INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2006; **52**:1315–1337
Published online 18 April 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1225

High-order compact numerical schemes for non-hydrostatic free surface flows

Stephen L. Anthonio*,† and Kevin R. Hall

Department of Civil Engineering, Queen's University, Kingston, Ont., Canada K7L 3N6

SUMMARY

The development of a numerical scheme for non-hydrostatic free surface flows is described with the objective of improving the resolution characteristics of existing solution methods. The model uses a high-order compact finite difference method for spatial discretization on a collocated grid and the standard, explicit, single step, four-stage, fourth-order Runge–Kutta method for temporal discretization. The Cartesian coordinate system was used. The model requires the solution of two Poisson equations at each time-step and tridiagonal matrices for each derivative at each of the four stages in a time-step. Third- and fourth-order accurate boundaries for the flow variables have been developed including the top non-hydrostatic pressure boundary. The results demonstrate that numerical dissipation which has been a problem with many similar models that are second-order accurate is practically eliminated. A high accuracy is obtained for the flow variables including the non-hydrostatic pressure. The accuracy of the model has been tested in numerical experiments. In all cases where analytical solutions are available, both phase errors and amplitude errors are very small. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: non-hydrostatic; high-order compact; numerical dissipation; collocated grid; explicit Runge–Kutta; free surface flow

INTRODUCTION

Flow simulation in Civil Engineering has typically been carried out using the hydrostatic approximation of the Navier–Stokes equations. This has been useful for cases where the vertical scales of motion are small compared with the horizontal scales of motion. For many other flows, the hydrostatic approximation becomes inappropriate. Examples of these include, propagation of short gravity waves, buoyancy-driven flows and flows occurring in rapidly changing bottom topography. The full solution of the momentum equations of fluid flow becomes

*Correspondence to: Stephen L. Anthonio, Department of Civil Engineering, Queen's University, Kingston, Ont., Canada K7L 3N6.
†E-mail: stephen@civil.queensu.ca

Contract/grant sponsor: Canadian Commonwealth Scholarship and Fellowship program
Contract/grant sponsor: Water Research Institute of the Council for Scientific and Industrial Research

Received 14 September 2005
Revised 14 February 2006
Accepted 15 February 2006

necessary. The continued advancement of computer technology has spurred the realization of numerical models using the complete Reynolds-averaged Navier–Stokes equations (see for instance References [1–6]). These models have been formulated on a staggered grid. Time integration is carried out either explicitly, semi-implicitly or implicitly. They are, at best, second-order accurate but may sometimes be first-order accurate in time when the popular projection method is used without any special treatment of the diffusive terms. In such cases the projection method introduces a splitting error that is proportional to the time-step and the eddy viscosity coefficient [7].

The implications of the second-order accuracy are that the grid spacings have to be small enough to achieve an acceptable level of accuracy. Larger grid spacings add to numerical dissipation. This statement is applicable to explicit and semi-implicit as well as implicit discretization methods. The small grid spacings required adds to the computational demands of the numerical scheme in terms of data storage and the total time taken for the computations. In other cases, small grid spacings may violate the cell Reynolds number constraint (i.e. the cell Reynolds number $R_c \leqslant 2$) leading to oscillations in the solution. The advantages of an implicit scheme are primarily that a larger time-step could be possible and still obtain numerically convergent schemes. For accuracy purposes, however, the chosen time-step requires some control. With discretizations of the Navier–Stokes equations applied to convective dominated flows as often occurs in hydraulic engineering, considerations of accuracy eliminate the advantages of the implicit scheme.

The usual practice with spatial discretization in the existing numerical solutions have been to use the staggered grid introduced by Harlow and Welch [8]. Without the staggered grid, the solution exhibit the so-called checker-board instability. A disadvantage of the staggered grid is that it introduces additional levels of complexity in programming. As an example, using the cell-centred discretization, the horizontal components of velocity are defined on different cell faces and the pressure at the centre of the cell. If multigrid methods have to be used to accelerate the convergence of the solution, the grid transfer operators are often difficult to define. In complex terrains where general coordinates have to be used, the programming becomes inefficient as the metrics of the transformation are required for all the different positions on the cell that the variables are defined. The use of collocated grid requires only one metric of the transformation to be defined for each computational cell.

Other challenges faced with the numerical modelling using second-order methods include the treatment of the top pressure boundary. It is often assumed to be atmospheric and no boundary condition is provided for the velocity components. With the use of Cartesian coordinates, there is the problem of the treatment of the velocity flux across the surface. It is sometimes assumed that the gradient of the horizontal velocity components across the top boundary is zero [1]. This leads to a first-order accurate solution that yields numerically dissipative results. The alternative high-order top boundary yields unstable results. Another alternative that has proven useful has been the transformation of the vertical axis to σ coordinates. Solution of the transformed equations may, given certain circumstances, yield spurious results referred to as *hydrostatic inconsistency*. The anomaly is attributed to the truncation errors of the discretization and the pressure gradient term.

Recently, there have been progress in the treatment of the free surface pressure. An example is provided by Stelling and Zijlema [9] who used an edge-based instead of a cell-centred scheme to approximate the pressure more accurately. Another example is presented by Yuan and Wu [6] who used the cell-centred scheme by integrating the pressure from the free surface

to the centre of the cell. Both examples by treating the pressure for the surface cell more accurately make possible the use of fewer vertical layers for simulating wave transformation. The accuracy of the method as pointed out by Stelling and Zijlema [9], however, is expected to approach that of the Boussinesq-type wave model of Peregrine [10] as the number of vertical layers approaches one.

The Boussinesq-type wave models essentially transforms what is a three-dimensional phenomenon to a two-dimensional one by depth-averaging. By doing so, fluid properties that vary significantly in the vertical are not well represented. The advantage of the non-hydrostatic wave model over the Boussinesq-type wave model is its ability to resolve the vertical variation of fluid properties more accurately. As an example, the simulation of the effects of viscosity or turbulence on the flow is better carried out with the non-hydrostatic wave model. The fewer the number of vertical layers, the lesser the advantages of the non-hydrostatic free surface model over the Boussinesq-type wave models.

The formulation of the high-order (defined as an order of accuracy greater than or equal to three) model has become necessary to improve the resolution characteristics, increase accuracy relating to numerical dissipation and make possible the use of a larger time-step to obtain numerically stable schemes. With the use of the current model, a smaller number of grid spacings are required to obtain the same level of accuracy as for second-order methods.

High-order boundaries have been formulated to take account of the required high-order internal solution. It should be noted that the order of accuracy of the boundary can be one less than that of the accuracy of the internal solution to achieve an overall accuracy determined by the internal solution [11]. For the fourth-order solution, third- and fourth-order accurate boundaries have been developed. The compact high-order scheme adopted has a smaller truncation error compared with the standard fourth-order scheme. Besides, the compact scheme requires only three grid points to define its numerical stencil just as the second-order schemes. Comparatively, the standard fourth order requires five grid points. The computational demands for the high-order compact schemes are the same for both cases. Unlike the second-order schemes, however, the truncation errors of the compact scheme are smaller.

Model development

The spatial discretization has been carried out on a collocated grid to reduce the level of complexity associated with the staggered grid and facilitate the efficient programming for its use with general coordinates. The first-order time accurate Euler method is presented followed by the single-point, four-stage fourth-order Runge–Kutta method. The high-order time integration became necessary to avoid some of the constraints imposed by the Courant–Freiedrichs–Levy (CFL) condition and cell Reynolds number. As a result, convectively stable solutions are possible even for moderate-to-high Reynolds numbers. In the proposed algorithm, all the required velocity components are computed explicitly after which the pressure is determined from a Poisson equation from the known velocities. This is different from the projection method where an intermediate velocity is computed first, which does not satisfy continuity with the assumption of hydrostatic conditions. In the next step, the non-hydrostatic pressure is determined by ensuring that the continuity equation is satisfied after which the velocities are corrected. The projection method yields a first-order solution in time.

The model has been tested with the standing wave in a basin simulations to demonstrate the elimination of the problem of *checker-board* instability. With the proposed method, the results

show a significant reduction of numerical dissipation as compared with existing models. Further properties of the model are investigated including the number of Poisson solves required for each time-step to achieve the high-order accuracy. More importantly, the time taken for the total scheme is assessed for the various possible number of Poisson solves suggested in the literature. Other tests used in the model verification include propagation of solitary wave in a flat channel and wind-driven circulation.

