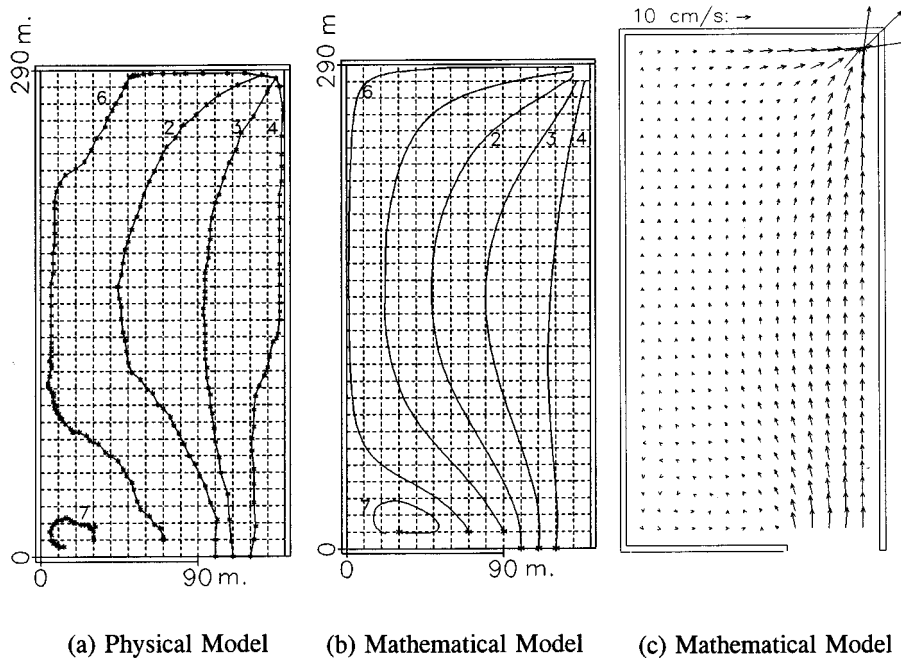


Figure 5. Float paths and velocity distributions at the surface layer, when the morning-glory is placed on the right end corner of the marina.



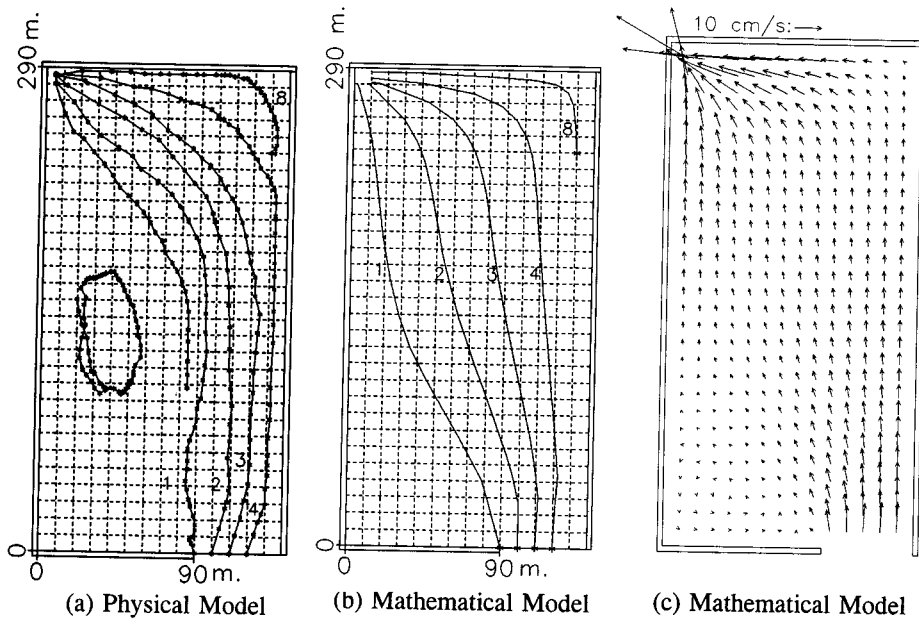


Figure 6. Float paths and velocity distributions at the surface layer, when the morning-glory is placed on the left end corner of the marina.

along the paths followed by the floats in both physical and mathematical models are compared in Table III.

In Case I, the intake structure is placed on the right end corner of the marina (135 × 285 m). The best agreement is obtained for the path followed by the float number 2 released at location

Table III. Average velocities along the float paths.

Float number	Release location (x, y) in m	Physical model average velocity (cm/s)		Mathematical model average velocity (cm/s)	
		Case I	Case II	Case I	Case II
1	90 × 0	**	4.04	**	4.15
2	100 × 0	3.64	5.27	3.66	4.51
3	110 × 0	3.61	5.16	3.88	5.04
4	120 × 0	3.72	3.88	3.91	4.01
5	130 × 0	**	**	**	**
6	70 × 10	2.24	**	1.98	**
7	30 × 10	0.93	**	0.70	**
8	130 × 240	**	2.72	**	3.51

(100 × 0 m) with an average velocity relative error of 0.55 per cent. The highest relative error in the average velocity is calculated as 24.4 per cent for the float number 7 which released at location (30 × 10 m). For this path, the velocities predicted by the numerical model are less than the velocities observed in the physical model. There is a dead zone where the released float does not move through the intake, since there occurs a gyre. Therefore, the float number 7 follows a counter clockwise elliptical route. The location of the dead zone is almost the same for physical and numerical models.

In Case II, the intake structure is placed on the left end corner of the marina (5 × 285 m). The closest agreement in the path trajectory is obtained for the float number 8, which is released at location (130 × 240 m). However, the relative error in its average velocity is the highest, 29 per cent. In the physical model, a dead zone is observed in the left-hand side of the basin towards the middle of the  $y$ -axis. On the other hand, the dead zone predicted by the numerical model is on the left entrance corner of the basin and it occupies a rather restricted area.

Periodical removal of surface water from a marina by pumping through a morning-glory type structure may be a feasible solution to enhance flushing of the basin as a remedy against water pollution. The pumping operation should be carried out during flood tide only to derive the greatest efficiency from such a system. The best location of the intake for removal of water and the optimum discharge rate are two crucial issues for designing such a scheme. The first question becomes especially important in the case of complex basins where the problems of 'dead regions' are normally more significant. Such features have traditionally been investigated in physical models. The three-dimensional numerical model developed is shown to be capable of predicting induced circulation patterns and the level of flushing enhancement with a very reasonable accuracy.

## 5. CONCLUSIONS

A comprehensive baroclinic three-dimensional numerical model of transport processes in coastal areas is developed. The model computes the full spatial distribution of velocities of unsteady flow induced by wind, tide or water density differences. The model predictions are verified by using several experimental and analytical results published in the literature [11,23,24].

The numerical solution method is a composite finite element–finite difference method. The governing equations are solved by the Galerkin weighted residual method in the vertical plane and by finite difference approximations in the horizontal plane. The water depths are divided into the same number of layers following the bottom topography. Grids can be concentrated near the surface, near the bottom layer or around any specified water depth by grid clustering. There may occur static instabilities in the numerical scheme when the ratio of layer thicknesses of successive nodal points is greater than 10. A higher horizontal mesh resolution should be used in such areas that have a sudden change in the bottom slope.

Since a non-orthogonal grid system is used, the length of the each element may differ from one point to another depending on the water depth. Therefore, interpolation functions have horizontal gradients. The integrals of the functions are performed, after all the horizontal gradient terms are corrected by considering the horizontal gradients of the interpolating functions. In the finite difference approximations, variable horizontal mesh sizes can be utilized. In shallow water body applications, the shoreline boundaries can be treated as movable by taking into account the flooding and drying processes, resulting from water level changes. The stability of the numerical scheme is assured by the implicit Crank–Nicholson method.

