INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **48**:169–197
Published online 3 December 2004 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.821

Further experiences with computing non-hydrostatic free-surface flows involving water waves

Marcel Zijlema[1,2,*,†] and Guus S. Stelling[1]

[1]*Environmental Fluid Mechanics Section, Faculty of Civil Engineering and Geosciences,*
Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands
[2]*National Institute for Coastal and Marine Management/RIKZ, P.O. Box 20907,*
2500 EX The Hague, The Netherlands

SUMMARY

A semi-implicit, staggered finite volume technique for non-hydrostatic, free-surface flow governed by the incompressible Euler equations is presented that has a proper balance between accuracy, robustness and computing time. The procedure is intended to be used for predicting wave propagation in coastal areas. The splitting of the pressure into hydrostatic and non-hydrostatic components is utilized. To ease the task of discretization and to enhance the accuracy of the scheme, a vertical boundary-fitted co-ordinate system is employed, permitting more resolution near the bottom as well as near the free surface. The issue of the implementation of boundary conditions is addressed. As recently proposed by the present authors, the Keller-box scheme for accurate approximation of frequency wave dispersion requiring a limited vertical resolution is incorporated. The both locally and globally mass conserved solution is achieved with the aid of a projection method in the discrete sense. An efficient preconditioned Krylov subspace technique to solve the discretized Poisson equation for pressure correction with an unsymmetric matrix is treated. Some numerical experiments to show the accuracy, robustness and efficiency of the proposed method are presented. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: water waves; non-hydrostatic; finite volume; vertical boundary-fitted co-ordinate; semi-implicit; pressure correction

1. INTRODUCTION

Recently, we have proposed an improved non-hydrostatic, free-surface flow model governed by the incompressible Euler equations for simulating short waves with linear dispersion [1]. This model utilizes an edge-based compact difference scheme known as the Keller-box scheme [2] for the approximation of vertical gradient of the non-hydrostatic pressure located at the interfaces between layers. Such a scheme allows straightforward implementation of the zero

[*] Correspondence to: M. Zijlema, Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands.
[†] E-mail: m.zijlema@citg.tudelft.nl

Received 10 October 2003
Revised 19 June 2004
Accepted 14 October 2004

pressure boundary condition at the free surface without the need for special attention at interior points near that surface. Moreover, the discretization error is four to six times smaller than the error of common finite differences of the same order and involving the same number of vertical grid points. As a result, accurate wave characteristics can be obtained with just one or two vertical layers as shown in Reference [1], whereas with common differences at least 10–20 layers are required in order to resolve the dispersion of a short wave correctly. However, we have formulated our method in Cartesian co-ordinates and integrated the governing equations in time in an explicit manner. This may not be very efficient since the applications of our interest often involve both deep and shallow water in coastal regions. As a consequence, time steps might become very small due to the CFL condition related to the depth. Furthermore, because of the relatively large variation in the water depth (a few tenfold meters to a few meters), efficient simulations in the Cartesian framework are seldom the case.

In this paper, we have reformulated our model by employing the terrain-following co-ordinates followed by a semi-implicit time stepping so that an efficient and stable solution can be obtained. However, the implementation of a compact differencing scheme within a semi-implicit framework is considered to be much more complicated because of its inherent implicitness. For example, the Keller-box scheme takes the average of the vertical momentum equation at both the upper and lower interface of the same layer (see Equation (20) of our paper [1]). Hence, only one layer is used supporting two degrees of freedom.[‡] Combined with the semi-implicit time stepping, this will result in a non-sparse system of equations and thus, involve prohibitive computer time and storage. In this paper, it will be demonstrated how to circumvent this problem.

Before we continue, a review of non-hydrostatic computing will be given below. A number of studies have been reported since the second half of the 1990s in which the features of, among others, time integration and co-ordinate systems in the vertical direction have been investigated. These issues are presented from the perspective of constructing our numerical method to follow.

The numerical solution of the Euler or Navier–Stokes equations is often obtained by means of the fractional step procedure. Examples can be found in References [3–9]. This procedure consists of two steps and exploits the decoupling of the pressure into hydrostatic and non-hydrostatic parts. During the first step, the free-surface condition and momentum equations without the non-hydrostatic pressure are solved. In the second step, the achieved velocity field is corrected by means of the non-hydrostatic pressure of which its Poisson-like equation is obtained by taking the divergence of the momentum equations and employing the condition of a divergence-free velocity field. A difficulty with this procedure, however, is that it introduces a splitting error, since the advection and the pressure gradient do not commute. Therefore, the model is only first order accurate in time. As will be demonstrated in this paper, this affects adversely the wave propagation as the waves damped significantly.

A number of approaches have been proposed to eliminate or at least to reduce the splitting error. In References [10, 11], the method of Casulli and Stelling [3] is improved by considering a correction for the water level during the second fractional step. This correction is based on the assumption of the hydrostatic pressure in each computational cell below the

[‡] This will not imply overdetermination of the system of equations in our case since, the vertical velocity at the bottom is prescribed.

surface level. We come back to this point later. In the same step, the free-surface condition is reconsidered to assure global mass conservation. From a computational point of view, this improvement implies no requirement of the first fractional step. Hence, this method has a reduced splitting error. An alternative is proposed by Chen [12], where the pressure Poisson equation is first solved to obtain the non-hydrostatic pressure and subsequently correct the velocities. Thereafter, the surface elevation followed by the velocity field is updated. However, the momentum equations must be solved twice per time step, whereas usually the momentum equations are solved once per time step as done in References [3–11].

In this paper, a projection method, known as pressure correction technique, is presented. One of the aim of this paper is to show that this technique does not contain a splitting error and thus enables to compute propagating waves without any damping. Some earlier publications taking this route are [13–15]; it should be emphasized that the method of Koçyigit *et al.* [15] is an exact 3D extension of the 2DV flow model of Stansby and Zhou [13]. Pressure correction methods in combination with a time-marching method are very efficient for solving the unsteady Navier–Stokes equations and there is a vast amount of literature available on this subject; for an overview, see Reference [16] and the references quoted there. An important advantage of such methods is that they are generally second order accurate in time, as has been proved by Van Kan [17] and demonstrated in Reference [18]. While the fractional step method is based on a full splitting of the treatment of the pressure and advection in different substeps, the pressure correction method consists of a predictor–corrector procedure between the velocity and pressure fields. The approach is as follows. Firstly, an estimate of the velocity field is achieved by means of solving the momentum equations that contain the non-hydrostatic pressure at the preceding time level. This is fortunate, because the surface elevation will be directly affected by the non-hydrostatic pressure. Secondly, the pressure correction, i.e. the difference between the new and old non-hydrostatic pressure, is computed by means of solving the discretized Poisson equation. This equation is obtained by taking the divergence of the discretized momentum equations and subsequently employing the incompressibility constraint. Lastly, through the pressure correction, the intermediate velocity is corrected, resulting in a divergence-free velocity field. It must be stressed that first the space discretization is carried out and thereafter the pressure correction and so to obtained the discrete Poisson equation. Some researchers first derived the continuous Poisson equation after which the space discretization is applied; see References [5, 7, 8, 12, 19–21]. The consequence of this is the need to define artificial boundary conditions for the pressure. We come back to this issue later.

Computing non-hydrostatic, free surface flow within an implicit framework can also be carried out without splitting the pressure into hydrostatic and non-hydrostatic parts; this is done in References [19–25]. Some of them, like [22–25], do not even use a fractional step or pressure correction technique. The method as proposed in References [22, 23] is simply based on integrating the vertical momentum equation to obtain the pressure and subsequent substituting it in the horizontal momentum equations, after which they are solved. The vertical velocity is determined from the continuity equation. This process is iterated until convergence. This approach, however, is likely to suffer from instability when the vertical acceleration is relatively strong because of the lack of direct coupling between the incompressibility constraint and the pressure. An alternative method is presented in References [24, 25] where the free-surface condition and the momentum equations are solved simultaneously for 2D vertical plane problems. This approach provides a block tri-diagonal system of equations for the horizontal velocity that can be readily solved by a direct method. Indeed, this is the case when central differences are employed for the convection terms in the *x–z* plane. The approach would thus appear to be cumbersome when higher order upwind schemes are used, especially for 3D computations.

The methods using a fractional step technique involving non-splitted pressure proposed in the publications [19–21] are probably less efficient than those that exploit the decomposition of the pressure. This has been demonstrated in Reference [4]. Moreover, we think it is desirable to bring out explicitly the role of the non-hydrostatic pressure. In fact, it is a correction to the hydrostatic pressure and relatively small compared to the latter. Due to the splitting round-off errors can be avoided to a large extent and hence, the pressure gradients can be computed more accurately.

Another point of discussion is the use of the artificial boundary conditions for the (non-hydrostatic) pressure, as presented in References [3–5, 7–15, 19, 20]. In fact, for an incompressible flow no boundary conditions for the pressure are required and our method follow here will not need them either. This feature is closely connected with the fact that there is no equation of state. The pressure acts as a Lagrange multiplier in order to keep the flow always and everywhere incompressible. The issue of boundary conditions for the pressure Poisson equation is discussed extensively in Reference [26]. Furthermore, the handling of the zero pressure boundary condition at the free surface need to be addressed. This condition stems from the continuity of the normal stresses at the interface between air and water (apart from the surface tension). As a consequence, the normal velocity component at the surface can be found only by means of the solution of its momentum equation. This approach is already implemented in our method [1]. The usual treatment, however, is to set the non-hydrostatic pressure in the surface cells to zero after which the normal velocity component is derived from the continuity equation [3, 9–11, 13–15, 24]. However, we shall show that, contrary to the former approach, the latter one will lead to an incorrect wave celerity (see also Reference [25]).

So far, no word on time integration is said. Stable schemes can be constructed by means of semi-implicit time stepping. For example, following Casulli [10], the horizontal gradients in the momentum equations and the free-surface condition are discretized by the so-called θ-method, whereas the vertical viscosity terms are integrated implicitly. As a consequence, unconditionally stability is achieved with respect to the celerity of gravity waves and vertical viscosity terms. This is the approach that we shall follow and is also presented in References [4, 12–15, 24, 25]. In Reference [8], an alternating direction implicit (ADI) scheme is employed to obtain the hydrostatic pressure. In References [5, 7, 19–21], an explicit time discretization is used so that the time step is restricted by the deepest part of the coastal area or the smallest grid size.

The applications of our interest often cover areas where the bathymetry may vary rapidly and the free surface vary as a function of time. Hence, the local water depth depends both on the horizontal position and time. In the literature, often two co-ordinate systems in the vertical direction are proposed, namely the Cartesian and σ-co-ordinate. On the question of whether one of these systems are preferable the last word has not yet been said. In References [3, 5, 6, 9, 10, 12] a Cartesian mesh has been employed, whereas in References [4, 7, 8, 13–15, 19, 22–25] the σ-co-ordinate has been adopted. Our purpose here is to develop an efficient and accurate algorithm that enables to simulate water waves propagating from deep water through the surf zone. We shall therefore use a vertical boundary-conforming co-ordinate system of which the σ-transformation is a special case. This system allows defining a number of layers having a uniform thickness for each layer. In this way, large horizontal differences in the magnitude of bottom stresses is prevented. Moreover, the numerical error caused by unequal vertical grid spacing near a bottom with steep slope is minimized.