While the model can be formulated in the implicit form, the convective-dominated flows used in the simulations make the implicit form unnecessary. The implicit form is suitable for diffusive-dominated flows. In such situations, a semi-implicit method is suggested. This will be the subject of a future paper.

In the following section, the governing equations of fluid flow are presented. The total pressure is decomposed into hydrostatic and non-hydrostatic components and new equations are derived based on the non-hydrostatic pressure and the gradient of the free surface. The numerical solution method for the governing equations is then presented. This is based on a fourth order in space method. Two time integration methods, the explicit first-order Euler method and explicit fourth-order Runge–Kutta method are described. Results for the verification of the algorithm are presented to demonstrate some of the advantages of the high-order solution method. These are discussed in the next section and the main conclusions from this work are presented.

GOVERNING EQUATIONS

The mathematical equations governing incompressible free surface flows are obtained from the conservation of mass and momentum.

$$\nabla \cdot \mathbf{u} = 0 \tag{1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \frac{1}{\rho}\nabla P = \nabla \cdot (\nu \nabla \mathbf{u}) + \mathbf{g} - 2\zeta \times \mathbf{u} + \mathbf{f} \tag{2}$$

where $\mathbf{u}$ represents the instantaneous velocity at time t with corresponding pressure P and density ρ. The kinematic viscosity is ν. The acceleration due to gravity and the coriolis force are represented by g and ζ, respectively. All other body forces are represented by $\mathbf{f}$. Equation (1) is the continuity equation and Equation (2) the well-known Navier–Stokes equations of fluid flow.

The continuity equation (1) and the momentum equations, Equation (2), ignoring the coriolis and other external forces can be written as

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{3}$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} + \frac{\partial p}{\partial x} = \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) \tag{4}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} + \frac{\partial p}{\partial y} = \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) \tag{5}$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} + \frac{\partial p}{\partial z} = \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) - g \tag{6}$$

where p is the kinematic pressure, P/ρ, which comprises the non-hydrostatic (hydrodynamic), q, and hydrostatic, p_h, components, i.e.

$$p = p_h + q = g(\eta - z) + q \tag{7}$$

Substituting Equation (7) into the momentum equations, yield

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} + g\frac{\partial \eta}{\partial x} + \frac{\partial q}{\partial x} = \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) \tag{8}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} + g\frac{\partial \eta}{\partial y} + \frac{\partial q}{\partial y} = \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) \tag{9}$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} + \frac{\partial q}{\partial z} = \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) \tag{10}$$

where x, y and z, are the two horizontal and one vertical Cartesian coordinates, respectively. The corresponding velocity components along the horizontal and vertical axes are u, v and w, respectively.

Free surface boundary condition

At the free surface, two boundary conditions are applied. The first is the Kinematic free surface boundary condition

$$\frac{\partial \eta}{\partial t} + u\frac{\partial \eta}{\partial x} + v\frac{\partial \eta}{\partial y} = w \tag{11}$$

and the second is the dynamic free surface boundary condition which is given by the prescription of the pressure at the free surface usually assumed to be atmospheric.

A convenient form of the kinematic free surface boundary condition is obtained by integrating the continuity equation from the bottom to the free surface. Applying the Leibniz rule and the kinematic condition yields

$$\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x}\left(\int_{-d}^{\eta} u\,\mathrm{d}z\right) + \frac{\partial}{\partial y}\left(\int_{-d}^{\eta} v\,\mathrm{d}z\right) = 0 \tag{12}$$

Equation (12) represents the conservative form of the free surface boundary condition while at the same time it accounts for an impermeable bottom. It is often useful when the bottom is fixed. Its main constraint is that the gradient of the surface profile $\partial\eta/\partial x$ must be less than the grid aspect ratio, $\gamma = \Delta x/\Delta z$ i.e.

$$\frac{\partial \eta}{\partial x} \leqslant \frac{\Delta z}{\Delta x} \tag{13}$$

and

$$\frac{\partial \eta}{\partial y} \leqslant \frac{\Delta z}{\Delta y} \tag{14}$$

Assuming the surface tension is negligible and in the absence of tangential stresses at the surface (i.e. wind) the dynamic free surface boundary condition takes the form

$$q = 0 \quad \text{at the free surface} \tag{15}$$

Impermeable boundary condition

At impermeable boundaries, the normal velocity and the gradient of tangential velocities are zero. For example, at the vertical wall parallel to the vertical axis the velocity normal to the wall, u, and the tangential velocity, w, are given by

$$u = 0 \tag{16}$$

$$\frac{\partial w}{\partial x} = 0 \tag{17}$$

Outflow boundary

The outflow boundary for free surface flows are typically obtained from the Sommerfeld's radiation condition given by

$$\frac{\partial \phi}{\partial t} + c\,\cos(\alpha)\frac{\partial \phi}{\partial x} + c\,\sin(\alpha)\frac{\partial \phi}{\partial y} = 0 \tag{18}$$

where α is the angle that the normal to the boundary makes with the outgoing waves, ϕ represents anyone of the variables u, v, w and η. The wave phase velocity is represented by c. Equation (18) performs reasonably well for monochromatic waves but far from perfect for irregular waves.

NUMERICAL FORMULATION

The discretization of the governing equations is carried out with the finite difference method on a Cartesian coordinate system. The computational domain is discretized into $N_x \times N_y \times N_z$ uniform rectangular cells using a collocated grid (see Figure 1) on a cell-centred mesh where all the variables are defined at the centre of the cell. The fourth-order accurate in space,

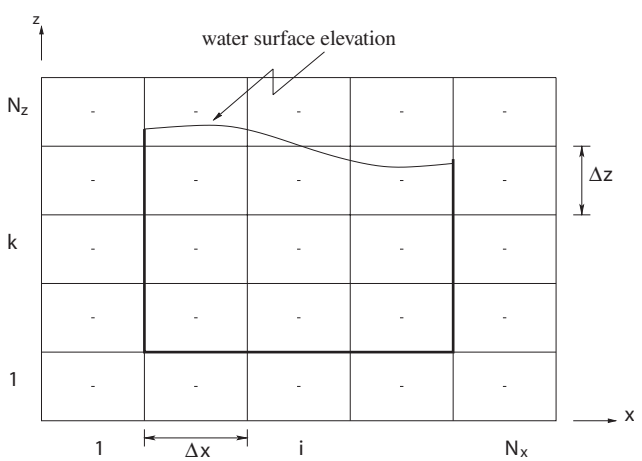

Figure 1. Layout of the cell-centred collocated grid used in the numerical scheme.

compact discretization method has been presented by several authors (see for instance References [12, 13]). The formula for approximating the first and second derivatives are given by

$$\frac{\partial}{\partial x} = \left(1 + \frac{\Delta x^2}{6}\delta_x^2\right)^{-1} \delta_{x0} + O(\Delta x)^4 \tag{19}$$

$$\frac{\partial^2}{\partial x^2} = \left(1 + \frac{\Delta x^2}{12}\delta_x^2\right)^{-1} \delta_x^2 + O(\Delta x)^4 \tag{20}$$

where the finite difference operators δ_{x0} and δ_x^2 are the standard second-order central difference formula for the first and second derivatives, respectively, i.e.

$$\delta_{x0} f_i = \frac{f_{i+1} - f_{i-1}}{2\Delta x} \tag{21}$$

$$\delta_x^2 f_i = \frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2} \tag{22}$$

As first proposed by Hirsh [14], the partial differential equations are solved by treating the derivatives as unknowns. Let F and S represent the first and second derivatives, respectively, i.e.

$$\frac{\partial f_i}{\partial x} = F_i \tag{23}$$

$$\frac{\partial^2 f_i}{\partial x^2} = S_i \tag{24}$$

By eliminating the differentials from Equations (19) and (20), the following equations are obtained:

$$F_{i+1} + 4F_i + F_{i-1} = \frac{3}{\Delta x}(f_{i+1} - f_{i-1}) \tag{25}$$

$$S_{i+1} + 10S_i + S_{i-1} = \frac{12}{\Delta x^2}(f_{i+1} - 2f_i + f_{i-1}) \tag{26}$$

The two tridiagonal equations (25) and (26) are solved directly for the derivatives, F and S. The tridiagonal equations can also be represented in matrix form as

$$\mathbf{A}u = \mathbf{b} \tag{27}$$

where $\mathbf{A}$ is the tridiagonal matrix. This matrix is constant and typically solved through an LU decomposition that is carried out once at the start of the simulations. The resulting L and U matrices are stored and the solution of the matrix is obtained by the computation of the right-hand side, $\mathbf{b}$, followed by a forward and backward substitution. Note that Equation (25) can also be used to integrate, in which case it becomes *Simpson's one-third rule*.