The numerical model is shown to be capable of predicting the induced circulation patterns and the level of flushing enhancement in a marina with reasonable accuracy. The sink flow, which has significant velocity gradients is successfully generated. The induced circulation patterns seem to be more sensitive to the horizontal exchange. The application of the standard isotropic  $k$ – $\epsilon$  model fails to estimate the horizontal eddy viscosity coefficients, since the horizontal length scale is much larger than the water depth. On contrast, the Smagorinsky sub/grid scale turbulence model, provides better agreement with the measurements.

The numerical model can serve as a powerful design tool and can be implemented in a Decision Support System. It may be used in diverse coastal applications including the induced circulation patterns, the prediction of natural flushing rates caused by tidal motion, wind effect, a water inflow due to longshore currents driven by breaking waves and fresh water inflow in the case of a river marina.

#### APPENDIX A

If open boundary is parallel to the  $x$ -axis, then, horizontal velocities along the  $y$ -axis are computed in the middle of the cell where the water depth is known. Similar treatment is performed for the horizontal velocities along the  $x$ -axis, if the open boundary is parallel to the  $y$ -axis. Finite difference formulations of the typical terms are given below.

$$\left(\frac{\partial v}{\partial y}\right)_{i,1} = \frac{-(\Delta y_1 + \Delta y_2) v_{i,1}}{\frac{\Delta y_1}{2} \left(\frac{\Delta y_1}{2} + \Delta y_2\right)} + \frac{\left(\frac{\Delta y_1}{2} + \Delta y_2\right) v_{i,2}}{\frac{\Delta y_1}{2} \Delta y_2} - \frac{\frac{\Delta y_1}{2} v_{i,3}}{\Delta y_2 \left(\frac{\Delta y_1}{2} + \Delta y_2\right)} \quad (\text{A.1})$$

$$\left(\frac{\partial^2 v}{\partial y^2}\right)_{i,1} = 2 \left( \frac{v_{i,1}}{\frac{\Delta y_1}{2} \left(\frac{\Delta y_1}{2} + \Delta y_2\right)} - \frac{v_{i,2}}{\frac{\Delta y_1}{2} \Delta y_2} + \frac{v_{i+1,3}}{\frac{\Delta y_1}{2} \left(\frac{\Delta y_1}{2} + \Delta y_2\right)} \right) \quad (\text{A.2})$$

$$\begin{aligned} \left(\frac{\partial u}{\partial y}\right)_{i+1/2,1} &= \frac{-u_{i+1/2,1} \left(\frac{3\Delta y_2}{2} + \frac{\Delta y_3}{2} + \Delta y_1\right) + u_{i+1/2,2} \left(\Delta y_2 + \frac{\Delta y_1 + \Delta y_3}{2}\right)}{\frac{(\Delta y_1 + \Delta y_2)}{2} \left(\Delta y_2 + \frac{\Delta y_1 + \Delta y_3}{2}\right)} + \frac{u_{i+1/2,3} \left(\frac{\Delta y_1 + \Delta y_2}{2}\right)}{\frac{(\Delta y_1 + \Delta y_2)}{2} \left(\Delta y_2 + \frac{\Delta y_1 + \Delta y_3}{2}\right)} \\ &\quad - \frac{u_{i+1/2,1} \left(\frac{\Delta y_1 + \Delta y_2}{2}\right)}{\frac{(\Delta y_1 + \Delta y_2)}{2} \left(\Delta y_2 + \frac{\Delta y_1 + \Delta y_3}{2}\right)} \end{aligned} \quad (\text{A.3})$$

where  $u_{i+1/2,m}$  ( $m = 1, 2, 3$ ) can be approximated as:

$$u_{i+1/2,m} = \frac{u_{i-1,m} \frac{(\Delta x_i)^2}{4}}{\Delta x_{i-1} (\Delta x_i + \Delta x_{i-1})} + \frac{u_{i,m} \left(\frac{\Delta x_i}{2} + \Delta x_{i-1}\right)}{2 \Delta x_{i-1}} + \frac{u_{i+1,m} \left(\frac{\Delta x_i}{2} + \Delta x_{i-1}\right)}{2 (\Delta x_i + \Delta x_{i+1})} \quad (\text{A.4})$$

$$\left(\frac{\partial u}{\partial x}\right)_{i+1/2,1} = \frac{u_{i+1,1} - u_{i,1}}{\Delta x_i} \quad (\text{A.5})$$

$$\left(\frac{\partial v}{\partial x}\right)_{i,1} = \frac{(\Delta x_i + \Delta x_{i+1})(v_{i,1} - v_{i-1,1})}{(\Delta x_i + \Delta x_{i-1}) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} + \frac{(\Delta x_i + \Delta x_{i-1})(v_{i+1,1} - v_{i,1})}{(\Delta x_i + \Delta x_{i+1}) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \quad (\text{A.6})$$

$$\begin{aligned} \left(\frac{\partial^2 v}{\partial x^2}\right)_{i,1} &= 2 \left( \frac{v_{i-1,1}}{\left(\frac{\Delta x_{i-1} + \Delta x_i}{2}\right) \left(\Delta x_i + \frac{\Delta x_{i-1} + \Delta x_{i+1}}{2}\right)} - \frac{v_{i,1}}{\left(\frac{\Delta x_i + \Delta x_{i+1}}{2}\right) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \right) \\ &\quad + \frac{v_{i+1,1}}{\left(\frac{\Delta x_i + \Delta x_{i+1}}{2}\right) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \end{aligned} \quad (\text{A.7})$$

## APPENDIX B

### B.1. Finite element approximations

The length of the each element may differ from one point to another depending on the water depth, since the non-orthogonal grid system is used. Therefore, the interpolation functions have horizontal gradients. Since the treatment is similar for the  $y$ -direction, the horizontal derivatives of interpolation functions are solely listed below for the  $x$ -direction:

$$\frac{\partial N_1}{\partial x} = \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial x} + \frac{z}{l_k^2} \frac{\partial l_k}{\partial x} \tag{B.1}$$

$$\frac{\partial N_2}{\partial x} = -\frac{\partial N_1}{\partial x}, \quad \frac{\partial^2 N_2}{\partial x^2} = -\frac{\partial^2 N_1}{\partial x^2} \tag{B.2}$$

$$\frac{\partial^2 N_1}{\partial x^2} = \frac{z_2}{l_k^2} \frac{\partial^2 z_1}{\partial x^2} - \frac{z_1}{l_k^2} \frac{\partial^2 z_2}{\partial x^2} + \frac{2z_1}{l_k^3} \frac{\partial z_2}{\partial x} - \frac{2z_2}{l_k^3} \frac{\partial l_k}{\partial x} \frac{\partial z_1}{\partial x} - \frac{2z}{l_k^3} \left(\frac{\partial l_k}{\partial x}\right)^2 \frac{\partial^2 l_k}{\partial x^2} \tag{B.3}$$

$$\begin{aligned} \frac{\partial}{\partial y} \left( \frac{\partial N_1}{\partial x} \right) &= \frac{z_2}{l_k^2} \frac{\partial}{\partial y} \left( \frac{\partial z_1}{\partial x} \right) + \frac{\partial z_1}{\partial x} \left( \frac{1}{l_k^2} \frac{\partial z_2}{\partial y} - \frac{2z_2}{l_k^3} \frac{\partial l_k}{\partial y} \right) - \frac{z_1}{l_k^2} \frac{\partial}{\partial y} \left( \frac{\partial z_2}{\partial x} \right) - \frac{\partial z_2}{\partial x} \left( \frac{1}{l_k^2} \frac{\partial z_1}{\partial y} - \frac{2z_1}{l_k^3} \frac{\partial l_k}{\partial y} \right) \\ &\quad - \frac{2z}{l_k^3} \frac{\partial l_k}{\partial x} \frac{\partial l_k}{\partial y} + \frac{z}{l_k^2} \frac{\partial}{\partial y} \left( \frac{\partial l_k}{\partial x} \right) \end{aligned} \tag{B.4}$$