It is well-known that some inaccuracies may occur in σ-co-ordinates through the evaluation of horizontal gradients near a steep bed and to avoid this these are calculated in physical space following Stelling and Van Kester [27]. This has led some investigators to choose retaining the horizontal pressure gradient terms in Cartesian co-ordinates; see References [13–15]. Nevertheless, we argue to take into account the curvature terms in the horizontal pressure gradient due to σ-transformation since, experiences show that these terms appear not to be sensitive to the relative large changes in bathymetry. Other publications in this direction are [4, 7, 8, 19].

Because of the application of vertical boundary-fitted co-ordinate system, only non-symmetric pressure Poisson systems are involved. A remarkable claim of Lin and Li [19] which stated that they have derived a symmetric positive definite Poisson matrix in spite of the use of σ-co-ordinate. The numerical solution of the Poisson equation for the pressure correction is a crucial step of the whole approach, since the overall efficiency of the numerical code will depend on its performance. Hence, iterative solution methods with possibly acceleration techniques should be applied. Usually, the well-known efficient and robust preconditioned conjugate gradient method (see, e.g. Reference [16]) works fine, if and only if the discretization of the Poisson equation results in linear system of equations with a symmetric positive definite matrix. This is the case, for instance, when the Cartesian co-ordinates are employed. Examples can be found in References [3, 5, 6, 9, 10]. In our method and Reference [12], a very popular Krylov subspace method is employed, namely BiCGSTAB [28], appropriate for solving non-symmetric systems. Furthermore, we combined this solver with preconditioners based on incomplete LU decompositions. Special attention is paid to the further optimization of the preconditioning in terms of efficiency and robustness. An alternative to the solution may be a multigrid technique; this is done in References [4, 7, 8, 20]. The papers of Stansby and Zhou [13], Zhou and Stansby [14] and Koçyigit *et al.* [15] do not indicate which iterative method has been employed in their codes.

The outline of this paper is as follows. After formulating the mathematical model in Section 2, we discretize the underlying differential equations with the finite volume method in Section 3. In Section 4, an extensive description of the solution procedure is given. Section 5 presents the results of three numerical experiments to assess the performance of the proposed method in terms of accuracy, robustness and efficieny. We conclude this paper with drawning some conclusions in Section 6.

2. MATHEMATICAL FORMULATION

2.1. Governing equations

We consider a two-dimensional vertical plane that is bounded by the free surface $z=\zeta(x,t)$ and the bottom $z=-d(x)$, where t is time and x and z are Cartesian co-ordinates. See Figure 1. The extension of the approach to three dimensions is straightforward and has been successfully tested. The primitive variables are the velocity components u, w in x- and z-direction, respectively, and the pressure p normalized through division by a constant

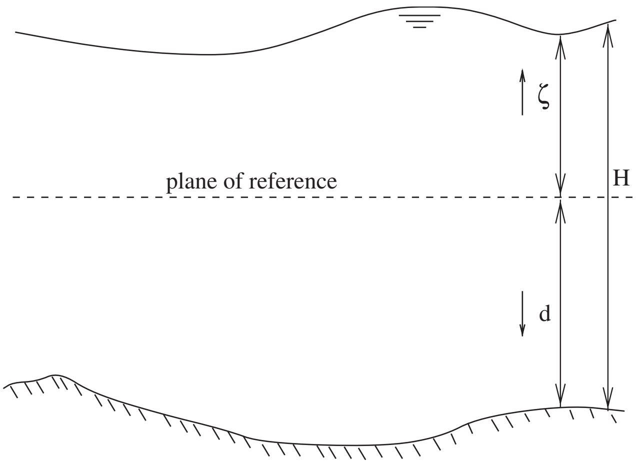

Figure 1. Water area with free surface and bottom.

reference density. Furthermore, we split the pressure into hydrostatic and non-hydrostatic ones, as follows:

$$p = g(\zeta - z) + q \tag{1}$$

so that rounding errors can be largely avoided. The quantity q denotes the non-hydrostatic pressure and g is the acceleration of gravity. The equations of motion that are considered here are

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial wu}{\partial z} + g\frac{\partial \zeta}{\partial x} + \frac{\partial q}{\partial x} = 0 \tag{2}$$

$$\frac{\partial w}{\partial t} + \frac{\partial uw}{\partial x} + \frac{\partial w^2}{\partial z} + \frac{\partial q}{\partial z} = 0 \tag{3}$$

For the sake of clarity, we have omitted effects of turbulence, Coriolis force, atmospheric pressure and baroclinic pressure gradient, because they are not needed for studying the wave propagation. Though, simulation with these effects is feasible; see, e.g. Reference [9].

The equation of continuity for an incompressible fluid is given by

$$\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} = 0 \tag{4}$$

By integrating Equation (4) over the water depth $H = \zeta + d$ and using the kinematic conditions, given by

$$w|_{z=\zeta} = \frac{\partial \zeta}{\partial t} + u\frac{\partial \zeta}{\partial x} \quad w|_{z=-d} = -u\frac{\partial d}{\partial x} \tag{5}$$

the free-surface condition is obtained:

$$\frac{\partial \zeta}{\partial t} + \frac{\partial}{\partial x} \int_{-d}^{\zeta} u \, \mathrm{d}z = 0 \tag{6}$$

2.2. Boundary conditions

To get a unique solution, boundary conditions are required at all boundaries of the physical domain considered. We distinguish four type of boundaries: (i) free surface, (ii) bottom, (iii) open and (iv) closed boundaries. Only one normal and one tangential component of the velocity and/or stress need to be described at these boundaries. The stress that is described on a boundary is the summation of the pressure p and the viscous stress tensor. Since, we are dealing with irrotational flows, the latter part of the stress will not be considered in this paper. Since, the pressure p is a part of the stress and the fact that it is not a thermodynamic variable, there is no need to prescribe the pressure explicitly on the appropriate boundaries.

We present the following boundary conditions for each type of boundary.

i. *Free surface*
The continuity of normal and tangential stresses is enforced. Concerning the tangential stress, it equals the wind stress and is thus neglected here. The normal stress is prescribed as follows

$$-p|_{z=\zeta} = 2\frac{\gamma}{R} \tag{7}$$

where γ is the surface tension and R is the radius of surface curvature. Usually, in the motions of gravity waves the surface tension can be neglected. By virtue of (1), we have

$$q|_{z=\zeta} = 0 \tag{8}$$

It has to be stressed here that the normal velocity w at the surface is obtained by solving its momentum equation (3). Because of the vertical gradient of non-hydrostatic pressure occuring in that equation, one is enable to implement condition (8). Some researchers [3, 9–11, 13–15, 24], however, assume the surface cells to be hydrostatic, i.e. $q = 0$ inside such cells, and accordingly the component w is determined by applying the continuity equation (4). The side effect of this is a full hydrostatic computation when one layer is taken. In case of more than one layer, it will lead to incorrect phase velocity, as will be demonstrated in Section 5.1; see also Reference [25].

ii. *Bottom*
The normal velocity and the tangential stress are prescribed. The normal velocity, i.e. w, is imposed through the kinematic condition (5). The tangential stress is the bottom stress and neglected in our case.

iii. *Open boundaries*
A distinction is made between the inflow and outflow boundary.

- At inflow, usually the velocities are known, i.e. the normal component equals the incident wave celerity available from analysis or measurements. Furthermore, the tangential velocity component is set to zero. An example can be found in Reference [1]. Since, no momentum equations are solved at the regarding boundary, there is no need to prescribe both the water level and non-hydrostatic pressure.

However, in some cases, instead of the velocity, the surface elevation should be specified at the inflow. In fact, the normal stress is prescribed. Here, it is implicitly assumed that $q=0$, which is often the case. Also, the tangential stress is set to zero.

- At outflow, the normal stress and the tangential velocity are imposed. Again, it is assumed that near the outflow the flow is hydrostatic. As a consequence, the water level is then prescribed. Often, the prescription of the water level and tangential velocity is based on the so-called Sommerfeld's radiation condition, which allows the waves to cross the outflow boundaries without reflections. This condition is given by

$$\frac{\partial f}{\partial t} + c\frac{\partial f}{\partial x} = 0 \tag{9}$$

where f represents the surface elevation and the tangential velocity components and c is the wave phase velocity vector. Usually, $c=\sqrt{gH}$. Treatment of this type of condition and other alternatives to absorbing-generating boundary conditions can be found in Reference [29].

iv. *Closed boundaries*
Both the normal velocity and the tangential stress are set to zero. This type of boundary conditions is also known as the free-slip condition.

3. FINITE VOLUME DISCRETIZATION

The physical domain can be discretized by subdivision of the continuum into cells of arbitrary shape and size. A structured grid is employed, which means that each interior cell is surrounded by the same number of cells. Application to unstructured grids can be found in Reference [11]. A distinction is made between the definition of the grid in the horizontal and vertical direction. In the vertical direction, the computational domain is divided into a fixed number of layers in a such a way that both the bottom topography and the free surface can be accurately represented. This will be outlined in the next section. In the horizontal planes, Cartesian or boundary-fitted co-ordinates can be considered. For the sake of clarity, we restrict ourselves to the Cartesian co-ordinates. Extension to a boundary-fitted co-ordinate system is straightforward, see, e.g. Reference [16]. A horizontal Cartesian grid containing I cells is given by

$$\{\mathbf{x} \,|\, x_{i+1/2} = i\Delta x, \quad i=0,\ldots,I\} \tag{10}$$

with Δx the length of the cell.

3.1. Vertical grid schematization

In the vertical direction, the physical domain is divided into K layers, see Figure 2. The interface between two layers is defined as

$$z_{k+1/2} = z_{k+1/2}(x,t), \quad k=0,\ldots,K \tag{11}$$

Note that $z_{1/2}=-d$ and $z_{K+1/2}=\zeta$. The layer thickness h_k may be defined in a relative way, i.e. a constant part of the water depth similar to the σ-co-ordinate, or in an absolute way,

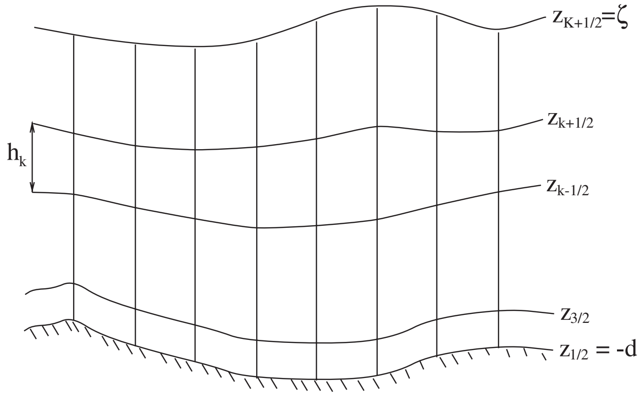

Figure 2. Vertical grid definition with layer interfaces.

i.e. a constant layer thickness. To make sure that the sum of the layer thicknesses equals the water depth, as least one layer must be defined in a relative way. Let us consider a layer k having a constant thickness h_k. The layer interface can be computed as

$$z_{k+1/2} = z_{k-1/2} + h_k \tag{12}$$

If the layer k has a relative thickness, given by a fraction f_k, then the layer interface is determined as follows

$$z_{k+1/2} = z_{k-1/2} + f_k\,(H - h_c) \tag{13}$$

where $H = \sum_{k=1}^{K} h_k$ and h_c is the sum of all constant layer thicknesses. However, in regions of tidal flats it is possible that $H < h_c$. In such a case we set the sum of all constant layer thicknesses exactly to half the water depth H. The remaining half of the depth is partitioned in the same relative fashion. Thus, we have

$$z_{k+1/2} = z_{k-1/2} + \beta_k H \quad \text{if } \; H \leqslant 2h_c \tag{14}$$

with

$$\beta_k = \begin{cases} h_k/2h_c & \text{if layer } k \text{ has a constant thickness } h_k \\ \dfrac{1}{2} f_k & \text{if layer } k \text{ has a relative thickness } h_k \end{cases} \tag{15}$$

A drying and flooding procedure as described in Reference [30] is used to prevent the water depth H to become negative.