At the boundaries, the derivatives are determined by a backward difference formula that must be at least one order less than the interior accuracy to maintain a fourth-order accurate

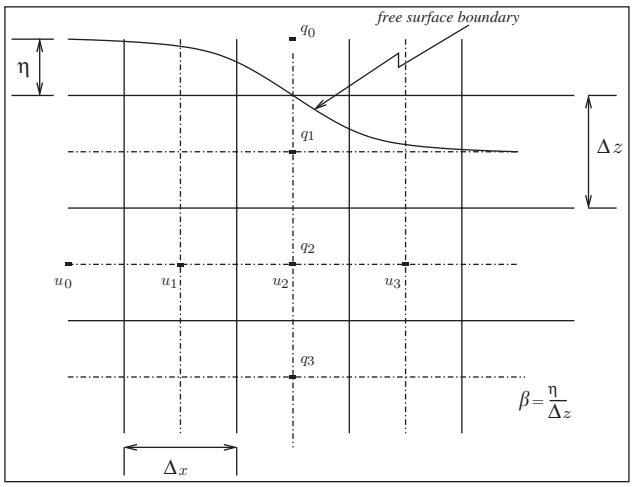

Figure 2. Diagram of top pressure boundary.

overall solution. The third- and fourth-order solution have been derived by Lele [13], i.e.

$$F_0 + 2F_1 = \frac{1}{\Delta x}\left(-\frac{5}{2} f_0 + 2f_1 + \frac{1}{2} f_2\right), \quad O(\Delta x)^3 \tag{28}$$

$$F_0 + 3F_1 = \frac{1}{\Delta x}\left(-\frac{17}{6} f_0 + \frac{3}{2} f_1 + \frac{3}{2} f_2 + -\frac{1}{6} f_3\right), \quad O(\Delta x)^4 \tag{29}$$

For the second derivative boundary, the third-order formula is given by

$$S_0 + 11S_1 = \frac{1}{\Delta x^2}(13 f_0 + -27 f_1 + 15 f_2 - f_3), \quad O(\Delta x)^3 \tag{30}$$

The boundary conditions provided above do not account for the systems boundary whether Dirichlet or Neumann. The formulation above is only used to compute the derivatives. The Dirichlet or Neumann boundaries are applied for the cell-centred discretization (see Figure 2) using the following formulae:

$$u_0 = \tfrac{21}{23} u_1 + \tfrac{3}{23} u_2 - \tfrac{1}{23} u_3, \quad O(\Delta x)^4 \quad \text{Neumann boundary} \tag{31}$$

$$u_0 = -3u_1 + u_2 - \tfrac{1}{5} u_3, \quad O(\Delta x)^4 \quad \text{Dirichlet boundary} \tag{32}$$

The third-order boundaries are given by

$$u_0 = u_1, \quad O(\Delta x)^3 \quad \text{Neumann boundary} \tag{33}$$

$$u_0 = -2u_1 + \tfrac{1}{3} u_2, \quad O(\Delta x)^3 \quad \text{Dirichlet boundary} \tag{34}$$

The solution method used for the momentum equations follow. Define u^*, v^* and w^* by excluding the terms with the dynamic pressure, q.

$$\frac{u^* - u^n}{\Delta t} = -\left[u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} + g\frac{\partial \eta}{\partial x} - \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)\right]^n \tag{35}$$

$$\frac{v^* - v^n}{\Delta t} = -\left[u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} + g\frac{\partial \eta}{\partial x} - \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right)\right]^n \tag{36}$$

$$\frac{w^* - w^n}{\Delta t} = -\left[u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} - \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right)\right]^n \tag{37}$$

so that the discrete form of Equations (8) and (9) can be written as

$$\frac{u^{n+1} - u^*}{\Delta t} = -\left[\frac{\partial q}{\partial x}\right]^n \tag{38}$$

$$\frac{v^{n+1} - v^*}{\Delta t} = -\left[\frac{\partial q}{\partial y}\right]^n \tag{39}$$

$$\frac{w^{n+1} - w^*}{\Delta t} = -\left[\frac{\partial q}{\partial z}\right]^n \tag{40}$$

The superscript refers to the discrete time-level (e.g. n refers to time, $t = n\Delta t$). The first order in time explicit method is given by

$$\begin{aligned}\frac{u^{n+1} - u^n}{\Delta t} &= -\left[u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} + g\frac{\partial \eta}{\partial x} + \frac{\partial q}{\partial x} - \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)\right]^n \\ &= \mathscr{F}_u\end{aligned} \tag{41}$$

$$\begin{aligned}\frac{v^{n+1} - v^n}{\Delta t} &= -\left[u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} + g\frac{\partial \eta}{\partial y} + \frac{\partial q}{\partial y} - \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right)\right]^n \\ &= \mathscr{F}_v\end{aligned} \tag{42}$$

$$\begin{aligned}\frac{w^{n+1} - w^n}{\Delta t} &= -\left[u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} + \frac{\partial q}{\partial z} - \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right)\right]^n \\ &= \mathscr{F}_w\end{aligned} \tag{43}$$

$$\begin{aligned}\frac{\eta^{n+1} - \eta^n}{\Delta t} &= -\left[\frac{\partial}{\partial x}\left(\int_{-d}^{\eta} u\,\mathrm{d}z\right) + \frac{\partial}{\partial y}\left(\int_{-d}^{\eta} v\,\mathrm{d}z\right)\right]^n \\ &= \mathscr{F}_\eta\end{aligned} \tag{44}$$

where $\mathscr{F}_u$, $\mathscr{F}_v$, $\mathscr{F}_w$ and $\mathscr{F}_\eta$ are the approximations of the explicit terms and d, is the depth measured from the still water level. The integration required in Equation (12) from the bottom to the free surface is evaluated using Equation (25). Since the top cells vary in size with

time and space, the integration of the top cells require special attention. First the horizontal velocity is interpolated vertically using a fourth-order mid-point interpolation method [13], which gives the values at the cell faces. The integration from the free surface to the still water level is carried out with a direct fit polynomial passing through the five nodes comprising the interpolated (cell faces) and velocity nodes (cell centres). The details of the method are given in Reference [15].

$$I_f = \left[az + b\frac{z^2}{2} + c\frac{z^3}{3} + d\frac{z^4}{4} + e\frac{z^5}{5} \right]_0^{\eta} \tag{45}$$

where the coefficients a, b, c, d and e are determined from the direct fit polynomial. The rest of the vertical integration beneath the still water level to the bottom of the channel is obtained using Simpson's one-third rule, which is also fourth-order accurate.

The Poisson equation for the dynamic pressure distribution can be obtained by substituting for u^{n+1}, v^{n+1} and w^{n+1} from Equations (38)–(40) into the continuity equations giving

$$\nabla^2 q^n = \frac{1}{\Delta t}\left(\frac{\partial u^*}{\partial x} + \frac{\partial v^*}{\partial y} + \frac{\partial w^*}{\partial z} \right) = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^* \tag{46}$$

The variables u^*, v^* and w^* are functions of u^n, v^n and w^n. The kinematic dynamic pressure, q, is computed at the time level n, and not $n+1$ as in the pressure projection method. The solution of the Poisson equation also uses the fourth-order compact finite difference method.

As explained by Roache [16], by applying the divergence operator to the momentum equations to obtain the Poisson equation for the pressure, the order of the system under solution has been raised. Boundary conditions for the pressure are therefore required. It is important to note that no new arbitrary boundary conditions have been used. The boundary conditions for the pressure are consistent with the original equations being solved. These boundary conditions became necessary only because the derived pressure Poisson equation is of higher order. The required boundary condition for the pressure is determined by projecting the momentum equation onto the unit normal, $\mathbf{N}$, to the boundary giving the Neumann boundary.

$$\left[\frac{\partial q}{\partial N} \right]_{\Gamma}^{n} = -\frac{1}{\Delta t}(\mathbf{u}_{\Gamma}^{n+1} - \mathbf{u}_{\Gamma}^{*}) \cdot \mathbf{N} \tag{47}$$

The dynamic free surface boundary condition is applied using the following formula:

$$q_0 = aq_1 + bq_2 + cq_3 \tag{48}$$

where q_0 is the extrapolated dynamic pressure outside the domain, q_1, q_2 and q_3 are the interior dynamic pressures (see Figure 2). The coefficients, a, b and c are to be determined. A Taylor series expansion about these points with base on the free surface gives the equation (see Appendix A for the derivation)

$$\begin{pmatrix} r_1 & r_2 & r_3 \\ r_1^2 & r_2^2 & r_3^2 \\ r_1^3 & r_2^3 & r_3^3 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} s \\ s^2 \\ s^3 \end{pmatrix} \tag{49}$$

where

$$r_1 = \tfrac{1}{2} + \beta, \quad r_2 = \tfrac{3}{2} + \beta, \quad r_3 = \tfrac{5}{2} + \beta \tag{50}$$

$$s = -\left(\frac{1}{2} - \beta\right), \quad \beta = \frac{\eta}{\Delta z} \tag{51}$$

Equation (49) is fourth-order accurate. Third-order solutions can be obtained in a similar manner. Since, β, is a function of η, which varies with time and space, the coefficients also vary with time and space. They are determined by solving Equation (49) at every step during the iteration for the dynamic pressure. The Dirichlet boundary conditions for the velocities, Equations (32) and (34), can be obtained from Equation (49) when $\eta = 0$.