The Galerkin method is utilized in order to develop the element equations. The approximate solutions, are substituted into the related equations. Then, the residuals (R) are minimized by the Galerkin method which uses the interpolation functions as the weighting functions. Application of the Galerkin method is shown below on some of the terms, where  $\tilde{G}$ ,  $\tilde{F}$  and  $\tilde{E}$  represent the variables of concern, as:

$$\int_{z_1}^{z_2} N_1 \tilde{G} \, dz = l_k \left( \frac{G_1^k}{3} + \frac{G_2^k}{6} \right); \quad \int_{z_1}^{z_2} N_2 \tilde{G} \, dz = l_k \left( \frac{G_1}{6} + \frac{G_2}{3} \right) \tag{B.5}$$

$$\int_{z_1}^{z_2} N_1 \frac{\partial \tilde{G}}{\partial x} \, dz = \frac{l_k}{3} \frac{\partial G_1^k}{\partial x} + \frac{l_k}{6} \frac{\partial G_2^k}{\partial x} + \frac{1}{6} \frac{\partial l_k}{\partial x} (G_1^k - G_2^k) + \frac{l_k}{2} \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial x} \right) (G_1^k - G_2^k) \tag{B.6}$$

$$\int_{z_1}^{z_2} N_2 \frac{\partial \tilde{G}}{\partial x} \, dz = \frac{l_k}{6} \frac{\partial G_1^k}{\partial x} + \frac{l_k}{3} \frac{\partial G_2^k}{\partial x} + \frac{1}{3} \frac{\partial l_k}{\partial x} (G_1^k - G_2^k) + \frac{l_k}{2} \left[ \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial x} \right] (G_1^k - G_2^k) \tag{B.7}$$

$$\begin{aligned} \int_{z_1}^{z_2} N_1 \tilde{F} \frac{\partial \tilde{G}}{\partial x} \, dz &= \left( \frac{\partial l_k}{\partial x} \frac{(G_1^k - G_2^k)}{12} + \frac{l_k}{12} \frac{\partial G_2^k}{\partial x} \right) (F_1^k + F_2^k) + \frac{\partial G_1^k}{\partial x} \left( \frac{l_k}{12} F_2^k + \frac{l_k}{4} F_1^k \right) \\ &\quad + \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial x} \right) (G_1^k - G_2^k) \left( \frac{F_1^k}{3} + \frac{F_2^k}{6} \right) \end{aligned} \tag{B.8}$$

$$\begin{aligned} \int_{z_1}^{z_2} N_2 \tilde{F} \frac{\partial \tilde{G}}{\partial x} \, dz &= \left( \frac{\partial l_k}{\partial x} (G_1^k - G_2^k) + l_k \frac{\partial G_2^k}{\partial x} \right) \left( \frac{F_1^k}{12} + \frac{F_2^k}{4} \right) + \frac{l_k}{12} \frac{\partial G_1^k}{\partial x} (F_2^k + F_1^k) \\ &\quad + \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial x} \right) (G_1^k - G_2^k) \left( \frac{F_1^k}{6} + \frac{F_2^k}{3} \right) \end{aligned} \tag{B.9}$$

$$\int_{z_1}^{z_2} N_1 \frac{\partial}{\partial z} \left( \tilde{F} \frac{\partial \tilde{G}}{\partial z} \right) \, dz = -\frac{(F_1^k + F_2^k)}{2l_k} G_1^k + \frac{(F_1^k + F_2^k)}{2l_k} G_2^k - F_1^k \frac{\partial G_1^k}{\partial z} \tag{B.10}$$

$$\int_{z_1}^{z_2} N_2 \frac{\partial}{\partial z} \left( \tilde{F} \frac{\partial \tilde{G}}{\partial z} \right) dz = \frac{(F_1^k + F_2^k)}{2l_k} G_1^k - \frac{(F_1^k + F_2^k)}{2l_k} G_2^k + F_2^k \frac{\partial G_1^k}{\partial z} \quad (\text{B.11})$$

$$\int_{z_1}^{z_2} N_1 \tilde{F} \frac{\partial \tilde{G}}{\partial z} dz = \left( \frac{F_1^k}{3} + \frac{F_2^k}{6} \right) (G_2^k - G_1^k); \quad \int_{z_1}^{z_2} N_2 \tilde{F} \frac{\partial \tilde{G}}{\partial z} dz = \left( \frac{F_1^k}{6} + \frac{F_2^k}{3} \right) (G_2^k - G_1^k) \quad (\text{B.12})$$

$$\begin{aligned} \int_{z_1}^{z_2} N_2 \frac{\partial \tilde{F}}{\partial y} \frac{\partial \tilde{G}}{\partial x} dz &= \frac{\partial F_1^k}{\partial y} \left( \frac{\partial G_1^k}{\partial x} \frac{l_k}{12} + \frac{\partial G_2^k}{\partial x} \frac{l_k}{12} + \frac{\partial l_k}{\partial x} \frac{G_1^k - G_2^k}{12} \right) \\ &+ F_1^k \left( \frac{\partial l_k}{\partial y} \left( \frac{1}{12} \frac{\partial G_1^k}{\partial x} + \frac{1}{4} \frac{\partial G_2^k}{\partial x} \right) + \frac{\partial l_k}{\partial y} \frac{\partial l_k}{\partial x} \frac{G_1^k - G_2^k}{4l_k} \right) \\ &+ \frac{\partial F_2^k}{\partial y} \left( \frac{\partial G_1^k}{\partial x} \frac{l_k}{12} + \frac{\partial G_2^k}{\partial x} \frac{l_k}{4} + \frac{\partial l_k}{\partial x} \frac{(G_1^k - G_2^k)}{4} \right) \\ &+ F_2^k \left( -\frac{\partial l_k}{\partial y} \left( \frac{1}{12} \frac{\partial G_1^k}{\partial x} + \frac{1}{4} \frac{\partial G_2^k}{\partial x} \right) + \frac{\partial l_k}{\partial y} \frac{\partial l_k}{\partial x} \frac{(G_2^k - G_1^k)}{4l_k} \right) \\ &+ \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial x} \right) \left( \frac{1}{6} \frac{\partial F_1^k}{\partial y} + \frac{1}{3} \frac{\partial F_2^k}{\partial y} \right) (G_1^k - G_2^k) \\ &+ \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial y} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial y} \right) \left( \frac{1}{6} \frac{\partial G_1^k}{\partial x} - \frac{1}{3} \frac{\partial G_2^k}{\partial x} \right) (F_1^k - F_2^k) \\ &+ \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial x} \right) \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial y} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial y} \right) \frac{(G_1^k - G_2^k)}{2} (F_1^k - F_2^k) \\ &+ \left( \frac{\partial l_k}{\partial y} \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial x} \right) + \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial y} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial y} \right) \frac{\partial l_k}{\partial x} \right) \frac{G_1^k - G_2^k}{3} (F_1^k - F_2^k) \end{aligned} \quad (\text{B.13})$$