The vertical grid schematization gives rise to the definition of the vertical velocity with respect to the moving layer interfaces. The vertical velocity relative to layer interface $z_{k+1/2}$, denoted as $\omega_{k+1/2}$, is defined as the difference between the vertical velocity along the streamline and the vertical velocity along the interface, as follows

$$\omega_{k+1/2} = w(z_{k+1/2}) - \frac{\partial z_{k+1/2}}{\partial t} - u(z_{k+1/2})\frac{\partial z_{k+1/2}}{\partial x} \tag{16}$$

By virtue of Equation (5), we have

$$\omega_{1/2} = \omega_{K+1/2} = 0 \tag{17}$$

The layer-integrated horizontal velocity defined as

$$u_k = \frac{1}{h_k} \int_{z=z_{k-1/2}}^{z_{k+1/2}} u \, \mathrm{d}z \tag{18}$$

is considered as unknown. The horizontal velocity at layer interface $z_{k+1/2}$, as given in Equation (16), can be approximated, as follows:

$$u(z_{k+1/2}) \approx \frac{u_k h_{k+1} + u_{k+1} h_k}{h_k + h_{k+1}} \tag{19}$$

This approximation is sufficiently accurate, since the magnitude of first order term is expected to be very small. This vertical interpolation will be abbreviated by means of the over bar notation, as follows

$$\overline{u}^z_{k+1/2} = u(z_{k+1/2}) \tag{20}$$

The flow rate in x-direction can be approximated as

$$Q^x = \int_{z=-d}^{\zeta} u \, \mathrm{d}z = \sum_{k=1}^{K} h_k u_k \tag{21}$$

3.2. Location of grid variables

A staggered grid arrangement is used in which the velocity components u and w are located at the centre of the cell faces $(i + 1/2, k)$ and $(i, k + 1/2)$, respectively. The water level ζ is located at (i). Concerning the non-hydrostatic pressure q, two ways to assign this unknown to grid points are employed. This variable can be given either at the cell centre (i, k) or at the face $(i, k + 1/2)$. The choice depends on the discretization of the vertical pressure gradient, namely, explicit central differences referring as the classical case (similar to the horizontal counterpart) and the implicit Keller-box scheme [2], respectively. Since, this paper deals with the application to wave propagation, only the latter discretization will be considered. The former approximation is particularly meant for applications where vertical structures are important, e.g. stratified flows with density currents and flows over steep and rapidly varying bottoms, see, e.g. Reference [9]. Figure 3 shows the possible arrangements of the unknowns in the present model.

For each velocity component we define a collection of a finite number of non-overlapping control volumes that covers the whole domain. Each velocity component is at the centre of its control volume. Space discretization is based on the integration of the governing equations over the corresponding control volumes. The conservative property of this finite volume scheme is essential for the accurate calculation of higher order non-linear wave effects. The approach to follow is first integration over a layer k using the Leibniz' rule and thereafter, integration over a horizontal cell area followed by the application of the Gauss divergence theorem. Furthermore, the remaining integrals as well as integration terms containing, e.g. time derivatives, are approximated by means of the midpoint rule.

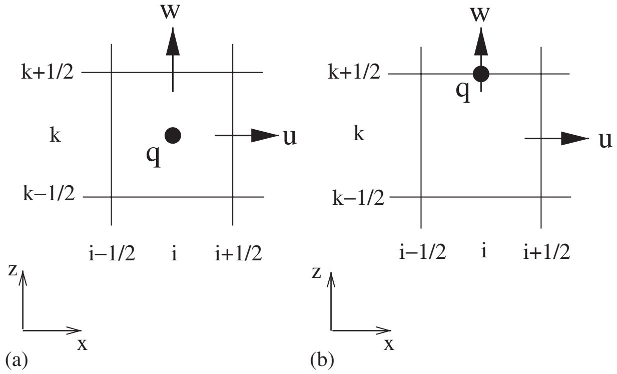

Figure 3. Applied arrangements of the unknowns in a staggered grid: (a) classical case; and (b) special case due to Keller-box scheme.

Unknowns not present at points where they are required are computed by interpolation using the fewest number of interpolation points. So, interpolation in the vertical is indicated by Equation (20), whereas $\overline{\phi}^{x}_{i,k}$ indicates arithmetic averaging of the unknown ϕ in x-direction over their two points of definition that are nearest to (i,k). In analogy, $\overline{\phi}^{xz}_{i,k}$ gives average value of ϕ at (i,k) resulted from the combination of two one-dimensional interpolation formulas in each direction. One such interpolation formula is Equation (20).

3.3. Space discretization of the continuity equation and free-surface condition

The layer-integrated continuity equation (4) for layer $1 \leqslant k \leqslant K$ is obtained as follows:

$$\int_{z_{k-1/2}}^{z_{k+1/2}} \left(\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} \right) \mathrm{d}z = \frac{\partial h_k u_k}{\partial x} - u \left. \frac{\partial z}{\partial x} \right|_{z_{k-1/2}}^{z_{k+1/2}} + w_{k+1/2} - w_{k-1/2} = 0 \tag{22}$$

Although, we could have simplified Equation (22) using Equation (16), i.e. removing the term $u\partial z/\partial x$ while adding $\partial h_k/\partial t$ and subsequent replacing $w_{k\pm 1/2}$ by $\omega_{k\pm 1/2}$, we intend here not to do so. The reason for this becomes clear in Section 3.5. Next, Equation (22) is further integrated over a horizontal cell with centre (i), giving

$$\begin{aligned} & h_{i+1/2,k} u_{i+1/2,k} - h_{i-1/2,k} u_{i-1/2,k} \\ & \quad - \overline{u}^{xz}_{i,k+1/2} (z_{i+1/2,k+1/2} - z_{i-1/2,k+1/2}) \\ & \quad + \overline{u}^{xz}_{i,k-1/2} (z_{i+1/2,k-1/2} - z_{i-1/2,k-1/2}) \\ & \quad + (w_{i,k+1/2} - w_{i,k-1/2}) \Delta x = 0 \end{aligned} \tag{23}$$

The total number of variables linked together in Scheme (23) is 6 u and 2 w. Thus, the present discretization lead to a denser system than the usual one obtained in the Cartesian co-ordinate system (2 u and 2 w), see, e.g. Reference [10].

The free-surface condition (6) is discretized as follows:

$$\frac{\partial \zeta_i}{\partial t}\Delta x + Q^x_{i+1/2} - Q^x_{i-1/2} = 0 \tag{24}$$

with the flow rate Q^x given by Equation (21).

3.4. Space discretization of the horizontal momentum equations

First, the horizontal momentum equations (2) is integrated over layer k and thereafter integrated over horizontal cell $(i + 1/2)$. We consider the space discretization of each of the following contributions separately: (i) time derivative, (ii) advective terms, (iii) surface level gradient and (iv) non-hydrostatic pressure gradient.

Using the Leibniz' rule, integration of the time derivative over layer k gives

$$\int_{z_{k-1/2}}^{z_{k+1/2}} \frac{\partial u}{\partial t}\,\mathrm{d}z = \frac{\partial h_k u_k}{\partial t} - u\,\frac{\partial z}{\partial t}\bigg|_{z_{k-1/2}}^{z_{k+1/2}} \tag{25}$$

The layer-integrated advective terms are obtained as follows:

$$\int_{z_{k-1/2}}^{z_{k+1/2}} \left(\frac{\partial u^2}{\partial x} + \frac{\partial wu}{\partial z}\right)\mathrm{d}z = \frac{\partial}{\partial x}\int_{z_{k-1/2}}^{z_{k+1/2}} u^2\,\mathrm{d}z$$

$$+\overline{u}^z_{k+1/2}\left(\omega_{k+1/2} + \frac{\partial z_{k+1/2}}{\partial t}\right) - \overline{u}^z_{k-1/2}\left(\omega_{k-1/2} + \frac{\partial z_{k-1/2}}{\partial t}\right) \tag{26}$$

in which Equation (16) has been substituted. One can verify that

$$\int_{z_{k-1/2}}^{z_{k+1/2}} u^2\,\mathrm{d}z = h_k u_k^2 + \int_{z_{k-1/2}}^{z_{k+1/2}} (u - u_k)^2\,\mathrm{d}z \tag{27}$$

The integral term in the right-hand side of Equation (27) is the dispersion term which is due to the vertical non-uniformities of the flow velocity. A common practice is to consider this dispersion effect as diffusion and thus may be neglected in our case.

The surface level gradient is independent of the depth and so, integration over layer k will not have an effect. On the other hand, layer-averaging of the non-hydrostatic pressure gradient is necessary. Recall that q is defined at the layer interfaces. We have

$$\int_{z_{k-1/2}}^{z_{k+1/2}} \frac{\partial q}{\partial x}\,\mathrm{d}z = \frac{\partial}{\partial x}\int_{z_{k-1/2}}^{z_{k+1/2}} q\,\mathrm{d}z - q_{k+1/2}\frac{\partial z_{k+1/2}}{\partial x} + q_{k-1/2}\frac{\partial z_{k-1/2}}{\partial x} \tag{28}$$

with the integral approximated by

$$\int_{z_{k-1/2}}^{z_{k+1/2}} q\,\mathrm{d}z \approx \frac{1}{2}h_k(q_{k+1/2} + q_{k-1/2}) = h_k\overline{q}^z_k \tag{29}$$

Note that the notation of averaging in the latter term in Equation (29) differs from Equation (20) because arithmetic averaging inside a layer is exact.

Putting all the terms together, the layer-integrated u-momentum equation reads

$$\begin{aligned}&\frac{\partial h_k u_k}{\partial t}+\frac{\partial h_k u_k^2}{\partial x}+\bar{u}^z_{k+1/2}\omega_{k+1/2}-\bar{u}^z_{k-1/2}\omega_{k-1/2}\\&\quad+gh_k\frac{\partial\zeta}{\partial x}+\frac{\partial h_k\bar{q}^z_k}{\partial x}-q_{k+1/2}\frac{\partial z_{k+1/2}}{\partial x}+q_{k-1/2}\frac{\partial z_{k-1/2}}{\partial x}=0\end{aligned}\tag{30}$$

This equation is discretized in horizontal plane by integration over a control volume centred at u-point. Using the Gauss divergence theorem and subsequently the midpoint rule, this gives

$$\begin{aligned}&\frac{\partial h_{i+1/2,k}u_{i+1/2,k}}{\partial t}\Delta x+\bar{u}^x_{i+1,k}\phi_{i+1,k}-\bar{u}^x_{i,k}\phi_{i,k}\\&\quad+(\bar{u}^z_{i+1/2,k+1/2}\bar{\omega}^x_{i+1/2,k+1/2}-\bar{u}^z_{i+1/2,k-1/2}\bar{\omega}^x_{i+1/2,k-1/2})\Delta x\\&\quad+g(\zeta_{i+1}-\zeta_i)h_{i+1/2,k}+h_{i+1,k}\bar{q}^z_{i+1,k}-h_{i,k}\bar{q}^z_{i,k}\\&\quad-\bar{q}^x_{i+1/2,k+1/2}(z_{i+1,k+1/2}-z_{i,k+1/2})\\&\quad+\bar{q}^x_{i+1/2,k-1/2}(z_{i+1,k-1/2}-z_{i,k-1/2})=0\end{aligned}\tag{31}$$

with $\phi=hu$ the cell-face value. Note that four pressure points are linked together in Equation (31). Further approximation is needed with respect to the cell-face values. In the literature, often higher order upwind schemes with or without flux limiting are adopted for approximating cell-face values (see, e.g. Reference [16]). Here, central differences are used since, no instabilities are encountered in our numerical experiments. For example, central differencing of $\phi_{i,k}$ is obtained by means of linear interpolation in the following way

$$\phi_{i,k}=\tfrac{1}{2}(\phi_{i-1/2,k}+\phi_{i+1/2,k})\tag{32}$$

Note that Equation (31) describes a momentum balance and thus guarantees conservation of momentum.

