INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2009; **61**:382–410
Published online 26 November 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1961

Non-hydrostatic 3D free surface layer-structured finite volume model for short wave propagation

L. Cea[1,*,†], G. Stelling[2] and M. Zijlema[2]

[1] *Environmental and Water Engineering Group (GEAMA), Faculty of Civil Engineering, University of A Coruña, Spain*
[2] *Environmental Fluid Mechanics Section, Faculty of Civil Engineering and Geosciences, Delft University of Technology, The Netherlands*

SUMMARY

In this paper a layer-structured finite volume model for non-hydrostatic 3D environmental free surface flow is presented and applied to several test cases, which involve the computation of gravity waves. The 3D unsteady momentum and mass conservation equations are solved in a collocated grid made of polyhedrons, which are built from a 2D horizontal unstructured mesh, by just adding several horizontal layers. The mesh built in such a way is unstructured in the horizontal plane, but structured in the vertical direction. This procedure simplifies the mesh generation and at the same time it produces a well-oriented mesh for stratified flows, which are common in environmental problems. The model reduces to a 2D depth-averaged shallow water model when one single layer is defined in the mesh. Pressure–velocity coupling is achieved by the Semi-Implicit Method for Pressure-Linked Equations algorithm, using Rhie–Chow interpolation to stabilize the pressure field. An attractive property of the model proposed is the ability to compute the propagation of short waves with a rather coarse vertical discretization. Several test cases are solved in order to show the capabilities and numerical stability of the model, including a rectangular free oscillating basin, a radially symmetric wave, short wave propagation over a 1D bar, solitary wave runup on a vertical wall, and short wave refraction over a 2D shoal. In all the cases the numerical results are compared either with analytical or with experimental data. Copyright © 2008 John Wiley & Sons, Ltd.

Received 25 February 2008; Revised 2 July 2008; Accepted 2 October 2008

KEY WORDS: non-hydrostatic free surface flow; short waves; solitary wave; runup; finite volume method; unstructured mesh

[*] Correspondence to: L. Cea, E.T.S. Ingenieros de Caminos Canales y Puertos, Campus de Elviña s/n, 15071, A Coruña, Spain.
[†] E-mail: lcea@udc.es

Contract/grant sponsor: Publishing Arts Research Council; contract/grant number: 98-1846389

1. INTRODUCTION

Numerical models have become widely used in the study of free surface flows in environmental engineering. A number of models exist for the simulation of free surface wave propagation in coastal engineering applications. Shallow water models are often used for the simulation of long shallow waves but not for short deep waves, where the 3D and non-hydrostatic effects are relevant. Even the 3D shallow water equations (3D-SWE), which are a hydrostatic pressure version of the 3D Reynolds-Averaged Navier–Stokes equations, are not able to compute the propagation of deep waves. This is because deep wave dispersion is mainly caused by a non-hydrostatic pressure distribution.

Depth-averaged models based on the Boussinesq wave equations are an extension of depth-averaged shallow water models. Boussinesq-type models are specially suited for the modelling of wave dispersion, shoaling, refraction, and diffraction in mild-slope shallow and intermediate waters. However, Boussinesq equations assume an irrotational and inviscid flow and therefore, they cannot consider the turbulent diffusion nor the wave-current interaction, which can be relevant in some problems.

Against the above mentioned models, non-hydrostatic models solve the 3D flow equations without any assumption about the pressure distribution. This is the most general and accurate approach to model free surface flows, although not the most commonly used due to its computational requirements, which are quite expensive. This is specially true in environmental problems, where the size of the spatial domain is very large and flow patterns of different length scales appear in the flow. Nonetheless, the increase in computational power is making feasible the use of 3D models in environmental hydraulics, and it will probably become a common practise in the following years. Olsen and Kjellesvig [1] used a 3D non-hydrostatic model in order to compute the flow in a spillway, where the vertical velocity is important, the extension of the spatial domain is small, and the geometry is simple. Wu *et al.* [2] used the same approach to compute the flow and sediment transport in open channels, while Casulli and Zanolli [3] used a non-hydrostatic free surface model to compute the flow in the Venice Lagoon. Concerning wave propagation, non-hydrostatic models have been used by Stelling and Zijlema [4], Zijlema and Stelling [5], Li and Fleming [6], and Lin and Li [7], among others, using in each case different numerical techniques to solve the 3D flow equations.