$$\begin{aligned} \int_{z_1}^{z_2} N_1 \frac{\partial \tilde{F}}{\partial y} \frac{\partial \tilde{G}}{\partial x} dz &= \frac{\partial F_1^k}{\partial y} \left( \frac{\partial G_1^k}{\partial x} \frac{l_k}{4} + \frac{\partial G_2^k}{\partial x} \frac{l_k}{12} + \frac{\partial l_k}{\partial x} \frac{G_1^k - G_2^k}{12} \right) \\ &+ F_1^k \left( \frac{1}{12} \frac{\partial l_k}{\partial y} \left( \frac{\partial G_1^k}{\partial x} + \frac{\partial G_2^k}{\partial x} \right) + \frac{\partial l_k}{\partial y} \frac{\partial l_k}{\partial x} \frac{(G_1^k - G_2^k)}{12l_k} \right) \\ &+ \frac{\partial F_2^k}{\partial y} \left( \frac{\partial G_1^k}{\partial x} \frac{l_k}{12} + \frac{\partial G_2^k}{\partial x} \frac{l_k}{12} + \frac{\partial l_k}{\partial x} \frac{(G_1^k - G_2^k)}{12} \right) \\ &+ F_2^k \left( -\frac{1}{12} \frac{\partial l_k}{\partial y} \left( \frac{\partial G_1^k}{\partial x} + \frac{\partial G_2^k}{\partial x} \right) + \frac{\partial l_k}{\partial y} \frac{\partial l_k}{\partial x} \frac{(G_2^k - G_1^k)}{12l_k} \right) \\ &+ \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial x} \right) \left( \frac{1}{3} \frac{\partial F_1^k}{\partial y} - \frac{1}{6} \frac{\partial F_2^k}{\partial y} \right) (G_1^k - G_2^k) \\ &+ \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial y} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial y} \right) \left( \frac{1}{3} \frac{\partial G_1^k}{\partial x} - \frac{1}{6} \frac{\partial G_2^k}{\partial x} \right) (F_1^k - F_2^k) \\ &+ \left( \frac{z_2}{l_k} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k} \frac{\partial z_2}{\partial x} \right) \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial y} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial y} \right) \frac{(G_1^k - G_2^k)}{2} (F_1^k - F_2^k) \end{aligned}$$

$$+ \left( \frac{\partial l_k}{\partial y} \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial x} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial x} \right) + \left( \frac{z_2}{l_k^2} \frac{\partial z_1}{\partial y} - \frac{z_1}{l_k^2} \frac{\partial z_2}{\partial y} \right) \frac{\partial l_k}{\partial x} \right) \frac{(G_1^k - G_2^k)}{6} (F_1^k - F_2^k) \tag{B.14}$$

The time derivative terms are approximated by:

$$\int_{z_1}^{z_2} N_1 \frac{\partial \tilde{G}}{\partial t} dz = \frac{l_k}{3} \frac{\partial G_1^k}{\partial t} + \frac{l_k}{6} \frac{\partial G_2^k}{\partial t} \tag{B.15}$$

$$\int_{z_1}^{z_2} N_2 \frac{\partial \tilde{G}}{\partial t} dz = \frac{l_k}{6} \frac{\partial G_1^k}{\partial t} + \frac{l_k}{3} \frac{\partial G_2^k}{\partial t} \tag{B.16}$$

*B.2. Finite difference approximations*

After the application of the Galerkin method, any derivative terms with respect to  $x$  and  $y$ , appearing in the equations are replaced by the central finite difference expressions, by paying attention to the possibility of unequal mesh sizes. Finite difference formulations of differential terms are prepared for the staggered schemes.

*B.2.1. Horizontal velocity component in the  $x$ -direction.* In the computations of the velocity component  $u$ , the finite difference formulations of differential terms are prepared for the staggered scheme shown in Figure B1.

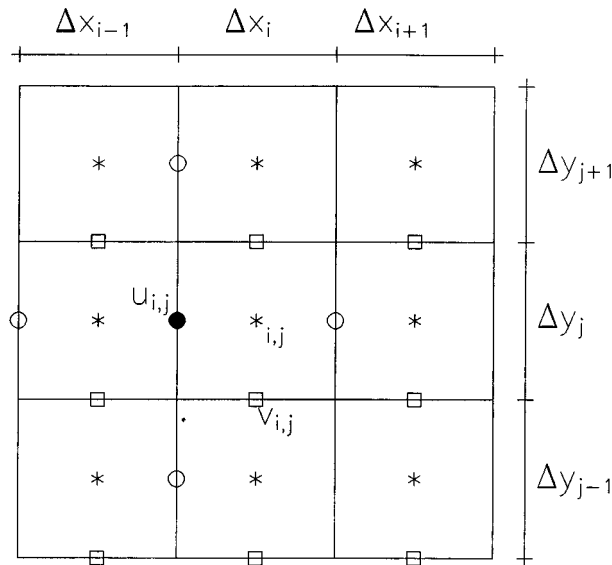


Figure B1. Finite difference molecule for computations of the horizontal velocity in the  $x$ -direction,  $u$ . (Symbols represent the variables as follows:  $u$   $\bullet$ ,  $v$   $\circ$ , all other variables  $*$ .)

The following approximations can be derived in respect to the computational molecule illustrated in Figure B1. It should be noted that, all computations are performed at point  $\bullet_{i,j}$ :

$$\left(\frac{\partial u}{\partial x}\right)_{i,j} = \frac{\Delta x_i(u_{i,j} - u_{i-1,j})}{\Delta x_{i-1}(\Delta x_i + \Delta x_{i-1})} + \frac{\Delta x_{i-1}(u_{i+1,j} - u_{i,j})}{\Delta x_i(\Delta x_i + \Delta x_{i-1})} \quad (\text{B.17})$$

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_{i,j} = 2\left(\frac{u_{i-1,j}}{\Delta x_{i-1}(\Delta x_i + \Delta x_{i-1})} - \frac{u_{i,j}}{\Delta x_{i-1}\Delta x_i} + \frac{u_{i+1,j}}{\Delta x_i(\Delta x_i + \Delta x_{i-1})}\right) \quad (\text{B.18})$$

$$\begin{aligned} \left(\frac{\partial^2 u}{\partial y^2}\right)_{i,j} = 2\left(\frac{u_{i,j-1}}{\left(\frac{\Delta y_{j-1} + \Delta y_j}{2}\right)\left(\Delta y_j + \frac{\Delta y_{j-1} + \Delta y_{j+1}}{2}\right)} - \frac{u_{i,j}}{\frac{(\Delta y_j + \Delta y_{j+1})(\Delta y_j + \Delta y_{j-1})}{2}} \right. \\ \left. + \frac{u_{i,j+1}}{\frac{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)}{2}}\right) \end{aligned} \quad (\text{B.19})$$

$$\left(\frac{\partial u}{\partial y}\right)_{i,j} = \frac{(\Delta y_j + \Delta y_{j+1})(u_{i,j} - u_{i,j-1})}{(\Delta y_j + \Delta y_{j-1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} + \frac{(\Delta y_j + \Delta y_{j-1})(u_{i,j+1} - u_{i,j})}{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_i + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} \quad (\text{B.20})$$