The corresponding high-order temporal discretization is obtained with the standard, explicit, single-point, four-stage, Runge–Kutta method.

$$k_1 = \Delta t \mathscr{F}(\mathbf{u}^n, q^n), \quad \mathbf{u}_1 = \mathbf{u}^n + \tfrac{1}{2}k_1 \tag{52}$$

$$q_1 = L^{-1}(\mathscr{G}(\mathbf{u}_1)) \tag{53}$$

$$k_2 = \Delta t \mathscr{F}(\mathbf{u}_1, q_1), \quad \mathbf{u}_2 = \mathbf{u}^n + \tfrac{1}{2}k_2 \tag{54}$$

$$q_2 = L^{-1}(\mathscr{G}(\mathbf{u}_2)) \tag{55}$$

$$k_3 = \Delta t \mathscr{F}(\mathbf{u}_2, q_2), \quad \mathbf{u}_3 = \mathbf{u}^n + k_3 \tag{56}$$

$$q_3 = L^{-1}(\mathscr{G}(\mathbf{u}_3)) \tag{57}$$

$$k_4 = \Delta t \mathscr{F}(\mathbf{u}_3, q_3) \tag{58}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{59}$$

$$q^{n+1} = L^{-1}(\mathscr{G}(\mathbf{u}^{n+1})) \tag{60}$$

where $\mathscr{F}$ is the approximation of the explicit terms (i.e. $\mathscr{F}_u$, $\mathscr{F}_v$, $\mathscr{F}_w$ and $\mathscr{F}_\eta$) in Equations (41), (42), (43) and (44), respectively. L is the approximation of the Laplace operator in the Poisson equation and $\mathscr{G}$ is the approximation of the right-hand side of the Poisson equation. The advantage of the high-order time discretization is that there is no constraint on the cell Reynolds number as explained by Weinan and Liu [17]. Larger time-steps can also be used to obtain stable solutions for convective-dominated flows compared with the explicit first-order Euler method.

MODEL VERIFICATION

The performance of the numerical model is assessed by comparing the model prediction with the analytical solution obtained for a standing wave in a basin and a solitary wave propagating within a horizontal channel. The standing wave in a basin simulations provide a good test case to examine any strengths and weaknesses within the numerical scheme. The extent of numerical dissipation can be determined as well as the accuracy and stability of the scheme under various conditions. The influence of the grid spacing and time-step on the stability of

the numerical scheme can be evaluated. It provides some grounds for the inter-comparison of numerical schemes for free surface flows. The solitary wave test case enables assessment of the non-linear effects on free surface flow.

Further numerical experiments are carried out to explore the performance of the model for wind-driven flow where the driving force for the flow is a constant wind stress imposed at the surface. Both convective and diffusive forces are important in this simulation. The frictional forces ensure that the fluid particles beneath the surface are set in motion in the course of time. After a certain time period, steady-state conditions are attained and there is no significant change in the velocity distribution within the flow.

Standing wave in a basin

The standing wave in a basin is probably the best test case for non-hydrostatic free surface flow numerical schemes. Water contained in a rectangular basin oscillates with time. In the absence of viscous forces, the oscillations should continue without loss of energy. A plot of the time series of any point on the free surface should give a sinusoidal wave. The initial conditions comprise a sinusoidal water surface instantaneously at rest. For the simulations discussed in this paper, the width of the channel is 10.0 m and the depth from the still water level is also 10.0 m. The initial surface profile is given by

$$\eta = a \cos kx, \quad 0 \leqslant x \leqslant 10 \tag{61}$$

where the wavenumber, $k = 2\pi/L$, the amplitude of the oscillations is $a = 0.1$ m and $L = 20$ m is the wavelength. Since the wave steepness $ak = \pi/10$ is small, the dispersion equation from linear wave theory [18] can be used to determine the period, T, of the oscillations i.e.

$$T = \sqrt{\frac{2\pi L}{g \tanh(kd)}} \tag{62}$$

For the particular case where $L = 20$ m, $d = 10.0$ m, and $g = 9.81$ m/s^2 the period of oscillations $T = 3.586$ s.

The plot of the time series at any point on the water surface distance, x, from the left end of the basin should yield a sinusoidal wave given by Equation (63). The horizontal and vertical velocities at any point, distance, x, within the flow at a depth, z, measured from the still water level are given by Equations (64) and (65). The corresponding kinematic dynamic pressure can be obtained from Equation (66).

$$\eta = a \cos kx \cos \omega t \tag{63}$$

$$u = a\frac{gT}{L} \frac{\cosh k(d+z)}{\cosh kd} \sin kx \sin \omega t \tag{64}$$

$$w = a\frac{gT}{L} \frac{\sinh k(d+z)}{\cosh kd} \cos kx \sin \omega t \tag{65}$$

$$q = g\eta \left(\frac{\cosh k(d+z)}{\cosh kd} - 1 \right) \tag{66}$$

where $\omega = 2\pi/T$.

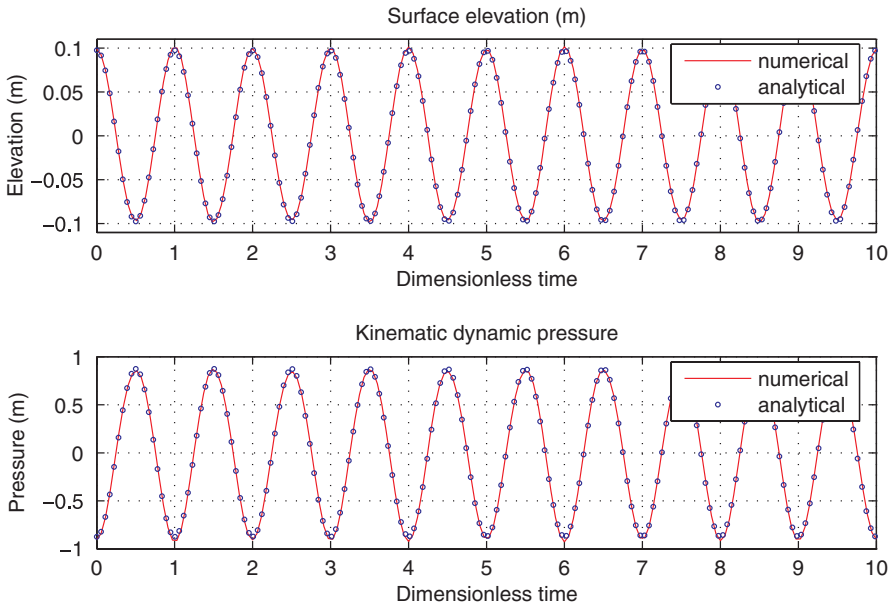

Figure 3. Surface elevation and kinematic dynamic pressure time series.

A comparison of the numerical and analytical results for the surface elevation time series at a point on the surface located 0.5 m from the left end of the basin and the kinematic dynamic pressure time series for a point located 9.5 m from the still water level and 0.5 m from the left end of the basin are presented in Figure 3. Horizontal and vertical velocity time series for a point located 0.5 m below the still water level and 4.5 and 0.5 m from the left end of the basin, respectively, are presented in Figure 4.

The grid spacing used in the simulations are 1.0 m for both the horizontal and vertical axes, which is $\Delta x = \Delta z = L/20$, and the time-step is $\Delta t = T/35$. Typical grid spacing used for similar simulations have been 0.5 m. The results show that numerical dissipation is practically eliminated and the phase error after 100 oscillations is estimated at 0.027. There is an insignificant change of the phase error with smaller wave steepness.