The main difficulty found in 3D free surface models compared with models over a fixed geometry is the computation of the unsteady free surface. A rather general classification of the available numerical techniques to compute the free surface position distinguishes between mesh methods and meshless methods. Meshless methods use a Lagrangian formulation in order to compute the movement of fluid particles applying Newton's second law. The most popular meshless method nowadays is the smoothed particle hydrodynamics method [8]. It has the advantage of being able to treat complicated free surface deformations, but its computational cost is extremely high.

Mesh methods can be classified in moving grid methods and fixed grid methods. Moving grid methods use a Lagrangian formulation in order to move the grid nodes and boundaries with the fluid [9, 10]. The free boundary is computed with a front-tracking technique. A popular moving grid method for small deformations of the free surface is the arbitrary Lagrangian–Eulerian method, in which the boundary of the grid moves with the fluid and the inner nodes of the mesh move arbitrarily in order to reduce mesh distortion, being the number of nodes constant through the calculation. The advantage of this approach is that the free surface is resolved sharply, without diffusing it. The main disadvantage is the computational cost, since all the nodes of the mesh

DOI: 10.1002/fld

move at each iteration and thus, the geometric properties of the mesh need to be recomputed. Fixed grid methods can use a fully Eulerian formulation (interface capturing) or a combined Eulerian–Lagrangian formulation (interface tracking). These kind of methods introduce a volume fraction function, which defines if an element is completely wet, partially wet, or completely dry. Among the many Eulerian methods that have been proposed, the Volume of Fluid (VOF) method, originally proposed by Hirt and Nichols in 1981 [11] should be mentioned. Combined Eulerian–Lagrangian methods use a front-tracking technique in order to follow the free surface (like Lagrangian methods), though all the computations are done over a fixed grid (like Eulerian methods). In this category fall the height-function methods [12] (which use a single-valued height function $z_s = f(x, y)$ in order to define the free surface), the surface marker methods [12], and the volume marker methods.

In the particular case of environmental free surface flows, the most suited methods are probably height-function methods for cases in which the free surface can be defined by a single-valued function $z_s = f(x, y)$. This can be usually done in environmental problems with few exceptions, as, for instance, when modelling the shape of a breaking wave.

This paper presents a layer-structured finite volume model for non-hydrostatic 3D free surface flow, specially designed for the computation of environmental flows. The 3D numerical mesh is built from an unstructured 2D horizontal mesh with several horizontal layers over the water depth, being the number of layers dependent on the expected complexity of the vertical velocity profile. The mesh built in such a way is unstructured in the horizontal plane, but structured in the vertical direction. The reasons for using such a kind of mesh are two: first, the mesh generation in rivers and coastal regions is simplified, since the geometry can be discretized with an unstructured 2D mesh and second, a vertical-structured mesh is specially suitable for problems where the free surface is mainly horizontal, as well as for stratified flows, which are common features in environmental problems. When just one horizontal layer is defined in the mesh, the model reduces to a 2D depth-averaged shallow water model. Section 2 presents the equations solved by the model. The numerical schemes used are explained in detail in Section 3. The convective flux is discretized with a high-order scheme, using the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) algorithm to achieve pressure–velocity coupling. The equations are solved in a non-staggered grid, using Rhie-Chow interpolation in order to stabilize the pressure field. An attractive property of the model proposed is the ability to compute the propagation of short waves with a rather coarse vertical discretization. This is possible due to a correct treatment of the free surface boundary condition. In Section 4 several test cases are solved in order to show the capabilities of the model, including a rectangular deep free oscillating basin, a radially symmetric wave, short wave propagation over a 1D bar, solitary wave runup on a vertical wall, and short wave propagation over a 2D shoal.