$$\begin{aligned} \left(\frac{\partial v}{\partial x}\right)_{i-1/2,j+1/2} = \frac{-v_{i-1,j+1/2}\left(\frac{3\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2}\right)}{\frac{(x_i + x_{i-1})}{2}\left(x_i + \frac{x_{i-1} + x_{i+1}}{2}\right)} + \frac{v_{i,j+1/2}\left(\Delta x_i + \frac{\Delta x_{i+1} - \Delta x_{i-1}}{2}\right)}{\frac{(\Delta x_i + \Delta x_{i-1})\Delta x_i + \Delta x_{i+1}}{2}} \\ + \frac{v_{i+1,j+1/2}\left(\frac{\Delta x_{i-1} - \Delta x_i}{2}\right)}{\frac{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)}{2}} \end{aligned} \quad (\text{B.21})$$

where  $v_{m,j+1/2}$  ( $m = i-1, i, i+1$ ) can be estimated by the following equation:

$$v_{m,j+1/2} = \frac{v_{m,j-1}\frac{(\Delta y_j)^2}{4}}{\Delta y_{j-1}(\Delta y_j + \Delta y_{j-1})} + \frac{v_{m,j}\left(\frac{\Delta y_j}{2} + \Delta y_{j-1}\right)}{2\Delta y_{j-1}} + \frac{v_{m,j+1}\left(\frac{\Delta y_j}{2} + \Delta y_{j-1}\right)}{2(\Delta y_j + \Delta y_{j-1})} \quad (\text{B.22})$$

$$\left(\frac{\partial v}{\partial y}\right)_{i-1/2,j+1/2} = \frac{v_{i-1/2,j+1} - v_{i-1/2,j}}{\Delta y_j} \quad (\text{B.23})$$

where  $v_{i-1/2,m}$  ( $m = j, j+1$ ) can be estimated as:



$$v_{i-1/2,m} = \frac{v_{i-1,m} \Delta x_i \left( \Delta x_i + \frac{\Delta x_{i+1}}{2} \right)}{(\Delta x_{i-1} + \Delta x_i) \left( \Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2} \right)} + \frac{v_{i,m} \Delta x_{i-1} \left( \Delta x_i + \frac{\Delta x_{i+1}}{2} \right)}{(\Delta x_i + \Delta x_{i-1}) \left( \frac{\Delta x_i + \Delta x_{i+1}}{2} \right)} - \frac{v_{i+1,m} \Delta x_{i-1} \left( \frac{\Delta x_i}{2} \right)}{(\Delta x_i + \Delta x_{i+1}) \left( \Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2} \right)} \tag{B.24}$$

$$\left( \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) \right)_{i-1/2,j+1/2} = \frac{\left. \frac{\partial v}{\partial x} \right|_{i-1/2,j+1} - \left. \frac{\partial v}{\partial x} \right|_{i-1/2,j}}{\Delta y_{i,j}} \tag{B.25}$$

Therefore, Equation (B.54) can be written as:

$$\begin{aligned} & \left( \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) \right)_{i-1/2,j+1/2} \\ &= \frac{(v_{i-1,j} - v_{i-1,j+1}) \left( \frac{3\Delta x_i}{2} + \frac{\Delta x_{i+1}}{2} \right)}{\frac{(x_i + x_{i-1})}{2} \left( x_i + \frac{x_{i-1} + x_{i+1}}{2} \right) \Delta y_j} + \frac{(v_{i,j+1} - v_{i,j}) \left( \Delta x_i + \frac{\Delta x_{i+1} - \Delta x_{i-1}}{2} \right)}{\frac{(\Delta x_i + \Delta x_{i-1})}{2} \frac{(\Delta x_i + \Delta x_{i+1})}{2} \Delta y_j} \\ &+ \frac{(v_{i+1,j+1} - v_{i+1,j}) \left( \frac{\Delta x_{i-1} - \Delta x_i}{2} \right)}{\frac{(\Delta x_i + \Delta x_{i+1})}{2} \left( \Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2} \right) \Delta y_j} \end{aligned} \tag{B.26}$$

The horizontal derivatives of the element length ( $l_k$ ) can be approximated as:

$$\begin{aligned} \left( \frac{\partial l}{\partial x} \right)_{i-1/2,j} &= \frac{-l_{i-1,j} \left( \frac{\Delta x_i}{2} + \Delta x_i + \frac{\Delta x_{i+1}}{2} \right)}{\frac{(x_i + x_{i-1})}{2} \left( x_i + \frac{x_{i-1} + x_{i+1}}{2} \right)} + \frac{l_{i,j} \left( \Delta x_i + \frac{\Delta x_{i+1} - \Delta x_{i-1}}{2} \right)}{\frac{(\Delta x_i + \Delta x_{i-1})}{2} \frac{\Delta x_i + \Delta x_{i+1}}{2}} \\ &+ \frac{l_{i+1,j} \left( \frac{\Delta x_{i-1} - \Delta x_i}{2} \right)}{\frac{(\Delta x_i + \Delta x_{i+1})}{2} \left( \Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2} \right)} \end{aligned} \tag{B.27}$$

$$\left(\frac{\partial l}{\partial y}\right)_{i-1/2,j} = \frac{(\Delta y_j + \Delta y_{j+1})(l_{i-1/2,j} - l_{i-1/2,j-1})}{(\Delta y_j + \Delta y_{j-1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} + \frac{(\Delta y_j + \Delta y_{j-1})(l_{i-1/2,j+1} - l_{i-1/2,j})}{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} \tag{B.28}$$

where  $l_{i-1/2,m}$  ( $m = j - 1, j, j + 1$ ) can be approximated as:

$$l_{i-1/2,m} = \frac{l_{i-1,m} \Delta x_i \left(\Delta x_i + \frac{\Delta x_{i+1}}{2}\right)}{(\Delta x_{i-1} + \Delta x_i) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} + \frac{l_{i,m} \Delta x_{i-1} \left(\Delta x_i + \frac{\Delta x_{i+1}}{2}\right)}{(\Delta x_i + \Delta x_{i-1}) \left(\frac{\Delta x_i + \Delta x_{i+1}}{2}\right)} - \frac{l_{i+1,m} \Delta x_{i-1} \left(\frac{\Delta x_i}{2}\right)}{(\Delta x_i + \Delta x_{i+1}) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \tag{B.29}$$

$$\left(\frac{\partial}{\partial y} \left(\frac{\partial l}{\partial x}\right)\right)_{i-1/2,j} = \frac{(\Delta y_j + \Delta y_{j+1}) \left(\left(\frac{\partial l}{\partial x}\right)_{i-1/2,j} - \left(\frac{\partial l}{\partial x}\right)_{i-1/2,j-1}\right)}{(\Delta y_j + \Delta y_{j-1}) \left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} + \frac{(\Delta y_j + \Delta y_{j-1}) \left(\left(\frac{\partial l}{\partial x}\right)_{i-1/2,j+1} - \left(\frac{\partial l}{\partial x}\right)_{i-1/2,j}\right)}{(\Delta y_j + \Delta y_{j+1}) \left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} \tag{B.30}$$