The four-stage Runge–Kutta method of time integration requires the solution of the Poisson equation at each stage. This would appear to impact adversely on the efficiency of the method. Numerical experiments have, however, shown that the second- and fourth-stage solutions converge much more quickly if the pressures computed prior to the iterations are used at the starting point for the new iterations. The same applies to the first and third stages although they take a relatively longer time.

It has been suggested that a single Poisson solve could be used for each time-step for the computation of pressure with the fourth-order Runge–Kutta method [19]. Since about 90% of the simulation time is taken in the solution of the Poisson equation, this would offer a more efficient solution method. The numerical experiments for the case of free surface flow does not appear to support this position. It was determined that a much smaller time-step is required for the single Poisson solve compared with four Poisson solves.

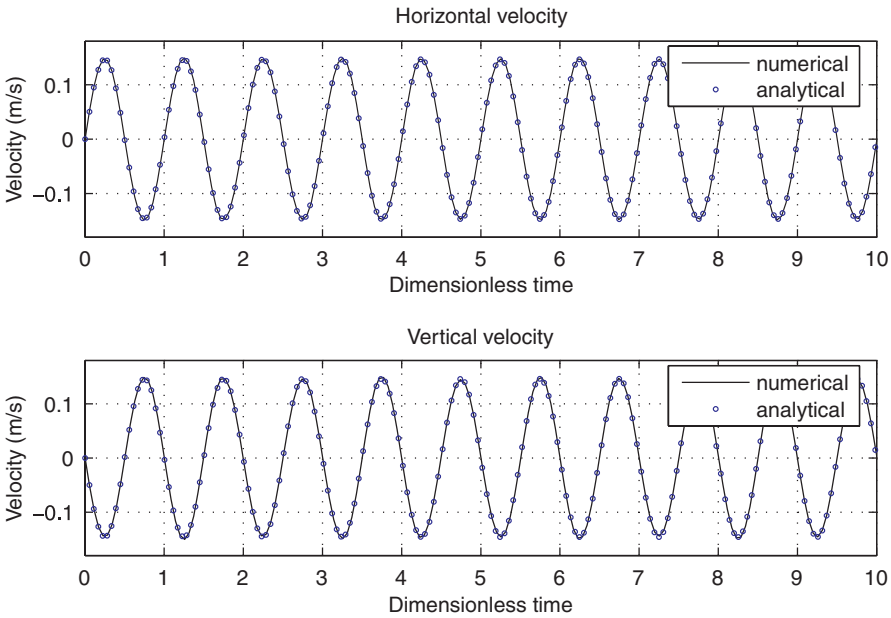

Figure 4. Horizontal and vertical velocity time series.

For long time simulations (i.e. covering 100 waves), about one-tenth the time-step used for the four Poisson solves is required to carry the simulations to the one-hundredth wave. Larger time-steps result in unstable solutions. Figures 5 and 6 show the numerical results for the first 10 waves and the last 10 waves in the 100 waves that was simulated. The solution involving the single Poisson becomes unstable between the 70th and 80th wave (not shown in plot). The solution using two and four Poisson solves show identical results for the same time-step. However, the four Poisson solves yields stable solutions with a time-step that is twice that used for the two Poisson solves.

These experiments suggest that the four Poisson solves are probably all right for simulations and this makes possible the use of the largest time-step. Two Poisson solves are probably the optimum number. It should be noted that numerical experiments with the standing wave in a basin suggest that the two Poisson solves require half the time-step that is possible with the four Poisson solves. The two Poisson solves are applied at the first and third stages of the standard Runge–Kutta method. With the use of dynamic pressures computed at the previous stage in the solution as the starting point for the iterations, a faster rate of convergence can be obtained.

Solitary wave in a long channel

The simulation of a solitary wave travelling in a channel with constant depth provides another means of verifying the performance of the numerical schemes. The solitary wave is a local elevation of the water surface above the mean water level without any other local elevation or depression of the surrounding surface. The wave produces transport of mass in the direction

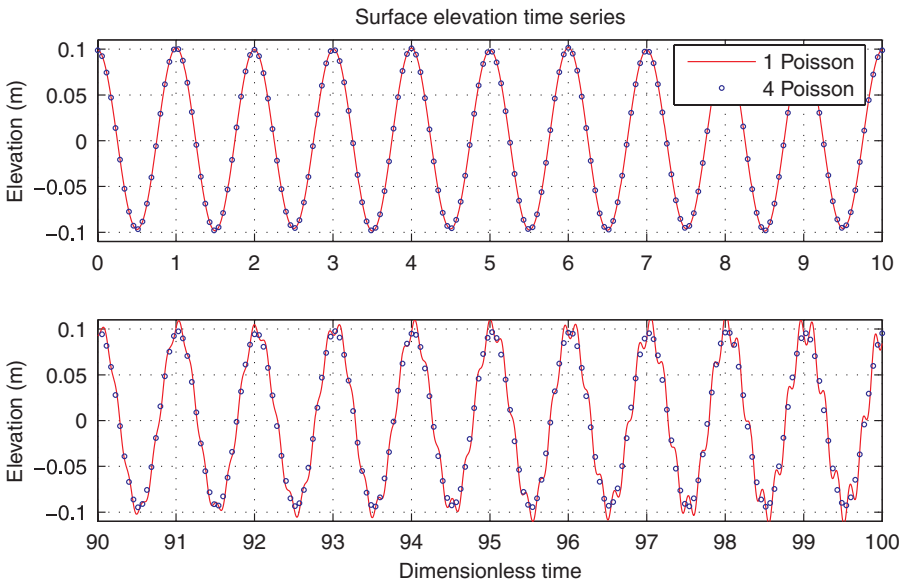

Figure 5. Effect of one Poisson solve per time-step on solution.

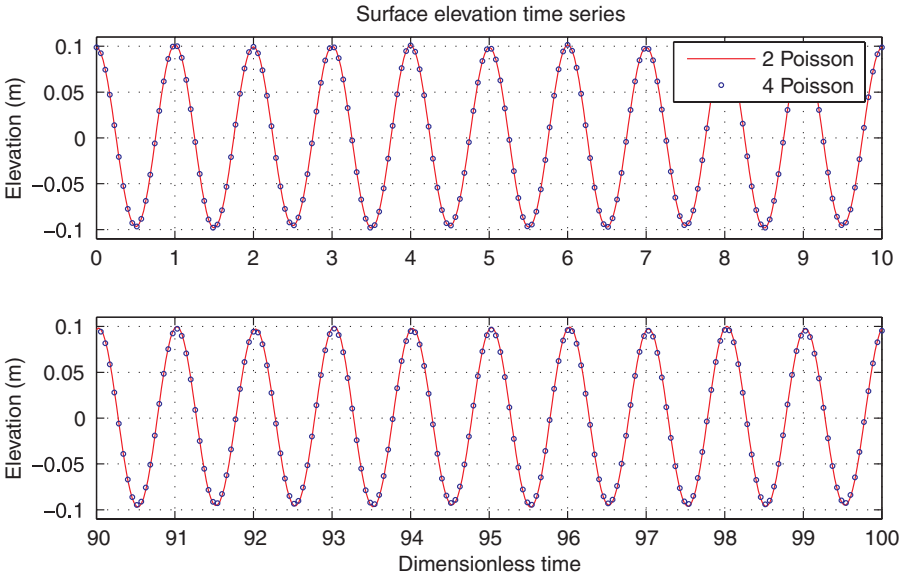

Figure 6. Effect of two and four Poisson solves per time-step on solution.

of propagation only. The velocity of the wave is constant for a constant water depth and the distribution of pressure and velocities under the wave remains constant as the wave travels. The analytical solution of the solitary wave is provided by Laitone [20]. The profile of the surface elevation, η, and velocities, u and w, at a depth, z, and time, t, for a solitary wave travelling with a celerity, c, are given by Laitone [20]

$$c = \sqrt{g(H+d)} \tag{67}$$

$$\eta = H \operatorname{sech}^2\left(\sqrt{\frac{3H}{4d^3}}(x-ct)\right) \tag{68}$$

$$u = \sqrt{gd}\frac{H}{d}\operatorname{sech}^2\left(\sqrt{\frac{3H}{4d^3}}(x-ct)\right) \tag{69}$$

$$w = \sqrt{3gd}\left(\frac{H}{d}\right)^{3/2}\frac{z}{d}\operatorname{sech}^2\left(\sqrt{\frac{3H}{4d^3}}(x-ct)\right)\tanh\left(\sqrt{\frac{3H}{4d^3}}(x-ct)\right) \tag{70}$$

where H is the height of the wave and d the depth of water.