2. A NON-HYDROSTATIC 3D FREE SURFACE FLOW MODEL

2.1. Mass and momentum equations

Non-hydrostatic free surface water flow is governed by the 3D-RANS equations for an incompressible Newtonian fluid of constant density, which can be expressed as

$$\frac{\partial U_j}{\partial x_j} = 0 \tag{1}$$

DOI: 10.1002/fld

$$\frac{\partial U_i}{\partial t}+\frac{\partial U_i U_j}{\partial x_j}=-\frac{1}{\rho}\frac{\partial P_\mathrm{t}}{\partial x_i}+\frac{\partial}{\partial x_j}\left[\frac{T^v_{ij}}{\rho}-\overline{u_i u_j}\right]-\delta_{i3}g \tag{2}$$

$$T^v_{ij}=\mu\left(\frac{\partial U_i}{\partial x_j}+\frac{\partial U_j}{\partial x_i}\right) \tag{3}$$

where U_i is the velocity, P_t is the total pressure, T^v_{ij} is the viscous stress tensor, ρ is the fluid density, g is the gravity acceleration, μ is the fluid dynamic viscosity, and $\rho\overline{u_i u_j}$ are the Reynolds stresses. The total pressure P_t can be decomposed into a hydrostatic contribution P_h and a dynamic contribution P_d as:

$$P_\mathrm{t}=P_\mathrm{h}+P_\mathrm{d} \tag{4}$$

The hydrostatic pressure balances the gravity force, being defined as:

$$\frac{\partial P_\mathrm{h}}{\partial z}=-\rho g, \quad P_\mathrm{h}(z=z_\mathrm{s})=0 \tag{5}$$

Vertical integration of Equation (5) from an arbitrary elevation z to the free surface elevation z_s gives:

$$P_\mathrm{h}=\rho g(z_\mathrm{s}-z) \tag{6}$$

Introducing Equations (4) and (6) into the 3D-RANS equations gives:

$$\frac{\partial U_j}{\partial x_j}=0 \tag{7}$$

$$\frac{\partial U_i}{\partial t}+\frac{\partial U_i U_j}{\partial x_j}=-g\frac{\partial z_\mathrm{s}}{\partial x_i}-\frac{1}{\rho}\frac{\partial P_\mathrm{d}}{\partial x_i}+\frac{\partial}{\partial x_j}\left[\frac{T^v_{ij}}{\rho}-\overline{u_i u_j}\right], \quad i=1,2 \tag{8}$$

$$\frac{\partial U_3}{\partial t}+\frac{\partial U_3 U_j}{\partial x_j}=-\frac{1}{\rho}\frac{\partial P_\mathrm{d}}{\partial x_3}+\frac{\partial}{\partial x_j}\left[\frac{T^v_{3j}}{\rho}-\overline{u_3 u_j}\right] \tag{9}$$

In all the problems presented in the paper, which concern wave propagation, turbulence, and viscous stresses can be neglected, and Equations (7)–(9) reduce to:

$$\frac{\partial U_j}{\partial x_j}=0 \tag{10}$$

$$\frac{\partial U_i}{\partial t}+\frac{\partial U_i U_j}{\partial x_j}=-g\frac{\partial z_\mathrm{s}}{\partial x_i}-\frac{1}{\rho}\frac{\partial P_\mathrm{d}}{\partial x_i}, \quad i=1,2 \tag{11}$$

$$\frac{\partial U_3}{\partial t}+\frac{\partial U_3 U_j}{\partial x_j}=-\frac{1}{\rho}\frac{\partial P_\mathrm{d}}{\partial x_3} \tag{12}$$

DOI: 10.1002/fld

2.2. *Initial and boundary conditions*

In order to solve Equations (10)–(12) in a spatial domain, initial and boundary conditions need to be specified. The initial value of all the variables (velocity, dynamic pressure, and free surface elevation) need to be specified:

$$\begin{aligned} U_i(x,y,z,t=0) &= U_i^{\mathrm{o}}, \quad i=1,3 \\ P_{\mathrm{d}}(x,y,z,t=0) &= P_{\mathrm{d}}^{\mathrm{o}} \\ z_{\mathrm{s}}(x,y,t=0) &= z_{\mathrm{s}}^{\mathrm{o}} \end{aligned} \tag{13}$$

Four kinds of boundary conditions are considered: inlet boundary, outlet boundary, wall boundary and free surface boundary.