**B.2.2. The horizontal velocity component in the y-direction.** In the computations of the velocity component  $v$ , the finite difference formulations of differential terms are prepared for the staggered scheme shown in Figure B2. The following approximations can be derived in respect to the computational molecule illustrated in Figure B2. It should be noted that, all computations are performed at point  $\blacksquare_{i,j}$ :

$$\left(\frac{\partial v}{\partial y}\right)_{i,j} = \frac{\Delta y_j (v_{i,j} - v_{i,j-1})}{\Delta y_{j-1} (\Delta y_j + \Delta y_{j-1})} + \frac{\Delta y_{j-1} (v_{i,j+1} - v_{i,j})}{\Delta y_j (\Delta y_j + \Delta y_{j-1})} \tag{B.31}$$

$$\left(\frac{\partial^2 v}{\partial y^2}\right)_{i,j} = 2 \left( \frac{v_{i,j-1}}{\Delta y_{j-1} (\Delta y_j + \Delta y_{j-1})} - \frac{v_{i,j}}{\Delta y_{j-1} \Delta y_j} + \frac{v_{i+1,j}}{\Delta y_j (\Delta y_j + \Delta y_{j-1})} \right) \tag{B.32}$$

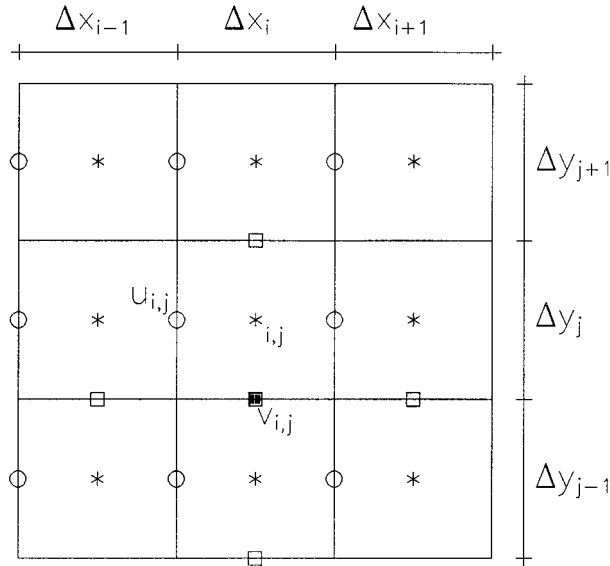


Figure B2. Finite difference molecule for computations of the horizontal velocity in the  $y$ -direction,  $v$ . Symbols represent the variables as follows:  $v$   $\square$ ,  $u$   $\circ$ , all other variables  $*$ .

$$\left(\frac{\partial^2 v}{\partial x^2}\right)_{i,j} = 2 \left( \frac{v_{i-1,j}}{\left(\frac{\Delta x_{i-1} + \Delta x_i}{2}\right) \left(\Delta x_i + \frac{\Delta x_{i-1} + \Delta x_{i+1}}{2}\right)} - \frac{v_{i,j}}{\frac{\Delta x_i + \Delta x_{i+1}}{2} \frac{\Delta x_i + \Delta x_{i-1}}{2}} + \frac{v_{i+1,j}}{\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right) \left(\frac{\Delta x_i + \Delta x_{i+1}}{2}\right)} \right) \tag{B.33}$$

$$\left(\frac{\partial v}{\partial x}\right)_{i,j} = \frac{(\Delta x_i + \Delta x_{i+1})(v_{i,j} - v_{i-1,j})}{(\Delta x_i + \Delta x_{i-1}) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} + \frac{(\Delta x_i + \Delta x_{i-1})(v_{i+1,j} - v_{i,j})}{(\Delta x_i + \Delta x_{i+1}) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \tag{B.34}$$

$$\left(\frac{\partial u}{\partial y}\right)_{i+1/2,j-1/2} = \frac{-u_{i+1/2,j-1} \left(\frac{3\Delta y_j}{2} + \frac{\Delta y_{j+1}}{2}\right)}{\frac{\Delta y_j + \Delta y_{j-1}}{2} \left(\Delta y_j + \frac{\Delta y_{j-1} + \Delta y_{j+1}}{2}\right)} + \frac{u_{i+1/2,j} \left(\Delta y_j + \frac{\Delta y_{j+1} - \Delta y_{j-1}}{2}\right)}{\frac{\Delta y_j + \Delta y_{j-1}}{2} \frac{\Delta y_j + \Delta y_{j+1}}{2}}$$

$$+ \frac{u_{i+1/2,j+1} \left( \frac{\Delta y_{j-1} - \Delta y_j}{2} \right)}{\frac{(\Delta y_j + \Delta y_{j+1}) \left( \Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2} \right)}{2}} \quad (\text{B.35})$$

where  $u_{i+1/2,m}$  ( $m = j-1, j, j+1$ ) can be approximated as:

$$u_{i+1/2,m} = \frac{u_{i-1,m} \frac{(\Delta x_i)^2}{4}}{\Delta x_{i-1} (\Delta x_i + \Delta x_{i-1})} + \frac{u_{i,m} \left( \frac{\Delta x_i}{2} + \Delta x_{i-1} \right)}{2(\Delta x_{i-1})} + \frac{u_{i+1,m} \left( \frac{\Delta x_i}{2} + \Delta x_{i-1} \right)}{2(\Delta x_i + \Delta x_{i+1})} \quad (\text{B.36})$$

$$\left( \frac{\partial u}{\partial x} \right)_{i+1/2,j-1/2} = \frac{u_{i+1,j-1/2} - u_{i,j-1/2}}{\Delta x_i} \quad (\text{B.37})$$

where  $u_{m,j-1/2}$  ( $m = i, i+1$ ) can be approximated as:

$$u_{m,j-1/2} = \frac{u_{m,j-1} \Delta y_j \left( \Delta y_j + \frac{\Delta y_{j+1}}{2} \right)}{(\Delta y_{j-1} + \Delta y_j) \left( \Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2} \right)} + \frac{u_{m,j} \Delta y_{j-1} \left( \Delta y_j + \frac{\Delta y_{j+1}}{2} \right)}{(\Delta y_j + \Delta y_{j-1}) \left( \frac{\Delta y_j + \Delta y_{j+1}}{2} \right)} - \frac{u_{m,j+1} \Delta y_{j-1} \left( \frac{\Delta y_j}{2} \right)}{(\Delta y_j + \Delta y_{j+1}) \left( \Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2} \right)} \quad (\text{B.38})$$

$$\left( \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial y} \right) \right)_{i+1/2,j-1/2} = \frac{\left( \frac{\partial u}{\partial y} \right)_{i+1,j-1/2} - \left( \frac{\partial u}{\partial y} \right)_{i,j-1/2}}{\Delta x_i} \quad (\text{B.39})$$

Therefore, Equation (B.68) can be written as:

$$\begin{aligned} & \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial y} \right)_{i+1/2,j-1/2} \\ &= \frac{(u_{i,j-1} - u_{i+1,j-1})(3\Delta y_j + \Delta y_{j+1})}{(\Delta y_j + \Delta y_{j-1}) \left( \Delta y_j + \frac{\Delta y_{j-1} + \Delta y_{j+1}}{2} \right) \Delta x_i} + \frac{(u_{i+1,j} - u_{i,j}) \left( \Delta y_j + \frac{\Delta y_{j+1} - \Delta y_{j-1}}{2} \right)}{\frac{(\Delta y_j + \Delta y_{j-1}) (\Delta y_j + \Delta y_{j+1})}{2} \Delta x_i} \\ &+ \frac{(u_{i+1,j+1} - u_{i,j+1})(\Delta y_{j-1} - \Delta y_j)}{(\Delta y_j + \Delta y_{j+1}) \left( \Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2} \right) \Delta x_i} \end{aligned} \quad (\text{B.40})$$