3.5. Space discretization of the vertical momentum equation

In the present paper, we discretize the momentum equation for the vertical physical velocity w instead of the relative vertical velocity ω that arise from the vertical grid schematization. The reason for this is to keep the formulation and thereby the discretization simple. Because the pressure correction technique will be employed, to be discussed in Section 4.1, it becomes clear why to retain the vertical physical velocity in the continuity equation (23).

Integration is carried out over a control volume of $w_{i,k+1/2}$. Since, the unknown $w_{i,k+1/2}$ is defined in its location the vertical integral of that unknown over the control volume is evaluated using the midpoint rule:

$$\int_{z_k}^{z_{k+1}} w_{i,k+1/2}\,\mathrm{d}z=h_{i,k+1/2}w_{i,k+1/2}\quad\text{with}\;\; h_{i,k+1/2}=\frac{1}{2}(h_{i,k}+h_{i,k+1})\tag{33}$$

We consider a layer k bounded by the interfaces $z_{k-1/2}$ and $z_{k+1/2}$. The intention is to use the Keller-box scheme [2] for an accurate discretization in the vertical. The concept of this box scheme consists of two steps. Firstly, we discretize the w-momentum equation at $z_{k-1/2}$ in which the pressure gradient $\partial q/\partial z$ is approximated through forward differencing and subsequent the w-momentum equation at $z_{k+1/2}$ where the approximation of $\partial q/\partial z$ is obtained by means of backward differencing. Secondly, we take the average of the discretized w-momentum equations at $z_{k-1/2}$ and $z_{k+1/2}$ onto the layer k.

The layer-integrated w-momentum equation can be derived in exactly the same manner as done for the u-momentum equation except for the pressure gradient, and is given by

$$\frac{\partial h_{k+1/2} w_{k+1/2}}{\partial t} + \frac{\partial h_{k+1/2} \overline{u}^z_{k+1/2} w_{k+1/2}}{\partial x} + \overline{w}^z_{k+1} \overline{\omega}^z_{k+1} - \overline{w}^z_k \overline{\omega}^z_k + \int_{z_k}^{z_{k+1}} \frac{\partial q}{\partial z}\, \mathrm{d}z = 0 \tag{34}$$

It should be noted that the interpolation of w in the vertical advective term is simply arithmetic averaging and thus equivalent to central differences. Next, the integral of pressure gradient is approximated by means of backward differencing, as follows:

$$\int_{z_k}^{z_{k+1}} \frac{\partial q}{\partial z}\, \mathrm{d}z = q(z_{k+1}) - q(z_k) \approx q_{k+1/2} - q_{k-1/2} \tag{35}$$

The w-momentum equation at interface $z_{k-1/2}$ is obtained from Equation (34) by decreasing the indices by 1. However, the integral of pressure gradient is evaluated using forward differencing. This gives

$$\int_{z_{k-1}}^{z_k} \frac{\partial q}{\partial z}\, \mathrm{d}z = q(z_k) - q(z_{k-1}) \approx q_{k+1/2} - q_{k-1/2} \tag{36}$$

Finally, we take the average of the w-momentum equations at interfaces $z_{k-1/2}$ and $z_{k+1/2}$ and thereafter, we integrate the resulting equation over a cell face centred at w-point and employ the Gauss divergence theorem and subsequently the midpoint rule, giving

$$\begin{aligned}
&\frac{1}{2}\left(\frac{\partial h_{i,k+1/2} w_{i,k+1/2}}{\partial t} + \frac{\partial h_{i,k-1/2} w_{i,k-1/2}}{\partial t}\right)\Delta x \\
&\quad + \tfrac{1}{2}[(L_x)_{i,k+1/2} + (L_x)_{i,k-1/2}] \\
&\quad + \tfrac{1}{2}[\overline{w}^z_{i,k+1} \overline{\omega}^z_{i,k+1} - \overline{w}^z_{i,k-1} \overline{\omega}^z_{i,k-1}]\Delta x \\
&\quad + (q_{i,k+1/2} - q_{i,k-1/2})\Delta x = 0
\end{aligned} \tag{37}$$

with

$$(L_x)_{i,k+1/2} = \overline{u}^z_{i+1/2,k+1/2} \overline{hw}^x_{i+1/2,k+1/2} - \overline{u}^z_{i-1/2,k+1/2} \overline{hw}^x_{i-1/2,k+1/2} \tag{38}$$

It must be emphasized that Equation (37) is solved for layers $2 \leqslant k \leqslant K$, i.e. including the free surface and excluding the bottom. Condition (8) can be readily incorporated in Equation (37) for $k = K$ as $q_{i,K+1/2} = 0$. At the bottom ($k = 1$), the kinematic condition (5) is imposed.

4. SOLUTION PROCEDURE

The space discretization yields systems of ordinary differential equations for which the solution can be found by means of time integration. In this paper, time integration defines the shift from time level n to level $n+1$ in two steps, namely the hydrostatic and non-hydrostatic steps. In both steps, a projection method is applied. In order to explain this procedure more clearly, we consider the systems of time-dependent equations after spatial discretization of (6), (4), (2) and (3), as follows:

$$\frac{\mathrm{d}\zeta_i}{\mathrm{d}t} + \frac{\delta Q_i}{\delta x} = 0, \quad Q_i = \sum_{k=1}^{K} h_{i,k} u_{i,k} \tag{39}$$

$$\frac{\delta u_{i,k}}{\delta x} + \frac{\delta w_{i,k}}{\delta z} = 0 \tag{40}$$

$$\frac{\mathrm{d}u_{i,k}}{\mathrm{d}t} + g\frac{\delta \zeta_i}{\delta x} + \frac{\delta q_{i,k}}{\delta x} = F_u \tag{41}$$

$$\frac{\mathrm{d}w_{i,k}}{\mathrm{d}t} + \frac{\mathrm{d}w_{i,k-1}}{\mathrm{d}t} + 2\frac{\delta q_{i,k}}{\delta z} = F_w \tag{42}$$

where F_u and F_w are linear algebraic operators arising from the space discretization of the convective terms occuring in equations for $u_{i,k}$ and $w_{i,k}$, respectively, and $\delta/\delta x$ and $\delta/\delta z$ are linear algebraic operators representing the gradients in x- and z-direction, respectively. Due to the use of the Keller-box scheme, Equation (42) contains two time derivatives for w. It must be stressed that the solution algorithm to follow does not rest on the choice of the type of grids and space discretizations and hence, the set of equations (39)–(42) is useful as a starting point. Note also the omission of half-indices. Instead, we employ the so-called group index convention: ζ-point (i), u-point $(i+1/2,k)$ and w- and q-point $(i,k+1/2)$ within each cell have the same group index (i,k).

Time discretization takes place by explicit time stepping for convection terms and semi-implicit time stepping using the θ-scheme for both surface level and pressure gradients as well as the free-surface condition, as follows:

$$\frac{\zeta_i^{n+1} - \zeta_i^n}{\Delta t} + \theta\frac{\delta Q_i^{n+1}}{\delta x} + (1-\theta)\frac{\delta Q_i^n}{\delta x} = 0 \tag{43}$$

$$\frac{\delta u_{i,k}^{n+1}}{\delta x} + \frac{\delta w_{i,k}^{n+1}}{\delta z} = 0 \tag{44}$$

$$\frac{u_{i,k}^{n+1} - u_{i,k}^n}{\Delta t} + g\left(\theta\frac{\delta \zeta_i^{n+1}}{\delta x} + (1-\theta)\frac{\delta \zeta_i^n}{\delta x}\right) + \theta\frac{\delta q_{i,k}^{n+1}}{\delta x} + (1-\theta)\frac{\delta q_{i,k}^n}{\delta x} = F_u^n \tag{45}$$

$$\frac{w_{i,k}^{n+1} - w_{i,k}^n}{\Delta t} + \frac{w_{i,k-1}^{n+1} - w_{i,k-1}^n}{\Delta t} + 2\left(\theta\frac{\delta q_{i,k}^{n+1}}{\delta z} + (1-\theta)\frac{\delta q_{i,k}^n}{\delta z}\right) = F_w^n \tag{46}$$

For stability, $\frac{1}{2} \leqslant \theta \leqslant 1$. Hence, the time step is not limited by the wave celerity. Note that this approach is conditionally stable due to the explicit discretization of the convective terms.

To obtain the solution of the system of equations (43)–(46), we first proceed the outline of the non-hydrostatic step in Section 4.1 and thereafter the hydrostatic step in Section 4.2. In Section 4.3, the overall solution algorithm will be presented.

4.1. Non-hydrostatic step

First, we consider the non-hydrostatic step, which is based on the pressure correction method (see, e.g. References [16, 17] for details and further references). The essence of the pressure correction algorithm is that Equations (44)–(46) are not solved as they stand, but first a prediction for the intermediate velocity field $(u^*_{i,k}, w^*_{i,k})$ is computed from that system with the non-hydrostatic pressure at the previous time level:

$$\frac{u^*_{i,k}-u^n_{i,k}}{\Delta t}+g\left(\theta\frac{\delta\zeta^*_i}{\delta x}+(1-\theta)\frac{\delta\zeta^n_i}{\delta x}\right)+\frac{\delta q^n_{i,k}}{\delta x}=F^n_u \tag{47}$$

$$\frac{w^*_{i,k}-w^n_{i,k}}{\Delta t}+\frac{w^*_{i,k-1}-w^n_{i,k-1}}{\Delta t}+2\frac{\delta q^n_{i,k}}{\delta z}=F^n_w \tag{48}$$

Equation (48) constitutes a set of bi-diagonal system of equations with unknowns $w^*_{i,k}$, $k=1,\ldots,K$, of which the solution can be found in one step. Note that Equation (47) includes the non-hydrostatic pressure term known from previous time step. Solution of this equation combined with

$$\frac{\zeta^*_i-\zeta^n_i}{\Delta t}+\theta\frac{\delta Q^*_i}{\delta x}+(1-\theta)\frac{\delta Q^n_i}{\delta x}=0 \quad \text{with } Q^*_i=\sum_{k=1}^{K}h^*_{i,k}u^*_{i,k} \tag{49}$$

for the intermediate water level ζ^*_i is characterized as the hydrostatic one and will be given in Section 4.2.