Numerical experiments were carried out for a water depth of 10.0 m in a channel of length 1500 m. The grid spacing used was $\Delta x = \Delta z = 2.0$ m and the time-step was 0.2 s. Profiles of the surface elevation, dynamic pressure and velocities were determined from the numerical model at times 60 and 100 s and presented in Figure 7. The velocities were obtained at a depth of 5.0 m from the mean water level. For the purposes of comparison, the corresponding analytical solutions are also presented.

Contours of the distribution of dynamic pressure, horizontal and vertical velocities are presented in Figures 8 and 9 at times 60.0 and 100.0 s. The initial conditions used for the simulations correspond to time 20.0 s. As can be observed from the plots, the dynamic pressures are positive under the wave with the highest occurring beneath the wave crest. It is symmetrical about the axis passing through the crest. Further away from the crest, it continues to decrease in both directions attaining a minimum before rising again and approaching hydrostatic conditions asymptotically further away from the wave crest.

The horizontal velocity distribution remains almost constant with depth as would be expected. Along the channel, it rises beneath the wave attaining a maximum beneath the crest. The profile of the horizontal velocity distribution is also symmetrical about the axis passing through the crest. The vertical velocity is maximum some distance in front of the wave and minimum the same distance after the wave. It is zero beneath the wave crest for all depths and approaches zero asymptotically away from the wave in both directions. The pressure distribution determined from the numerical solution do not exhibit numerical oscillations just as the contours of the velocity distributions. This gives further evidence that the problem of pressure–velocity decoupling has been eliminated with the use of the high-order solution method.

The distribution of dynamic pressures and velocities at the two separate times, 60.0 and 100.0 s, shows an invariant distribution of all the flow variables. The form of the distribution remains constant with time as the solitary wave propagates over constant depth. There is symmetry along the vertical line passing through the crest of the wave. The larger grid spacing used in the simulations compared with the grid spacing used for second-order solutions goes

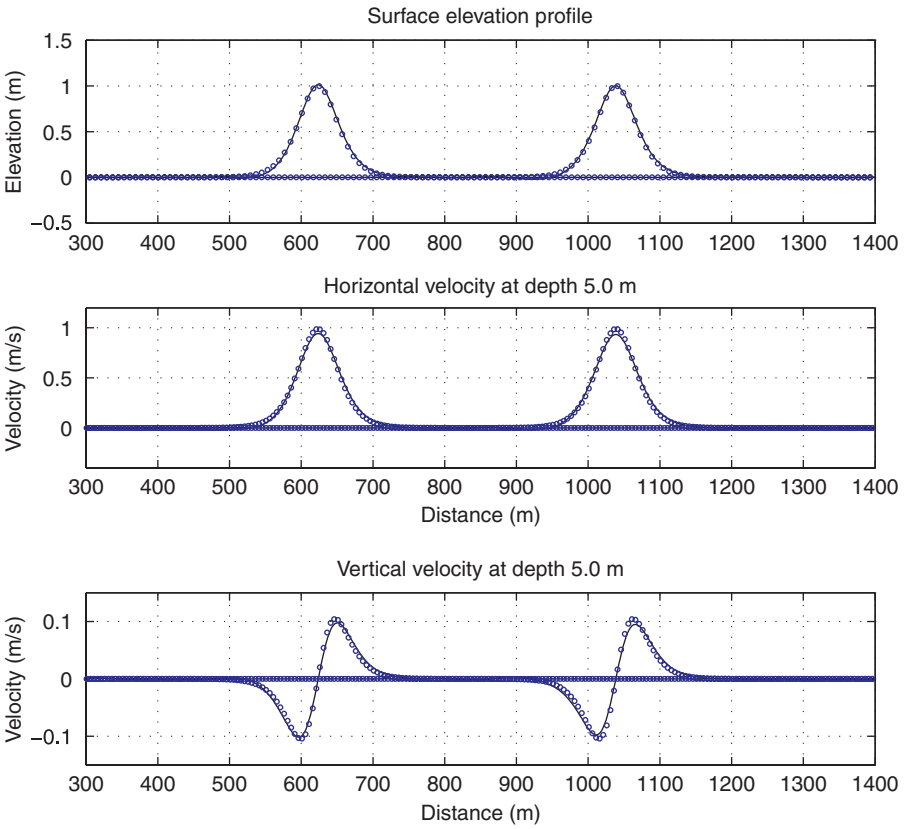

Figure 7. Comparison of analytical (circles) and numerical (solid lines) solutions for surface elevation profile for solitary wave 1.0 m height and depth of 10.0 m (top) at time 60.0 and 100 s, horizontal (middle) and vertical (bottom) velocities are at a depth of 5.0 m.

to demonstrate the efficiency and accuracy of the fourth-order solution method. Numerical dissipation is negligible for all practical purposes.

Wind-driven circulation

The simulation of flow induced by the application of stresses on the free surface by a wind blowing at a constant velocity was carried out next. This numerical experiment involves the diffusive terms unlike the other simulations. A closed basin, 5.0 m in depth and 20.0 m long is assumed. A wind speed of 10.0 m/s is applied to the free surface in the negative x direction until steady-state conditions are attained. The grid spacing used along both the vertical and horizontal axes is 0.5 m and the time-step was 0.1 s. The initial conditions assume the fluid to be at rest. Streamlines determined for the circulation pattern that develops in the flow field at different times 700 and 2000 s are presented in Figure 10. These show the general pattern of circulation before and at the attainment of steady-state conditions, which occurs some time at about 2000 s. The streamlines show smooth changes without oscillations close to the wall.

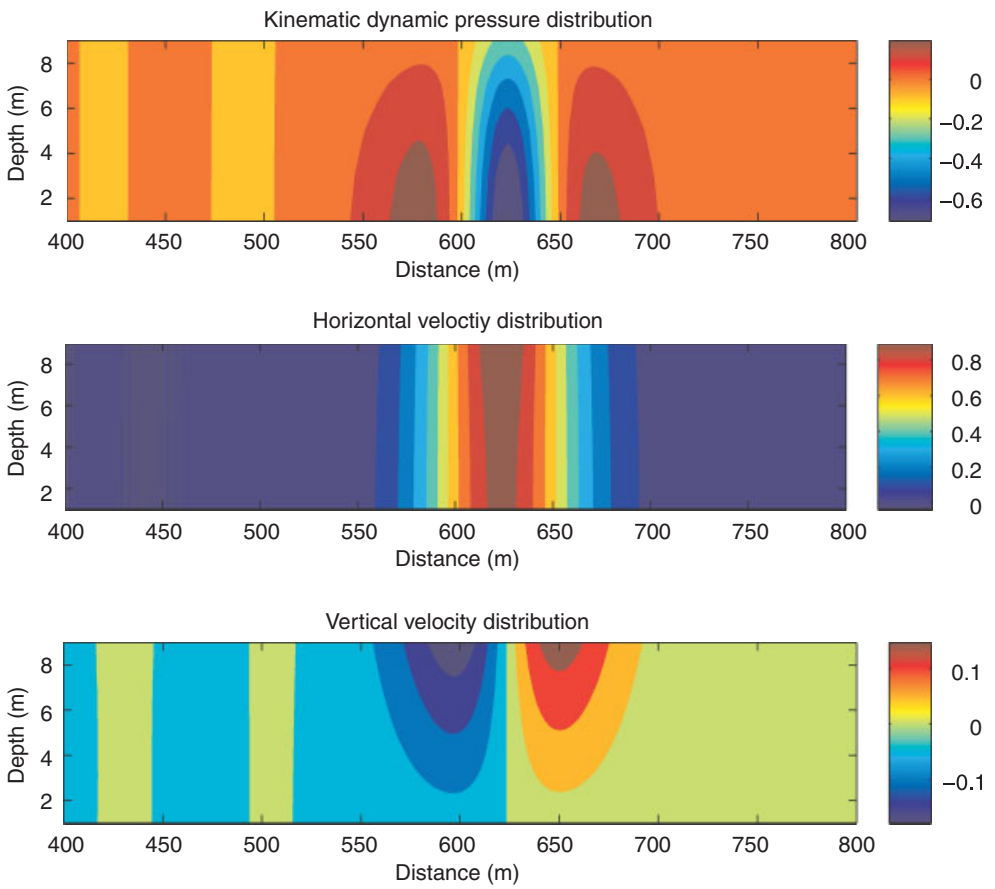

Figure 8. Distribution of kinematic dynamic pressure and velocities under solitary wave at time 60 s (pressures are in m, velocities in m/s).