The horizontal derivatives of the element length ( $l_k$ ) can be approximated as:

$$\begin{aligned} \left(\frac{\partial l}{\partial y}\right)_{i,j-1/2} &= \frac{-l_{i,j-1}\left(\frac{3\Delta y_j + \Delta y_{j+1}}{2}\right)}{\frac{(y_j + y_{j-1})\left(y_j + \frac{y_{j-1} + y_{j+1}}{2}\right)}{2}} + \frac{l_{i,j}\left(\Delta y_j + \frac{\Delta y_{j+1} - \Delta y_{j-1}}{2}\right)}{\frac{(\Delta y_j + \Delta y_{j-1})\left(\Delta y_j + \Delta y_{j+1}\right)}{2}} \\ &\quad + \frac{l_{i,j+1}\left(\frac{\Delta y_{j-1} - \Delta y_j}{2}\right)}{\frac{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)}{2}} \end{aligned} \tag{B.41}$$

$$\begin{aligned} \left(\frac{\partial^2 l}{\partial y^2}\right)_{i,j-1/2} &= 2\left(\frac{l_{i,j-1}}{\frac{(\Delta y_{j-1} + \Delta y_j)\left(\Delta y_j + \frac{\Delta y_{j-1} + \Delta y_{j+1}}{2}\right)}{2}} - \frac{l_{i,j}}{\frac{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \Delta y_{j-1}\right)}{2}}\right. \\ &\quad \left. + \frac{l_{i,j+1}}{\frac{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)}{2}}\right) \end{aligned} \tag{B.42}$$

$$\begin{aligned} \left(\frac{\partial l}{\partial x}\right)_{i,j-1/2} &= \frac{(\Delta x_i + \Delta x_{i+1})(l_{i,j-1/2} - l_{i-1,j-1/2})}{(\Delta x_i + \Delta x_{i-1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} + \frac{(\Delta x_i + \Delta x_{i-1})(l_{i+1,j-1/2} - l_{i,j-1/2})}{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \end{aligned} \tag{B.43}$$

$$\begin{aligned} \left(\frac{\partial^2 l}{\partial x^2}\right)_{i,j-1/2} &= 2\left(\frac{l_{i-1,j-1/2}}{\frac{(\Delta x_{i-1} + \Delta x_i)\left(\Delta x_i + \frac{\Delta x_{i-1} + \Delta x_{i+1}}{2}\right)}{2}} - \frac{l_{i,j-1/2}}{\frac{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \Delta x_{i-1}\right)}{2}}\right. \\ &\quad \left. + \frac{l_{i+1,j-1/2}}{\frac{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)}{2}}\right) \end{aligned} \tag{B.44}$$

where  $l_{m,j-1/2}$ , ( $m = i - 1, i, i + 1$ ) can be approximated as:

$$l_{m,j-1/2} = \frac{l_{m,j-1} \Delta y_j \left(\Delta y_j + \frac{\Delta y_{j+1}}{2}\right)}{(\Delta y_{j-1} + \Delta y_j) \left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} + \frac{l_{m,j} \Delta y_{j-1} \left(\Delta y_j + \frac{\Delta y_{j+1}}{2}\right)}{(\Delta y_j + \Delta y_{j-1}) \left(\frac{\Delta y_j + \Delta y_{j+1}}{2}\right)}$$

$$-\frac{l_{m,j+1} \Delta y_{j-1} \left(\frac{\Delta y_j}{2}\right)}{(\Delta y_j + \Delta y_{j+1}) \left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} \tag{B.45}$$

$$\begin{aligned} \left(\frac{\partial}{\partial x} \left(\frac{\partial l}{\partial y}\right)\right)_{i-1/2,j} &= \frac{(\Delta x_i + \Delta x_{i+1}) \left(\left(\frac{\partial l}{\partial y}\right)_{i,j-1/2} - \left(\frac{\partial l}{\partial y}\right)_{i-1,j-1/2}\right)}{(\Delta x_i + \Delta x_{i-1}) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \\ &+ \frac{(\Delta x_i + \Delta x_{i-1}) \left(\left(\frac{\partial l}{\partial y}\right)_{i+1,j-1/2} - \left(\frac{\partial l}{\partial y}\right)_{i,j-1/2}\right)}{(\Delta x_i + \Delta x_{i+1}) \left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \end{aligned} \tag{B.46}$$

*B.2.3. All the other variables except the horizontal velocities.* In the computations of the salinity (*S*), temperature (*T*), pollutant concentration (*C*), kinetic energy (*k*) or the rate of dissipation of kinetic energy (*ε*), the finite difference formulations of the differential terms are prepared for the staggered scheme shown in Figure B3.

The following approximations can be derived in respect to the computational molecule illustrated in Figure B3. It should be noted that, all computations are performed at point  $\otimes_{i,j}$ :

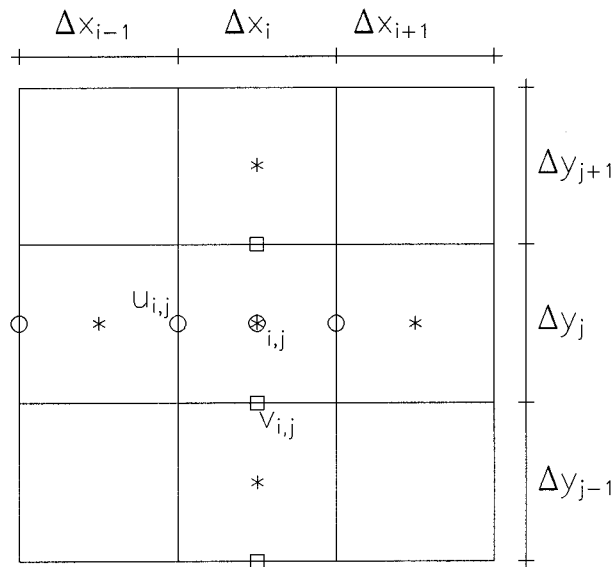


Figure B3. Finite difference molecule for computations of all other variables except the horizontal velocities. Symbols represent the variables as follows: *v* □, *u* ○, all other variables \*.

$$\left(\frac{\partial C}{\partial x}\right)_{i,j} = \frac{(\Delta x_i + \Delta x_{i+1})(C_{i,j} - C_{i-1,j})}{(\Delta x_i + \Delta x_{i-1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} + \frac{(\Delta x_i + \Delta x_{i-1})(C_{i+1,j} - C_{i,j})}{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \quad (\text{B.47})$$

$$\left(\frac{\partial C}{\partial y}\right)_{i,j} = \frac{(\Delta y_j + \Delta y_{j+1})(C_{i,j} - C_{i,j-1})}{(\Delta y_j + \Delta y_{j-1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} + \frac{(\Delta y_j + \Delta y_{j-1})(C_{i,j+1} - C_{i,j})}{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} \quad (\text{B.48})$$

$$\begin{aligned} \left(\frac{\partial^2 C}{\partial x^2}\right)_{i,j} = & 2\left(\frac{C_{i-1,j}}{\frac{(\Delta x_{i-1} + \Delta x_i)\left(\Delta x_i + \frac{\Delta x_{i-1} + \Delta x_{i+1}}{2}\right)}{2}} - \frac{C_{i,j}}{\frac{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \Delta x_{i-1}\right)}{2}}\right. \\ & \left. + \frac{C_{i+1,j}}{\frac{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)}{2}}\right) \end{aligned} \quad (\text{B.49})$$

$$\begin{aligned} \left(\frac{\partial^2 C}{\partial y^2}\right)_{i,j} = & 2\left(\frac{C_{i,j-1}}{\frac{(\Delta y_{j-1} + \Delta y_j)\left(\Delta y_j + \frac{\Delta y_{j-1} + \Delta y_{j+1}}{2}\right)}{2}} - \frac{C_{i,j}}{\frac{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \Delta y_{j-1}\right)}{2}}\right. \\ & \left. + \frac{C_{i,j+1}}{\frac{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)}{2}}\right) \end{aligned} \quad (\text{B.50})$$