During the non-hydrostatic step, a pressure correction $\Delta q_{i,k}\equiv q^{n+1}_{i,k}-q^n_{i,k}$ is calculated. To find the corresponding correction equation, first Equations (47) and (48) are subtracted from Equations (45) and (46), respectively, resulting in

$$\frac{u^{n+1}_{i,k}-u^*_{i,k}}{\Delta t}+g\theta\frac{(\zeta^{n+1}_i-\zeta^*_i)}{\delta x}+\theta\frac{\delta\Delta q_{i,k}}{\delta x}=0 \tag{50}$$

$$\frac{w^{n+1}_{i,k}-w^*_{i,k}}{\Delta t}+2\theta\frac{\delta\Delta q_{i,k}}{\delta z}=-\frac{w^{n+1}_{i,k-1}-w^*_{i,k-1}}{\Delta t} \tag{51}$$

In order to close the system, we neglect the difference $\zeta^{n+1}_i-\zeta^*_i$. This implies that the intermediate water level will not be corrected. Results of the test cases in this paper reveal that this neglect does not deteriorate the temporal accuracy.

The first term in the right-hand side of Equation (51) is merely due to the application of the Keller-box scheme and will complicate the solution technique for $K>1$ since, it implies the strong coupling of the unknowns $w_{i,k'}$ with $k'=1,\ldots,k-1$. As a consequence, a dense matrix will involved and the use of iterative methods may be impractical. However, we shall demonstrate that the difference $w^{n+1}_{i,k-1}-w^*_{i,k-1}$ can be neglected under certain circumstances. The horizontal and vertical velocity components are related to the horizontal and vertical displacement of a fluid particle, respectively. The orbital motion is elliptic of nature. In very

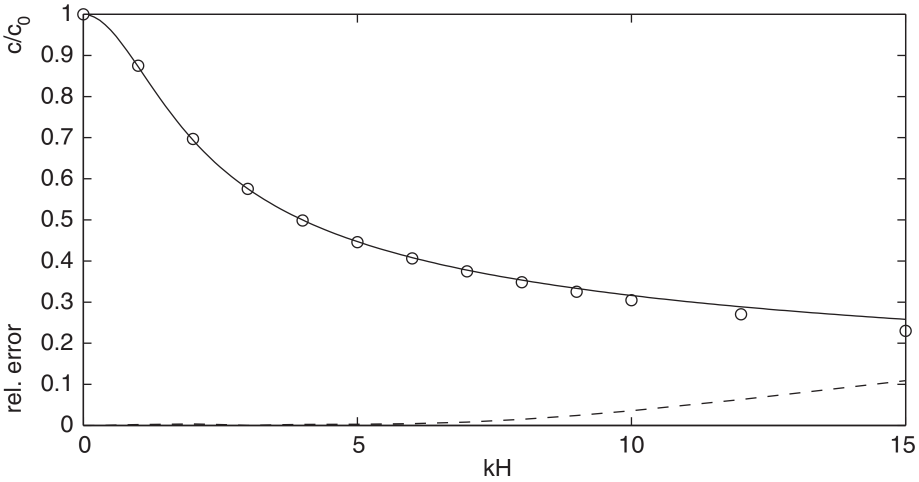

Figure 4. Normalized wave celerity vs relative depth for linear dispersion. Non-hydrostatic model based on Keller-box scheme with two layers (circles), exact (solid line), relative error (dashed line). The quantities c and $c_0 = \sqrt{gH}$ are the wave phase velocity and the long wave celerity, respectively, and the relative error is $|c_{\text{computed}} - c_{\text{exact}}|/c_{\text{exact}}$.

deep water, the orbit becomes a circle and the horizontal and vertical velocities are equally significant. According to linear wave theory, the elliptical orbits progressively become more horizontally stretched with decreasing depth. Hence, it is plausible that the difference $w^{n+1} - w^*$ at layer interface $k - 1/2$ is an order of magnitude smaller than that at layer interface $k + 1/2$, particularly when two or three layers are adopted. Therefore, we shall neglect the difference $w_{i,k-1}^{n+1} - w_{i,k-1}^{*}$ in Equation (51). Based on a rigorous numerical analysis, it appears that this neglect does not affect the modelling of linear dispersion (see also the explanation in Section 5 and Figure 4).

To sum up, the correction equations are now

$$\frac{u_{i,k}^{n+1} - u_{i,k}^{*}}{\Delta t} + \theta \frac{\delta \Delta q_{i,k}}{\delta x} = 0 \tag{52}$$

and

$$\frac{w_{i,k}^{n+1} - w_{i,k}^{*}}{\Delta t} + 2\theta \frac{\delta \Delta q_{i,k}}{\delta z} = 0 \tag{53}$$

Substitution of Equations (52) and (53) into Equation (44) gives a Poisson equation for the pressure correction $\Delta q_{i,k}$. Once $\Delta q_{i,k}$ is obtained, we can calculate $u_{i,k}^{n+1}$ and $w_{i,k}^{n+1}$, respectively, through Equations (52) and (53). This velocity field is divergence-free.

Let denote the discretized divergence matrix, achieved from (44), by D and the discretized pressure gradient matrix, derived from (52) and (53), by G. Furthermore, the pressure correction and the intermediate velocity vectors are indicated by $\Delta\vec{q}$ and $\vec{v^*}$, respectively. Then, the Poisson equation is symbolically written as

$$DG\Delta\vec{q} = \frac{D\vec{v^*}}{\theta\Delta t} \tag{54}$$

In fact, the matrix DG is a discrete Laplacian. Note that factor 2 in Equation (53) is incorporated in matrix G. Due to the use of the vertical boundary-fitted co-ordinate system, this matrix in three dimensions contains 20 non-zero diagonals[§] and is non-symmetric. Almost all of the computational effort to complete a time step goes into solving Equation (54), so it pays to do this efficiently. Here, we adopt the BiCGSTAB method [28]. Since, preconditioning techniques can change the spectrum of the pressure system in a way that is favourable for fast convergence, the BiCGSTAB algorithm is therefore preconditioned. A very successful class of preconditioning methods consists of incomplete LU factorizations. Well-known examples are the ILU preconditioner as described in Reference [31] and the MILU (Modified ILU) preconditioner of Gustafsson [32].

A number of issues are addressed with respect to further optimization of the convergence rate of the employed solver in non-hydrostatic computing.

- It is common to use the reduction of the residual as a stopping criterion, because the BiCGSTAB method requires calculation of the residual. When solving the system $Ax = b$, after m iterations we have an approximate solution x_m and the residual $r_m = b - Ax_m$ is related to the convergence error $e_m = x - x_m$ by $Ae_m = r_m$, so the reduction of the residual results in the reduction of the convergence error. Concerning the preconditioned system, however, it is better to solve $AU^{-1}L^{-1}y = b$ and $x = U^{-1}L^{-1}y$ instead of $U^{-1}L^{-1}Ax = U^{-1}L^{-1}b$, because then the termination criterion does not depend on the preconditioner, i.e. r_m is not influenced by $U^{-1}L^{-1}$. The iteration process stops at each time step if the ratio of the 2-norm of the residual and of the right-hand side is less than a given accuracy: $\|r_m\|_2/\|b\|_2 < \varepsilon$, with ε in the order of 10^{-2}. In this way, the number of iterations depends on the initial estimate x_0. If the iteration process is started with an accurate estimate, the number of iterations is less than using an inaccurate start vector. A good choice for the starting vector appears to be the solution from the previous time step.
- It is well-known that an average of the ILU and MILU preconditioner may improve the rate of convergence [33]. Based on several numerical experiments, an optimum in the convergence rate is found by taking 55% of MILU and 45% of ILU.
- It has been observed that the pressure correction is slowly time varying. This means that there is no need for the system of equations (54) to be preconditioned at every time step. Since preconditioning is relative expensive with respect to amount of work, much of CPU-time can be saved by preconditioning the system every ten to twenty time steps, as suggested by our experiments.
- The matrix DG may contain a number of positive off-diagonal elements. So, there is no guarantee that the preconditioner can be constructed through the incomplete LU decomposition of DG (see Reference [31]). To overcome this difficulty, a matrix P is constructed by which each positive off-diagonal element of DG is lumped into the main

[§] The three-dimensional stencil of DG is composed of four horizontal slices with each five-point star.

diagonal after which its position is filled with zero. As a consequence, the ILU factorization of P exists and can be used as a preconditioner for system (54). A disadvantage, however, is that this so-called lumped matrix needs to be stored in memory. Fortunately, our experiments show that the preconditioner of the lumped matrix is not only robust but it also leads to faster convergence in many cases (up to 30–50%).

- In the case with a single layer ($K = 1$), one observes that Equation (54) reduces to a standard five-point discretization of the Laplacian and hence, the pressure matrix DG becomes a non-symmetric five-diagonal one. This special case can be solved very effective by means of the strongly implicit procedure (SIP) of Stone [34]. This method constructs an incomplete lower–upper factorization that has the same sparsity as the original matrix. Afterward, the resulting system is solved in an iterative manner by forward and backward substitutions. Although, this iterative process generally required more number of iterations per time step than the preconditioned BiCGSTAB solver, the amount of work per iteration is much lower and on balance, the SIP solver is faster than the BiCGSTAB method. Though, convergence is not guaranteed, like the BiCGSTAB method, the SIP solver often has a very regular convergence behaviour and it rarely breaks down.
- It is well-known that data in the cache can be accessed faster than from conventional memory. In this respect, the matrix DG should be stored as a three-dimensional array with structure `dg(1:20,1:kmax,1:mnmax)` and not `dg(1:mnmax,1:kmax,1:20)`, which may be usual in the Fortran programming. The integer `mnmax` represents the total number of wet points in the horizontal plane and `kmax` equals the total number of layers. The former structure will deliver one block for the matrix elements in the cache, whereas the latter will give 20 blocks. Moreover, it is advantageous to correspond the innermost loop with the first index, which varies most rapidly. This is known for vector computers but it applies to cache hit as well.

Since, we are dealing with the Poisson equation for the pressure correction we should expect to need boundary conditions for the unique solution, whereas no physical boundary conditions for the pressure are required. Fortunately, the boundary conditions for the momentum equations restrict in some sense the matrix G at the boundary, and in this way they implicitly define boundary conditions for the pressure correction equation. For example, if the normal velocity is described at a boundary then no momentum equation for this velocity component need to be solved and hence, the gradient matrix G contains no virtual pressure points. This results in a matrix DG that may be interpreted as the normal gradient of pressure given at that boundary. If, on the other hand, the normal stress is prescribed then again G contains no virtual points since, the pressure is a part of that stress. Hence, at the boundary in question, this results in a Laplacian molecule with Dirichlet boundary condition. However, one difficulty arise with respect to the matrix D. Because of the vertical boundary-conforming co-ordinate, this matrix contains some virtual velocity points near the free surface and the bottom. In our implementation, all these virtual velocity unknowns are expressed in unknowns in the interior by means of linear extrapolation. To sum up, the discrete operator DG given by Equation (54) works exclusively on pressure values in grid points in the interior of the domain.