The velocity field for the time when the flow is steady is presented in Figure 11 corresponding to the time 2000 s. As can be observed, the surface fluid particles accelerate towards the left boundary moving horizontally until close to the boundary when they decelerate and change direction flowing downwards. The bottom fluid particles move in the opposite direction, from the left to right. The flow close to both wall boundaries exhibit a high vertical acceleration that is not reproduced by the model using the hydrostatic approximation.

CONCLUSION

The two simulations involving only the convective terms provide a means of testing the merits of the high-order numerical scheme. These simulations are the standing wave in a basin and the propagation of a solitary wave in a long channel. The availability of analytical solutions make possible comparison with numerical solutions. Coarser grid spacings were

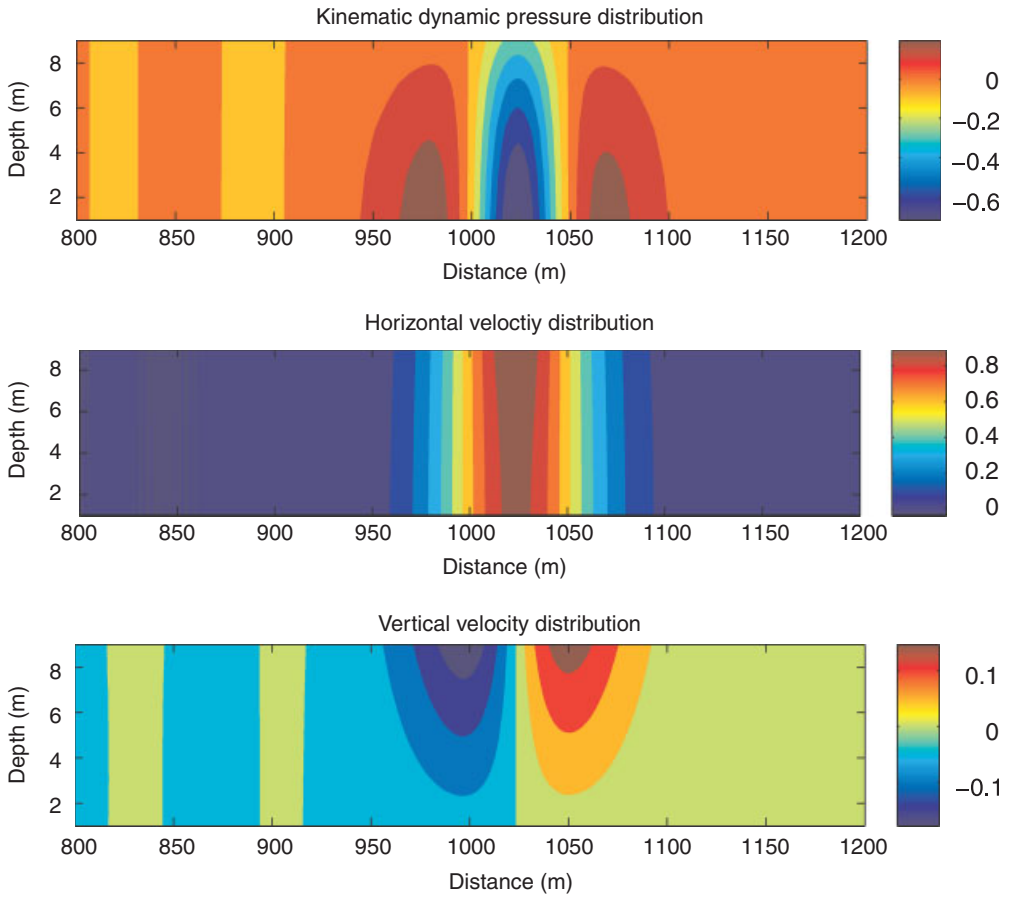

Figure 9. Distribution of kinematic dynamic pressure and velocities under solitary wave at time 100 s (pressures are in m, velocities in m/s).

used in all the simulations compared with typical simulations using second-order accurate methods. Tridiagonal matrices are solved for both first and second derivatives at each stage of the fourth-order Runge–Kutta method. The Poisson equation is solved twice at each time-step. The use of the high-order time integration makes possible convectively stable solutions for larger time-steps. Alternatively, the explicit first-order Euler method requires only one Poisson solve per time-step. However, the time-step necessary to obtain stable solutions is much smaller.

For the standing wave in a basin, the total number of cells used was 100 as compared with 400 that is typically reported in the literature. This has a profound influence on the time taken for the simulations and data storage requirements of the model. The smaller truncation errors associated with the compact high-order scheme also makes possible longer time simulations compared with the second-order accurate methods.

The high resolution characteristics of the high-order numerical scheme yields much smaller phase errors and numerical dissipation is practically non-existent. It is estimated that the

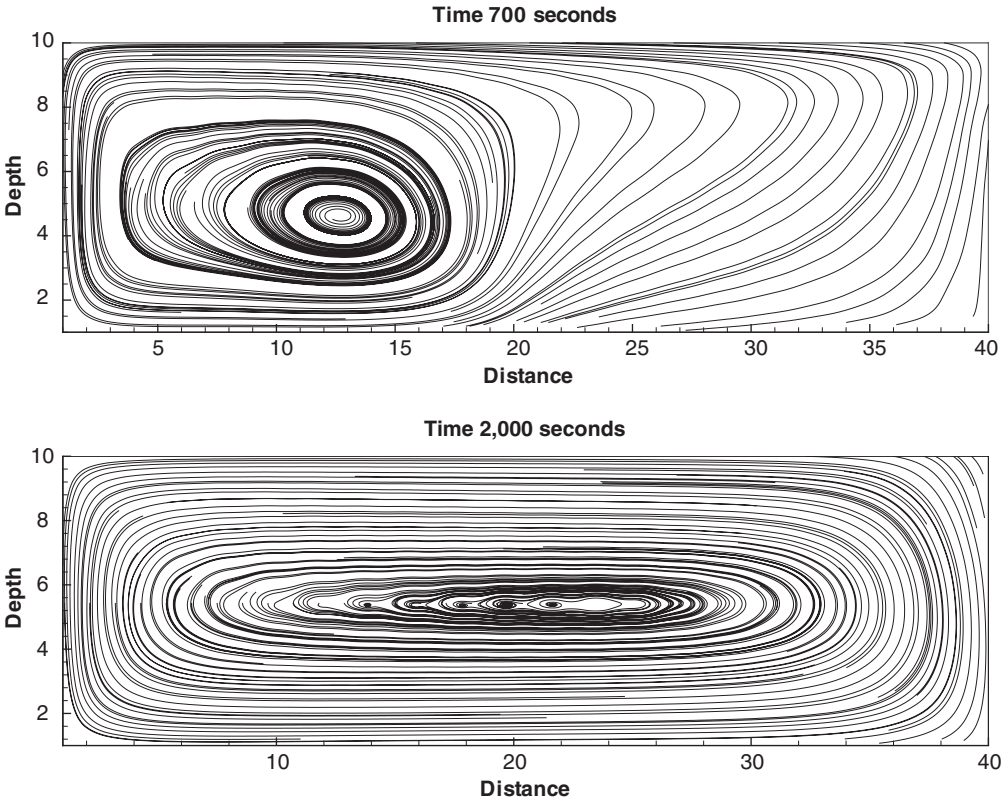

Figure 10. Streamlines for wind-driven circulation at time 700 and 2000 s.

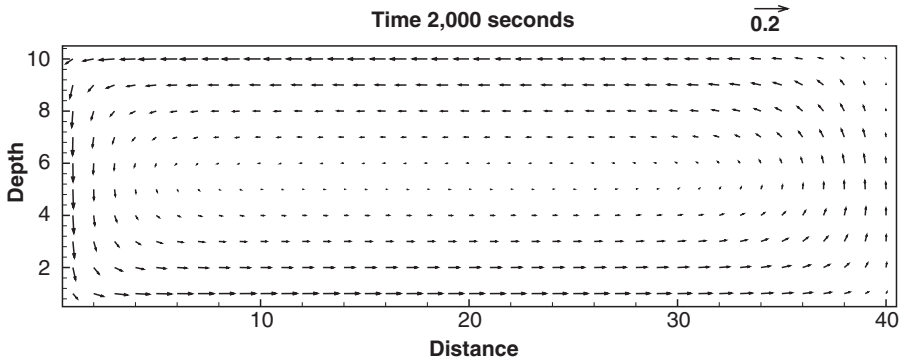

Figure 11. Velocity vectors at steady state (time 2000 s).

phase errors are of the order of 0.027% per wave period. The high-order time integration method was necessary to take advantage of larger time-steps for the simulations and remove the constraints of the cell Reynolds number. The explicit first order in time or the implicit second-order Crank–Nicolson methods are also possible alternatives.