$$\frac{\partial u}{\partial x} = \frac{u_{i+1,j} - u_{i,j}}{\Delta x_i}, \quad \frac{\partial v}{\partial y} = \frac{v_{i,j+1} - v_{i,j}}{\Delta y_j} \quad (\text{B.51})$$

$$\left(\frac{\partial l}{\partial x}\right)_{i,j} = \frac{(\Delta x_i + \Delta x_{i+1})(l_{i,j} - l_{i-1,j})}{(\Delta x_i + \Delta x_{i-1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} + \frac{(\Delta x_i + \Delta x_{i-1})(l_{i+1,j} - l_{i,j})}{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)} \quad (\text{B.52})$$

$$\begin{aligned} \left(\frac{\partial^2 l}{\partial x^2}\right)_{i,j} = & 2\left(\frac{l_{i-1,j}}{\frac{(\Delta x_{i-1} + \Delta x_i)\left(\Delta x_i + \frac{\Delta x_{i-1} + \Delta x_{i+1}}{2}\right)}{2}} - \frac{l_{i,j}}{\frac{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \Delta x_{i-1}\right)}{2}}\right. \\ & \left. + \frac{l_{i+1,j}}{\frac{(\Delta x_i + \Delta x_{i+1})\left(\Delta x_i + \frac{\Delta x_{i+1} + \Delta x_{i-1}}{2}\right)}{2}}\right) \end{aligned} \quad (\text{B.53})$$

$$\left(\frac{\partial l}{\partial y}\right)_{i,j} = \frac{(\Delta y_j + \Delta y_{j+1})(l_{i,j} - l_{i,j-1})}{(\Delta y_j + \Delta y_{j-1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} + \frac{(\Delta y_j + \Delta y_{j-1})(l_{i,j+1} - l_{i,j})}{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)} \quad (\text{B.54})$$

$$\begin{aligned} \left(\frac{\partial^2 l}{\partial y^2}\right)_{i,j} = & 2\left(\frac{l_{i,j-1}}{\left(\frac{\Delta y_{j-1} + \Delta y_j}{2}\right)\left(\Delta y_j + \frac{\Delta y_{j-1} + \Delta y_{j+1}}{2}\right)} - \frac{l_{i,j}}{\frac{(\Delta y_j + \Delta y_{j+1})(\Delta y_j + \Delta y_{j-1})}{2}}\right. \\ & \left. + \frac{l_{i,j+1}}{\frac{(\Delta y_j + \Delta y_{j+1})\left(\Delta y_j + \frac{\Delta y_{j+1} + \Delta y_{j-1}}{2}\right)}{2}}\right) \end{aligned} \quad (\text{B.55})$$

## REFERENCES

1. Lin B, Falconer RA. Three-dimensional layer-integrated modelling of estuarine flows with flooding and drying. *Estuarine, Coastal and Shelf Science* 1997; **44**: 737–751.
2. Kim C, Lee J. A three-dimensional PC-based hydrodynamic model using an ADI scheme. *Coastal Engineering* 1994; **23**: 271–287.
3. Casulli V, Cheng RT. Semi implicit finite difference methods for three-dimensional shallow water flow. *International Journal for Numerical Methods in Fluids* 1992; **15**: 629–648.
4. Davies AM, Stephens CV. Comparison of the finite difference and Galerkin methods as applied to the solution of the hydrodynamic equations. *Applied Mathematical Modelling* 1983; **7**: 226–240.
5. Gordon RB, Spaulding ML. Numerical simulation of the tidal and wind driven circulation in Narragansett Bay. *Estuarine, Coastal and Shelf Science* 1987; **24**: 611–636.
6. Blumberg AF, Mellor LG. A description of a three-dimensional coastal ocean circulation model. In *Three-Dimensional Coastal Ocean Models*, Heaps NS (ed.). American Geophysical Union: Washington, 1987; 1–16.
7. Stansby PK. Semi implicit finite volume shallow water flow and solute transport solver with  $k-\epsilon$  turbulence model. *International Journal for Numerical Methods in Fluids* 1997; **25**: 285–313.
8. Huang W, Spaulding M. 3D model of estuarine circulation and water quality induced by surface discharges. *Journal of Hydraulic Engineering, ASCE* 1995; **121**: 300–311.
9. Koutitas C, O'Connor B. Modelling three-dimensional wind-induced flows. *Journal of the Hydraulic Division, ASCE* 1980; **106**: 1843–1865.
10. Davies AM. On computing the three dimensional flow in a stratified sea using the Galerkin method. *Applied Mathematical Modelling* 1982; **6**: 347–351.
11. Balas L. Three-dimensional numerical modelling of transport processes in coastal water bodies. PhD Thesis, The Graduate School of Natural and Applied Sciences, Middle East Technical University, Ankara, Turkey, 1998.
12. Kowalik Z, Murty TS. Numerical modeling of ocean dynamics, advanced series on ocean engineering. *World Scientific* 1993; **5**: 12–13.
13. Rodi W. *Turbulence models and their application in hydraulics, IAHR Series* (3rd edn). IAHR: Delft, The Netherlands, 1993.
14. Mohammadi B, Pironneau O. *Analysis of the k-epsilon Turbulence Model*. John Wiley and Sons: London, UK, 1994.
15. Huq P, Stretch DD. Critical dissipation rates in density stratified turbulence. *Physics of Fluids* 1995; **7**: 1034–1039.
16. Large WG, Pond S. Open ocean momentum flux measurements in moderate to strong winds. *Journal of Physical Oceanography* 1981; **11**: 324–336.
17. Demuren AO, Rodi W. Side discharges into open channels: mathematical model. *Journal of Hydraulic Engineering, ASCE* 1983; **109**: 1707–1722.
18. Wardle AP, Hapoğlu H. On the solution of models of binary and multicomponent packed distillation columns using orthogonal collocation on finite elements. *Chemical Engineering Research and Design* 1994; **72**: 551–564.
19. Karacan S, Cabbar Y, Alpbaz M, Hapoğlu H. The steady state and dynamic analysis of packed distillation column based on partial differential approach. *Chemical Engineering and Processing* 1998; **37**: 379–388.



20. Anderson AD, Tannehill JC, Pletcher RH. *Computational Fluid Mechanics and Heat Transfer*. Hemisphere Publishing Corporation: New York, USA, 1984.
21. Tsanis KI, Leutheusser HJ. The structure of turbulent shear-induced countercurrent flow. *Journal of Fluid Mechanics* 1988; **189**: 531–552.
22. Özhan E, Balas L, Başaran SA. Forced flushing of marinas in enclosed seas. BORDOMER '97. *International Conference on Coastal Environment and Conservation* 1997; **2**: 65–70.
23. Balas L, Özhan E, Öztürk C. Three-dimensional modelling of hydrodynamic and transport processes in Ölüdeniz Lagoon. In *Proceedings of the Third International Conference on the Mediterranean Coastal Environment, MEDCOAST '97, MEDCOAST Secretariat*, vol. 2, Özhan E (ed.), 1997; 1097–1109.
24. Balas L, Özhan E. Three-dimensional modelling of transport processes in stratified coastal waters. In *Proceedings of International Conference on Hydroinformatics*, vol. 1, Babovic V, Larsen LC (eds). A.A. Balkema: Rotterdam, 1998; 97–104.