4.2. Hydrostatic step

During the hydrostatic step, Equations (49) and (47) need to be solved. Analogous to the pressure correction method, first an estimate $u_{i,k}^{**}$ for the velocity $u_{i,k}^{*}$ is computed from

Equation (47) with the water level obtained from the previous time step:

$$\frac{u_{i,k}^{**}-u_{i,k}^{n}}{\Delta t}+g\frac{\delta\zeta_i^n}{\delta x}+\frac{\delta q_{i,k}^n}{\delta x}=F_u^n \tag{55}$$

Next, a correction to the water level, defined as $\Delta\zeta_i \equiv \zeta_i^* - \zeta_i^n$, is applied to achieve the velocity $u_{i,k}^*$. The equation for $u_{i,k}^*$ is obtained by substracting Equation (55) from Equation (47) which yields

$$\frac{u_{i,k}^{*}-u_{i,k}^{**}}{\Delta t}+g\theta\frac{\delta\Delta\zeta_i}{\delta x}=0 \tag{56}$$

Multiplication of Equation (56) with $h_{i,k}^*$ and then summing it from the free surface to the bottom gives

$$Q_i^*=\sum_{k=1}^{K}h_{i,k}^*u_{i,k}^{**}-g\theta\Delta tH_i^*\frac{\delta\Delta\zeta_i}{\delta x}\quad\text{with } H_i^*=\sum_{k=1}^{K}h_{i,k}^* \tag{57}$$

A discretized equation for $\Delta\zeta_i$ is constructed by substituting Equation (57) into Equation (49), giving

$$\frac{\Delta\zeta_i}{\Delta t}-g\theta^2\Delta t\frac{\delta}{\delta x}\left(H_i^*\frac{\delta\Delta\zeta_i}{\delta x}\right)=-\theta\frac{\delta}{\delta x}\left(\sum_{k=1}^{K}h_{i,k}^*u_{i,k}^{**}\right)-(1-\theta)\frac{\delta Q_i^n}{\delta x} \tag{58}$$

which forms a tri-diagonal system of equations in the 2DV framework or a penta-diagonal system in the three-dimensional case. However, this system is non-linear, since the layer thickness $h_{i,k}^*$ and therefore the water depth H_i^* depends on the water level ζ_i^*. Hence, linearization is required and the sequence of steps (55), (58) and (56) must be repeated until a converged result is obtained. Thus, the iteration process is going from $s=1$ to $s=S$ and the system (58) is linearized by freezing $h_{i,k}^*$ and H_i^* at iteration level $s-1$. For accuracy reason, $S>1$. The iteration process stops if $\|(u_{i,k}^*)^s-(u_{i,k}^*)^{s-1}\|_\infty<\varepsilon$, where ε is usually of the order 10^{-4}. A theory of convergence is not available. The linearized system of equations for the water level correction can be solved by means of a direct method and the SIP method [34] in the case of two and three dimensions, respectively. It should be noted that many researchers ([3, 9, 10, 12–15, 19, 24]) replace $h_{i,k}^*$ in Equation (49) by $h_{i,k}^n$. Hence, in their case there is no need for an iteration process to obtain the new water level, i.e. $S=1$, but it is probably less accurate.

4.3. *Overall algorithm*

The sequence of the computation for a time step can be summarized as follows. First, the hydrostatic step is carried out, during which the free surface level ζ^{n+1} and the intermediate horizontal velocity u^* are updated.

1. Start the sequence by guessing the unknowns ζ^n, u^n, w^n, q^n, either initially or from the previous time level.
2. Solve the momentum equation (55) to obtain u^{**}.
3. Solve Equation (58) to obtain the correction $\Delta\zeta$ for water level.
4. Correct the water level and horizontal velocity by means of $\zeta^*=\zeta^n+\Delta\zeta$, Equation (56) for u^*.

5. Repeat steps 2, 3 and 4 until convergence.
6. Set $\zeta^{n+1} = \zeta^*$.

Then, during the non-hydrostatic step, both the velocities u^* and w^* as well as the non-hydrostatic pressure are adjusted in order to fulfil the local continuity equation. With this, the final velocity field (u^{n+1}, w^{n+1}) and non-hydrostatic pressure q^{n+1} is achieved.

7. Solve the momentum equation (48) to obtain w^*.
8. Solve the pressure correction equation (54) to obtain the correction Δq.
9. Update the non-hydrostatic pressure and velocities using $q^{n+1} = q^n + \Delta q$, Equation (52) for u^{n+1} and Equation (53) for w^{n+1}.
10. Update the relative vertical velocity $\omega_{k+1/2}$ from Equation (16).

It should be noted that the present model is also able to simulate fully three-dimensional hydrostatic flows in an efficient manner by means of a switch such that only steps 1 to 6 will be carried out followed by the determination of the relative vertical velocity $\omega_{k+1/2}$ via the layer-integrated continuity equation (derived from Equation (22) using Equation (16)):

$$\omega_{k+1/2} = \omega_{k-1/2} - \frac{\partial h_k}{\partial t} - \frac{\partial h_k u_k}{\partial x}, \quad 1 \leqslant k \leqslant K, \quad \omega_{1/2} = 0 \tag{59}$$

Moreover, the non-hydrostatic pressure is initially set to zero. As a consequence, $q^n \equiv 0$ at every time step and the final solution for the horizontal velocity is $u^{n+1} = u^*$.

5. RESULTS

Results have been obtained for standing wave in closed basin, wave over submerged bar and over circular shoal. The computations were carried out on a 2.0 GHz Pentium-4 Intel PC with 2GByte internal memory.

The first test case is included to establish the superior accuracy obtained by the proposed method using the pressure correction technique and the correct implementation of the zero pressure condition at the free surface.

Our main interest concerns the simulation of transformation, refraction and diffraction of non-linear waves over rapidly varying bathymetry in coastal zones. The present method using the Keller-box scheme is validated by applying it to the second and third test cases for which experimental data exist. Performance of the method with respect to reliability and cost is examined. With respect to the time integration employing the θ-method, $\theta = \frac{1}{2}$ is chosen in both these cases. Concerning the range of applicability of the model to values of kH, with k the wave number, indicating the relative importance of linear wave dispersion, results of our numerical analysis, as depicted in Figure 4, suggest that two layers are sufficient to compute linear dispersive waves up to $kH \leqslant 10$ with a relative error of at most 4%. Hence, only two layers are therefore taken in the considered numerical experiments.

5.1. *Standing short wave in closed basin*

The objective of this section is twofold. Firstly, we demonstrate the superior accuracy of the pressure correction technique, as treated in Section 4.1, in comparison with the fractional step approach as presented in, e.g. Reference [10]. Secondly, the consequence of the assumption of hydrostatic pressure in the surface cells is explored. For this purpose and for an appropriate comparison, we consider the classical case in which the non-hydrostatic pressure is located at the cell centre (cf. Figure 3a) and the vertical pressure gradient $\partial q/\partial z$ is approximated with explicit central differences instead of the Keller-box scheme (see, e.g. Reference [9]). As a consequence, Equations (48) and (51) are replaced by, respectively,

$$\frac{w^*_{i,k} - w^n_{i,k}}{\Delta t} + \frac{\delta q^n_{i,k}}{\delta z} = F^n_w \tag{60}$$

and

$$\frac{w^{n+1}_{i,k} - w^*_{i,k}}{\Delta t} + \theta \frac{\delta \Delta q_{i,k}}{\delta z} = 0 \tag{61}$$

For comparison reason, the fractional step approach, as formulated in Reference [10], will be constructed by carrying out the following changes in our model:

- replace $\delta q^n_{i,k}/\delta x$ in Equation (47) by $(1-\theta)\delta q^n_{i,k}/\delta x$ and $\delta q^n_{i,k}/\delta z$ in Equation (60) by $(1-\theta)\delta q^n_{i,k}/\delta z$ (these correspond to Equations (12) and (14) in Reference [10]), and
- replace $\Delta q_{i,k}$ in Equations (52) and (61) by $q^{n+1}_{i,k}$ (these correspond to Equations (20) and (22) in Reference [10]).

Note that the numerical model of Casulli and Stelling [3] is a special case obtained with $\theta = 1.0$. Contrary to the method of Casulli [10], we do not correct the intermediate water level during the second fractional step. Hence, an error is made. However, we shall demonstrate that this error does not deteriorate the temporal accuracy. Finally, we solve the Poison equation given by Equation (54) with $\Delta\vec{q}$ replacing by $\vec{q}^{\,n+1}$. Note that the corresponding system of equations is not equivalent to the set of equations (27)–(28) in Reference [10] since, no special measures with respect to the surface cells were taken in our case.

We consider a standing wave in a closed basin with length of 20 m and depth of 10 m. Initially, the following wave height is taken

$$\zeta = 0.1 \cos\left(\frac{\pi x}{10}\right), \quad 0 \leqslant x \leqslant 20 \tag{62}$$

The basin is divided horizontally into 20 grid cells of each 1 m, while the depth consists of 10 layers. The time step is taken as 0.1 s. See also Reference [1] for further details. We compare the computed time series of the surface elevation at $x = 17.5$ m obtained with the fractional step and the pressure correction approaches. We also plot the exact solution of the wave following from the linear wave theory. Two different temporal discretizations were employed namely, Crank–Nicolson ($\theta = 0.5$) and backward Euler ($\theta = 1.0$). The results are depicted in Figure 5. We have found a significant damping of the wave height in the fractional step framework, even when the Crank–Nicolson scheme is applied. Nevertheless, the computed propagation speed has not been affected by this approach. The use of the pressure correction method yields much more accurate waves, in which the amplitude is hardly changed. It must be stressed that the error due to the neglect of $\zeta^{n+1}_i - \zeta^*_i$ in Equation (50) appears to be much smaller than the splitting error of the fractional step approach.

Another point of discussion is the implementation of the zero pressure boundary condition at the free surface. Our implementation consists of solving the momentum equation for the vertical velocity w (3) at the free surface. This will introduce virtual pressure points near the

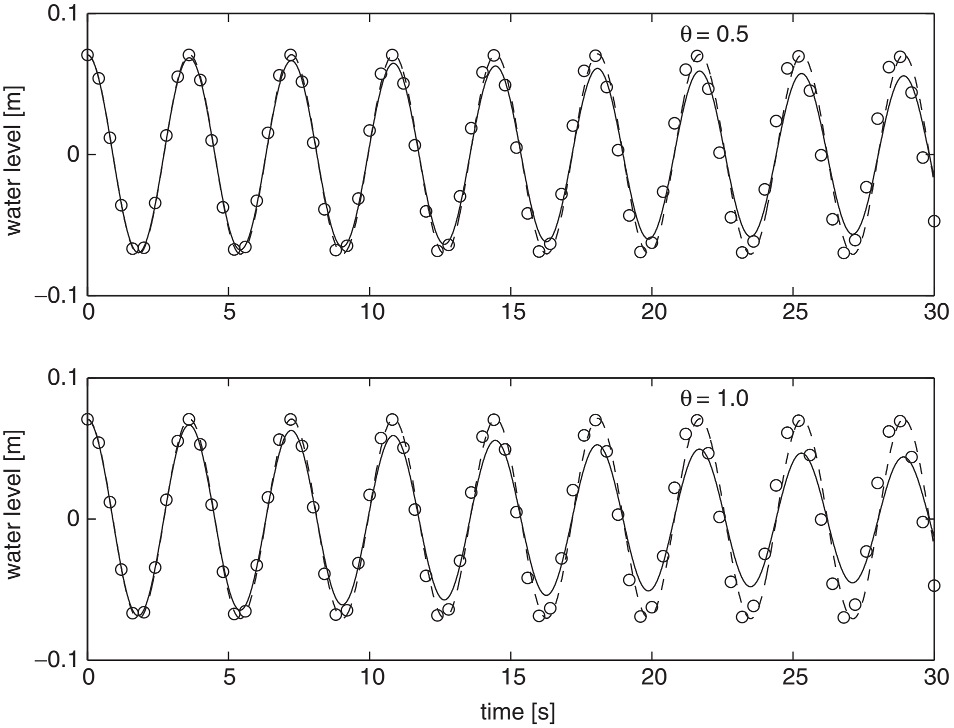

Figure 5. Comparison between computed and exact time series of surface elevation at $x = 17.5$ m for standing wave in closed basin. Fractional step method (solid line), pressure correction technique (dashed line), exact (circles). Time integration is done with both Crank–Nicolson ($\theta = 0.5$) and backward Euler ($\theta = 1$).

free surface that can be readily eliminated by means of a linear extrapolation using condition (8). The usual treatment, however, as presented in References [3, 9–11, 13–15, 24], is to set $q = 0$ inside the surface cells, and accordingly the vertical velocity w is determined by applying the continuity equation (4). To demonstrate the effect of these implementations, we reconsider the simulation of the standing wave in closed basin. Figure 6 shows this effect. Clearly, the computed wave celerity due to the latter implementation is totally wrong, whereas the former one shows almost the same propagation speed of the exact wave (see also Reference [25]).