Collocated grids were used to remove some of the complexities of the staggered grid system and facilitate the use of the multigrid method in three-dimensional cases for iterative solution of the pressure Poisson equation. The use of the collocated grid is also advantageous for discretizations using curvilinear coordinates as only one metric of the transformation is required for each cell. This provides a potentially more efficient method than the methods using the staggered grid.

The current numerical scheme is limited by the extent of non-linearity that it can accommodate. This limitation can be removed by changes in the manner in which the application of the dynamic free surface boundary condition is determined. This is the subject of ongoing work.

APPENDIX A

Derivation of pressure boundary conditions

The pressure at the surface is assumed to be atmospheric (i.e. $q=0$). The extrapolation to obtain the pressure outside the computational domain is carried out such that the surface pressure is accounted for (see Figure 2). Assume the pressure q_0 is in the form

$$q_0 = aq_1 + bq_2 + cq_3 \tag{A1}$$

and

$$\beta = \frac{\eta}{\Delta z} \tag{A2}$$

The distance from the water surface to point q_0 is given by

$$z = -(\tfrac{1}{2}\Delta z - \eta) = (\beta - \tfrac{1}{2})\Delta z \tag{A3}$$

Writing Taylor series for q_0, q_1, q_2 and q_3 with base point on the water surface, q_s, gives

$$q_0 = q_s + \left(\beta - \frac{1}{2}\right)\Delta z q_z + \left(\beta - \frac{1}{2}\right)^2 \frac{\Delta z^2}{2!} q_{zz} + \left(\beta - \frac{1}{2}\right)^3 \frac{\Delta z^3}{3!} q_{zzz} + O(z)^4 \tag{A4}$$

$$q_1 = q_s + \left(\beta + \frac{1}{2}\right)\Delta z q_z + \left(\beta + \frac{1}{2}\right)^2 \frac{\Delta z^2}{2!} q_{zz} + \left(\beta + \frac{1}{2}\right)^3 \frac{\Delta z^3}{3!} q_{zzz} + O(z)^4 \tag{A5}$$

$$q_2 = q_s + \left(\beta + \frac{3}{2}\right)\Delta z q_z + \left(\beta + \frac{3}{2}\right)^2 \frac{\Delta z^2}{2!} q_{zz} + \left(\beta + \frac{3}{2}\right)^3 \frac{\Delta z^3}{3!} q_{zzz} + O(z)^4 \tag{A6}$$

$$q_3 = q_s + \left(\beta + \frac{5}{2}\right)\Delta z q_z + \left(\beta + \frac{5}{2}\right)^2 \frac{\Delta z^2}{2!} q_{zz} + \left(\beta + \frac{5}{2}\right)^3 \frac{\Delta z^3}{3!} q_{zzz} + O(z)^4 \tag{A7}$$

The subscripts z, denote partial differentiation. For example

$$q_{zz} = \frac{\partial^2 q}{\partial z^2}$$

By matching the coefficients and noting that $q_s = 0$ the following equations are obtained:

$$\begin{bmatrix} (\beta+\frac{1}{2}) & (\beta+\frac{3}{2}) & (\beta+\frac{5}{2}) \\ (\beta+\frac{1}{2})^2 & (\beta+\frac{3}{2})^2 & (\beta+\frac{5}{2})^2 \\ (\beta+\frac{1}{2})^3 & (\beta+\frac{3}{2})^3 & (\beta+\frac{5}{2})^3 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} (\beta-\frac{1}{2}) \\ (\beta-\frac{1}{2})^2 \\ (\beta-\frac{1}{2})^3 \end{bmatrix} \tag{A8}$$

The above boundary equation is fourth-order accurate. A similar equation can be developed with third-order accuracy which is derived from

$$q_0 = aq_1 + bq_2 \tag{A9}$$

$$\begin{bmatrix} (\beta+\frac{1}{2}) & (\beta+\frac{3}{2}) \\ (\beta+\frac{1}{2})^2 & (\beta+\frac{3}{2})^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} (\beta-\frac{1}{2}) \\ (\beta-\frac{1}{2})^2 \end{bmatrix} \tag{A10}$$

The boundary condition for the velocities normal to the walls (i.e. Dirichlet boundary conditions) can be derived using the above equations for $\beta = 0$. Solving for the coefficients a, b and c gives

$$a = -3.0, \quad b = 1.0, \quad c = -\tfrac{1}{5}, \quad O(\Delta z)^4 \tag{A11}$$

$$a = -2.0, \quad b = \tfrac{1}{3}, \quad O(\Delta z)^3 \tag{A12}$$

The boundary equations for the velocities normal to the walls apply also to the pressure boundary at the walls.

ACKNOWLEDGEMENTS

This work was sponsored in part by the Canadian Commonwealth Scholarship and Fellowship program. The Water Research Institute of the Council for Scientific and Industrial Research, Accra, Ghana also supported the first author in the conduct of this work.

REFERENCES

1. Mayer S, Garapon A, Sørensen LS. A fractional step method for unsteady free-surface flow with applications to non-linear wave dynamics. *International Journal for Numerical Methods in Fluids* 1998; **28**:293–315.
2. Casulli V. A semi-implicit finite difference method for non-hydrostatic, free surface flows. *International Journal for Numerical Methods in Fluids* 1999; **30**:425–440.
3. Li B, Fleming CA. Three-dimensional model of Navier–Stokes equations for water waves. *Journal of Waterway, Port, Coastal, and Ocean Engineering* (ASCE) 2001; **127**:16–25.
4. Kocyigit MB, Falconer RA, Lin B. Three-dimensional numerical modeling of free surface flows with non-hydrostatic pressure. *International Journal for Numerical Methods in Fluids* 2002; **40**:1145–1162.
5. Chen X. A fully hydrodynamic model for three-dimensional, free-surface flows. *International Journal for Numerical Methods in Fluids* 2003; **42**:929–952.
6. Yuan H, Wu C-H. A two-dimensional vertical non-hydrostatic σ model with an implicit method for free-surface flows. *International Journal for Numerical Methods in Fluids* 2004; **44**:811–835.
7. Orszag SA, Israeli M, Deville M. Boundary conditions for incompressible flows. *Journal of Scientific Computing* 1986; **1**:75–111.
8. Harlow F, Welch E. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Computers and Fluids* 1965; **8**:2182–2189.

9. Stelling G. Zijlema M. An accurate and efficient finite-difference algorithm for non-hydrostatic free-surface flow with application to wave propagation. *International Journal for Numerical Methods in Fluids* 2003; **43**:1–23.
10. Peregrine DH. Long waves on a beach. *Journal of Fluid Mechanics* 1967; **27**:815–827.
11. Gustafsson B. The convergence rate for difference approximations to mixed initial boundary value problems. *Mathematics of Computation* 1975; **29**(130):396–406.
12. Collatz L. *The Numerical Treatment of Differential Equations* (3rd edn). Springer-Verlag: New York, 1960.
13. Lele SK. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics* 1992; **103**:16–42.
14. Hirsh R. Higher order accurate difference solutions of fluid mechanics problems by a compact differencing technique. *Journal of Computational Physics* 1975; **19**:90–109.
15. Anthonio SL. High-order compact discretization scheme for non-hydrostatic free surface flows. *Ph.D. Thesis*, Queen's University, Kingston, Ontario, Canada, 2005.
16. Roache PJ. *Fundamentals of Computational Fluid Dynamics*. Hermosa Publishers: 1998.
17. Weinan E, Liu J-G. Essentially compact schemes for unsteady viscous incompressible flows. *Journal of Computational Physics* 1996; **126**:122–138.
18. Dean RG, Dalrymple RA. *Water Wave Mechanics for Engineers and Scientists*. Prentice-Hall: Englewood Cliffs, NJ, 1984.
19. Johnston H, Liu J-G. Finite difference schemes for incompressible flow based on local pressure boundary conditions. *Journal of Computational Physics* 2002; **180**:120–154.
20. Laitone EV. The second approximation to cnoidal and solitary waves. *Journal of Fluid Mechanics* 1960; **9**:430–444.