5.2. Non-linear dispersive waves over a submerged bar

The propagation of non-breaking waves over a submerged bar with relatively steep slopes is considered. The configuration of Ohyama *et al.* [35] has been selected and is depicted in Figure 7. The length of the wave flume in their physical experiments is 65 m. In the deepest part of the flume, the still water depth is 0.5 m and is reduced to 0.15 m in the shallow water region. Both the offshore and onshore slope of the bar is 1:2. Surface elevations are measured using wave gauges at five different locations as indicated in Figure 7. Six conditions have been considered: three different wave periods T (short, intermediate and long) each

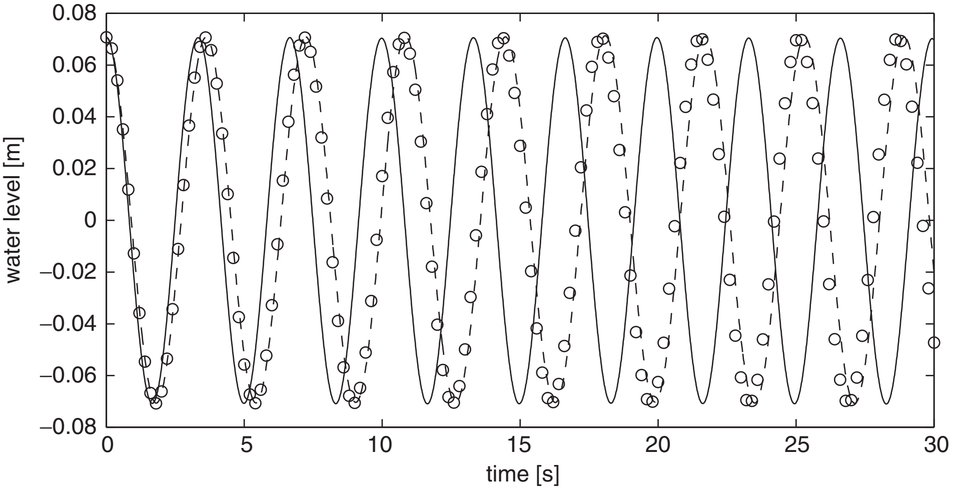

Figure 6. Comparison between computed and exact time series of surface elevation at $x = 17.5\,\mathrm{m}$ for standing wave in closed basin. Result obtained with $q = 0$ inside surface cells (solid line); result obtained with non-hydrostatic pressure in surface cells (dashed line), exact (circles).

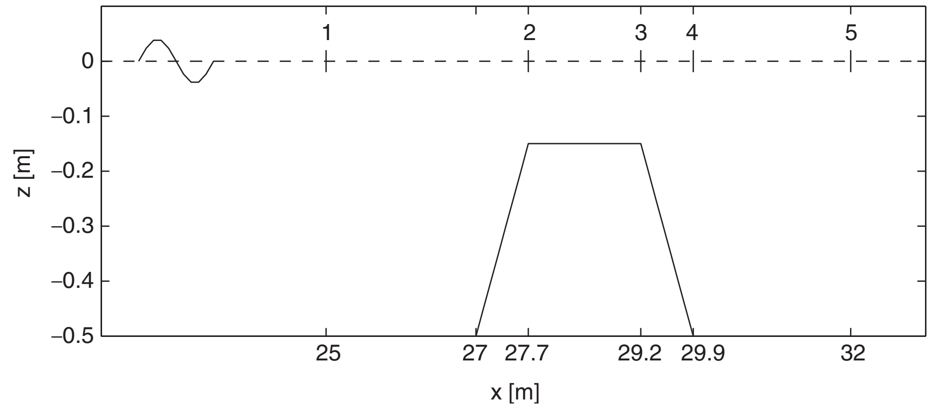

Figure 7. Geometry of submerged bar and location of wave gauges.

with two different wave heights a (small and large). In this way, effects of the various degrees of dispersivity (measured by kH) and non-linearity (measured by a/H) can be studied. Propagation of waves over the bar results in rapidly energy transfer from a primary wave component into its higher harmonics and thus become non-linear. Furthermore, the higher harmonics become more and more dispersive. Behind the bar the non-linearity is weak and as a consequence, the bound waves become free and at the same time highly dispersive. Details can be found in Reference [35]. Two of the six conditions are considered in this paper, namely Case 2 (short, large waves) and Case 4 (intermediate, large waves). They are representatives of the non-linear dispersive waves ocurring in coastal zones. The incident wave height in the

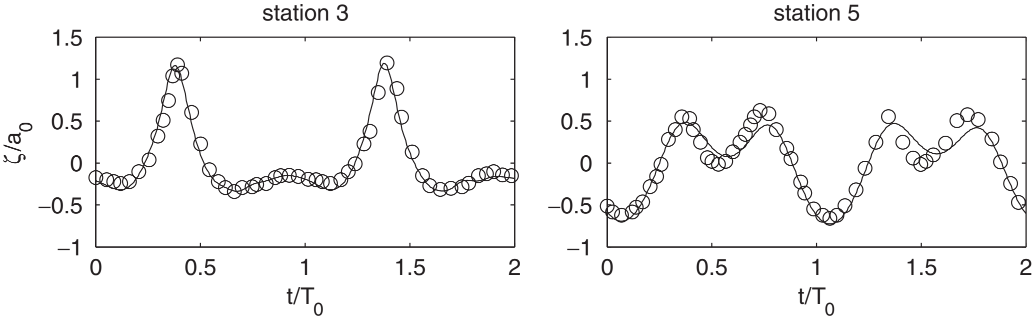

Figure 8. Computed wave profiles at two stations compared to the measured ones of Case 2 for the wave over submerged bar. Present method (solid line), experiment (circles).

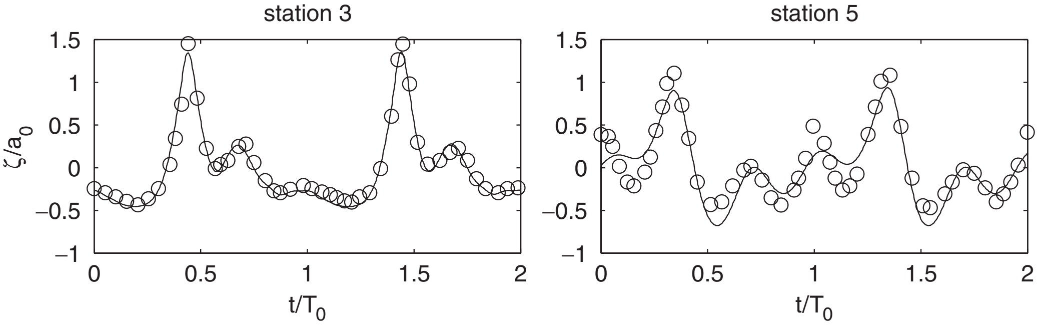

Figure 9. Computed wave profiles at two stations compared to the measured ones of Case 4 for the wave over submerged bar: present method (solid line); experiment (circles).

regarded cases is $a_0 = 0.05\,\mathrm{m}$. The incident wave period for Case 2 is $T_0 = 1.34\,\mathrm{s}$ and for Case 4 $T_0 = 2.01$ s. According to the linear wave theory, k_0H_0 is 1.30 and 0.77 in Cases 2 and 4, respectively ($H_0 = 0.5\,\mathrm{m}$).

The horizontal mesh size is set to $\Delta x = \lambda_0/50$ with λ_0 the incident wave length, and the time step is taken as $\Delta t = T_0/100$. Comparison between the model results and the experimental data is considered in both the shallow water region (Station 3) and deep water region (Station 5). Figure 8 displayed the time series of the surface elevation at both stations for Case 2. The measured profiles are very well predicted by the present non-hydrostatic model. Note that the model slightly underestimates the wave energy behind the bar. Wave profiles at Stations 3 and 5 for Case 4 are plotted in Figure 9. Again, the present method reproduces the experimental results at Station 3 considerable well. Clearly, behind the bar the higher harmonics are underestimated by the model. This discrepancy may be reduced by employing a finer horizontal mesh. Nevertheless, the quality of these results is better than that of the results obtained with an extended Boussinesq-type wave model as presented in Reference [35]. This is due to the steep bottom slopes which violates the inherent mild-slope assumption of the latter model. Earlier non-hydrostatic calculation related to this case has been carried out by Zhou and Stansby [14], where 20 σ-layers have been used. Compared to the present results, it appears that their results are less accurate. This may be due to insufficient mesh resolution in their computations.

To give an impression of the performance of the present method, information on the computing times will be given. The stopping criterion for the solution of the pressure correction is set to $\varepsilon = 0.01$ and for the outer iteration process is $\varepsilon = 5 \times 10^{-4}$. It appears that only 1 iteration per time stap was needed for the solution of the Poisson equation and typically three outer iterations were taken. The total CPU time needed for each time step and grid point was approximately 15 μs of which 35% is consumed by the building (18%) and solving (17%) of the Poisson equation.

5.3. Deformation of waves by a submerged circular shoal

In this section, we compare numerical results from a computation of wave transformation over a submerged circular shoal in a three-dimensional configuration with measured data from a physical experiment. The experimental data are taken from Reference [36]. Earlier computation related to this problem has been carried out by Chen *et al.* [37] using an extended Boussinesq wave model. To authors' knowledge, no results obtained with a non-hydrostatic free-surface flow model for this test case have been published earlier. Based on the configuration of Reference [36], the simulation is considered in a rectangle basin $[(x, y) : 0 \leqslant x \leqslant 22.0, 0 \leqslant y \leqslant 18.2]$ with a flat bottom on which a circular shoal is rested. The experimental layout is depicted in Figure 10 where wave heights along seven transects near the shoal, A to G, were measured. The still water depth is $H_0 = 0.45$ m and the depth on the shoal is given by

$$H = H_0 + 8.73 - \sqrt{82.81 - (x-5)^2 - (y-8.98)^2} \tag{63}$$

The shape of the shoal is represented by

$$(x-5)^2 + (y-8.98)^2 = 6.6049 \tag{64}$$

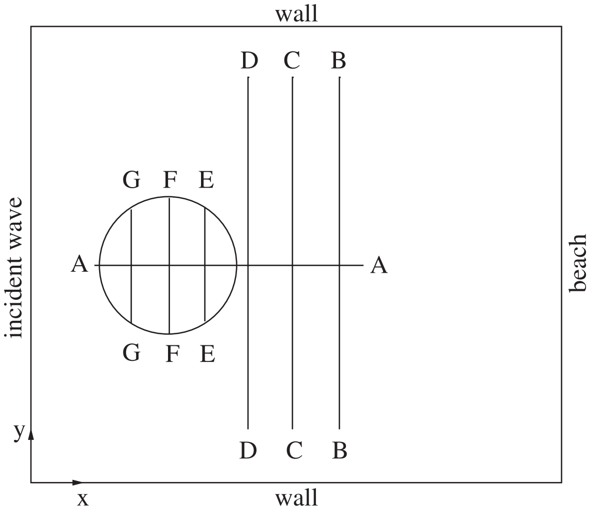

Figure 10. Sketch of basin with circular shoal and transects for collecting wave data.

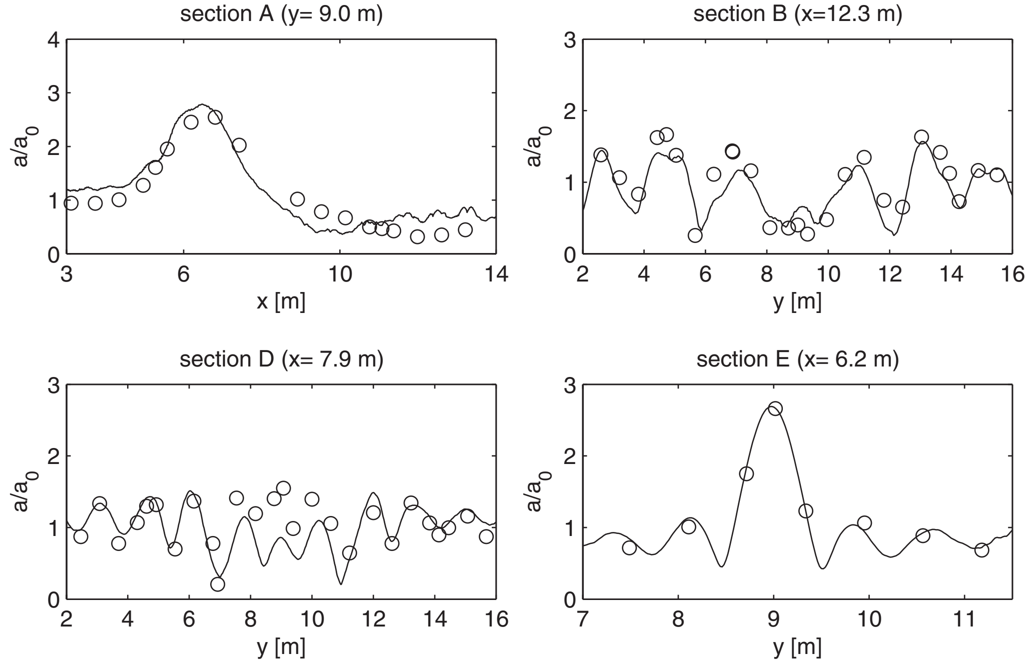

Figure 11. Computed (solid line) and measured (circles) normalized wave heights along sections A, B, D and E for the wave over circular shoal.

Only non-breaking waves are considered in the present computation. Monochromatic waves with wave height of $a_0 = 1.18$ cm and wave period of $T_0 = 1.0$ s are generated at left boundary. The right boundary is of the outflow type where the Sommerfeld's radiation condition (9) is applied. The lower and upper boundaries are insulated and the free-slip conditions are imposed.

The grid size in both directions is set to 0.02 m. The time step is taken as 0.01 s and the simulation period is set to 40 s, so that a steady state is reached. No instabilities were encountered. It turns out that on average 2 iterations per time step for the pressure correction ($\varepsilon = 0.01$) and approximately 50 CPU hours on the 2.0 GHz Pentium machine were required. Profiles of the normalized wave height along four transects, which are the most compelling ones, are given in Figure 11 and compared with the experimental data. The results of the calculation are qualitatively in good agreement with the measurements, as far as they can be represented by the present mesh. The comparison along section A indicates that both shoaling and focussing of waves are very well predicted by the present model. Also, the variation of the waves in cross direction representing the effects of combined refraction and diffraction is predicted fairly well as shown by the comparison of the computed and measured profiles along sections B, D and E. It is believed that the model results can be further improved by refining the grid to capture higher harmonics behind the shoal. Finally, both the present method and the Boussinesq wave model employed by Chen *et al.* [37] produce similar profiles.

6. CONCLUSIONS

A finite volume algorithm for non-hydrostatic, free surface flows involving water waves governed by the incompressible Euler equations on staggered grids has been presented in detail and validated by reference to experimental data. This algorithm made use of a boundary-conforming co-ordinate system in the vertical. This co-ordinate change allows a number of layers having a uniform constant thickness for each layer. Furthermore, discretization in the vertical is carried out by means of the Keller-box scheme that enables to calculate wave propagation with linear dispersion effects accurately using a few number of layers. For accuracy reason, the pressure is split-up into hydrostatic and non-hydrostatic parts. Semi-implicit time stepping is done in combination with projection methods, where correction to the velocity fields for the change in both surface elevation and non-hydrostatic pressure is incorporated. Moreover, space discretization precedes introduction of pressure correction, so that no artificial pressure boundary conditions are required. Accurate implementation of the zero pressure boundary condition at the free surface is treated. No special measures are required to reduce splitting errors. The iterative solution of the unsymmetric Poisson equation for pressure correction is the most time consuming part and therefore, the efficient BiCGSTAB method accelerated with incomplete LU type preconditioners is employed. Further reduction of CPU-time can be realized through, notably, constructing lumped preconditioners, averaging ILU and MILU preconditioners and preconditioning the pressure system every ten to twenty time steps. The present scheme is sufficiently accurate and is also locally and globally mass conservative. Computational results have demonstrated that the proposed method is suitable for accurate and efficient simulation of propagation of non-linear dispersive waves over uneven bottoms. The model can be applied in practical applications that comprise areas with spatial dimensions of the order of 10–100 wave lengths, particularly in the vicinity of the coast. In the near future, the model will be coupled to a spectral wave model that can be applied on a scale of the order of 100–1000 wave lengths.

ACKNOWLEDGEMENTS

The authors would like to thank Arun Chawla for providing us measurement data for the submerged circular shoal case.

REFERENCES

1. 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.
2. Keller HB. A new difference scheme for parabolic problems. In *Numerical Solutions of Partial Differential Equations II*, Hubbard B (ed.). Academic Press: New York, 1971; 327–350.
3. Casulli V, Stelling GS. Numerical simulation of 3D quasi-hydrostatic, free-surface flows. *Journal of Hydraulic Engineering* (ASCE) 1998; **124**:678–686.
4. Mahadevan A, Oliger J, Street R. A nonhydrostatic mesoscale ocean model. Part II: numerical implementation. *Journal of Physical Oceanography* 1996; **26**:1881–1900.
5. Marshall J, Adcroft A, Hill C, Perelman L, Heisey C. A finite-volume, incompressible Navier–Stokes model for studies of the ocean on parallel computers. *Journal of Geophysical Research* 1997; **102**:5753–5766.
6. Daily C, Imberger J. Modelling solitons under the hydrostatic and Boussinesq approximations. *International Journal for Numerical Methods in Fluids* 2003; **43**:231–252.
7. Li B, Fleming CA. Three-dimensional model of Navier–Stokes equations for water waves. *Journal of Waterway, Port, Coastal and Ocean Engineering* 2001; **127**:16–25.

8. Li B, Fleming CA. Three-dimensional hydrodynamic model for free surface flow. *Journal of Hydraulic Research* 2003; **41**:367–377.
9. Stelling GS, Busnelli MM. Numerical simulation of the vertical structure of discontinuous flows. *International Journal for Numerical Methods in Fluids* 2001; **37**:23–43.
10. 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.
11. Casulli V, Zanolli P. Semi-implicit numerical modeling of nonhydrostatic free-surface flows for environmental problems. *Mathematical and Computer Modelling* 2002; **36**:1131–1149.
12. Chen X. A fully hydrodynamic model for three-dimensional, free-surface flows. *International Journal for Numerical Methods in Fluids* 2003; **42**:929–952.
13. Stansby PK, Zhou JG. Shallow-water flow solver with non-hydrostatic pressure: 2D vertical plane problems. *International Journal for Numerical Methods in Fluids* 1998; **28**:514–563.
14. Zhou JG, Stansby PK. An arbitrary Lagrangian–Eulerian σ (ALES) model with non-hydrostatic pressure for shallow water flows. *Computer Methods in Applied Mechanics and Engineering* 1999; **178**:199–214.
15. Koçyigit MB, Falconer RA, Lin B. Three-dimensional numerical modelling of free surface flows with non-hydrostatic pressure. *International Journal for Numerical Methods in Fluids* 2002; **40**:1145–1162.
16. Wesseling P. *Principles of Computational Fluid Dynamics.* Springer: Berlin, 2001.
17. Van Kan JJIM. A second-order accurate pressure correction method for viscous incompressible flow. *SIAM Journal on Scientific and Statistical Computing* 1986; **7**:870–891.
18. Armfield S, Street R. An analysis and comparison of the time accuracy of fractional-step methods for the Navier–Stokes equations on staggered grids. *International Journal for Numerical Methods in Fluids* 2002; **38**:255–282.
19. Lin P, Li CW. A σ-coordinate three-dimensional numerical model for surface wave propagation. *International Journal for Numerical Methods in Fluids* 2002; **38**:1045–1068.
20. 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.
21. Molemaker MJ, Dijkstra HA. Stability of a cold core eddy in the presence of convection: hydrostatic versus nonhydrostatic modeling. *Journal of Physical Oceanography* 2000; **30**:475–494.
22. Johns B. The modelling of the free surface flow of water over topography. *Coastal Engineering* 1991; **15**: 257–278.
23. Li Z, Johns B. A numerical method for the determination of weakly non-hydrostatic non-linear free surface wave propagation. *International Journal for Numerical Methods in Fluids* 2001; **35**:299–317.
24. Namin MM, Lin B, Falconer RA. An implicit numerical algorithm for solving non-hydrostatic free-surface flow problems. *International Journal for Numerical Methods in Fluids* 2001; **35**:341–356.
25. Yuan H, Wu CH. 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.
26. Gresho PM, Sani RL. On pressure boundary conditions for the incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 1987; **7**:1111–1145.
27. Stelling GS, Van Kester JAThM. On the approximation of horizontal gradients in sigma co-ordinates for bathymetry with steep slopes. *International Journal for Numerical Methods in Fluids* 1994; **18**:915–935.
28. Van der Vorst HA. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1992; **13**:631–644.
29. Romate JE. Absorbing boundary conditions for free surface waves. *Journal of Computational Physics* 1992; **99**:135–145.
30. Stelling GS, Wiersma AK, Willemse JBThM. Practical aspects of accurate tidal computations. *Journal of Hydraulic Engineering* (ASCE) 1986; **112**:802–817.
31. Meijerink JA, Van der Vorst HA. An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix. *Mathematics of Computation* 1977; **31**:148–162.
32. Gustafsson I. A class of first order factorization methods. *BIT* 1978; **18**:142–156.
33. Axelsson O, Lindskog G. On the eigenvalue distribution of a class of preconditioning methods. *Numerische Mathematik* 1986; **48**:479–498.
34. Stone HL. Iterative solution of implicit approximations of multidimensional partial differential equations. *SIAM Journal on Numerical Analysis* 1968; **5**:530–558.
35. Ohyama T, Kioka W, Tada A. Applicability of numerical models to nonlinear dispersive waves. *Coastal Engineering* 1995; **24**:297–313.
36. Chawla A. Wave transformation over a submerged shoal. *M.Sc. Thesis*, University of Delaware, 1995.
37. Chen Q, Kirby JT, Dalrymple RA, Kennedy AB, Chawla A. Boussinesq modeling of wave transformation, breaking, and runup, II: 2D. *Journal of Waterway*, *Port*, *Coastal and Ocean Engineering* 2000; **126**:48–56.