2.2.1. Inlet and outlet open boundaries. At inlet boundaries the three velocity components need to be defined, while no condition is needed for the water depth. The first derivative of the dynamic pressure is set to zero:

$$\begin{aligned} &U_i(\Gamma_{\mathrm{in}},t) = U_{c,i}, \quad i=1,3 \\ &\frac{\partial P_{\mathrm{d}}}{\partial x_n}(\Gamma_{\mathrm{in}},t) = 0 \end{aligned} \tag{14}$$

where Γ_{in} is the inlet boundary, $U_{c,i}$ is the value imposed at the inlet boundary for each velocity component, and x_n is the normal direction to the boundary considered.

At outlet boundaries only the water surface elevation needs to be prescribed, and the first derivative of the dynamic pressure is also set to zero:

$$\begin{aligned} &z_{\mathrm{s}}(\Gamma_{\mathrm{out}},t) = z_c \\ &\frac{\partial P_{\mathrm{d}}}{\partial x_n}(\Gamma_{\mathrm{out}},t) = 0 \end{aligned} \tag{15}$$

where Γ_{out} is the outlet boundary and z_c is the imposed value for the water surface elevation at the outlet boundary.

2.2.2. Wall boundaries. Wall boundaries include the lateral walls as well as the bed surface. Both boundaries are impervious and therefore, the velocity component normal to the boundary is set to zero in both cases. As we are not considering viscous or turbulent stresses in the problems presented in this paper, the wall friction is neglected (free slip boundary condition). Regarding pressure, its first derivative is set to zero at wall boundaries.

$$\begin{aligned} &U_n(\Gamma_{\mathrm{wall}},t)=0 \\ &\frac{\partial P_{\mathrm{d}}}{\partial x_n}(\Gamma_{\mathrm{wall}},t)=0 \end{aligned} \tag{16}$$

where Γ_{wall} is the wall boundary and U_n is the velocity component normal to the wall boundary.

2.2.3. Free surface boundary. In addition, to compute the position and evolution in time of the free surface boundary, it is necessary to impose correct boundary conditions on it. The boundary

DOI: 10.1002/fld

condition imposed at the free surface is zero shear stress (the wind stress is not considered) and zero dynamic pressure (atmospheric pressure is taken to be the reference pressure). No condition is imposed over the velocity. It is actually the velocity at the free surface boundary, which is used to compute its position and evolution in time, as it will be detailed in Section 3.

3. A LAYER-STRUCTURED FINITE VOLUME SOLVER

3.1. Spatial discretization

The equations presented in Section 2 are solved with a layer-structured finite volume solver. Spatial discretization is done in a semi-structured cell-centred collocated grid. The 3D numerical mesh is built as a 2D unstructured horizontal mesh with several horizontal layers, which can have different thickness (Figure 1). Velocity and pressure are stored at the nodes of the control volumes (collocated grid). The 2D variables (free surface and bed surface elevation) are stored at the geometric centre of the 2D horizontal mesh cells, using the height function $z_{\mathrm{s}}(x, y, t)$ to define the free surface elevation and $z_{\mathrm{b}}(x, y, t)$ to define the bed elevation. The use of a single-valued height function to define the free surface precludes the simulation of plunging waves and other similar phenomena where several layers of fluid are separated with air. On the other hand, the mesh generation is simplified, and the mesh is well oriented for stratified flows.

The cells of the 2D horizontal mesh can have any shape, although most applications can be efficiently meshed with triangles and quadrilaterals. The 3D control volumes are extruded vertically from the 2D cells forming a prism, which usually has 5 or 6 faces, depending whether the 2D mesh is built from triangles or quadrilaterals (Figure 2). The orientation of the control volume faces is either horizontal (top and bottom faces) or vertical (lateral faces).

The total number of control volumes in the mesh (N_t) is the product of the number of 2D cells which form the horizontal mesh (N_i) times the number of horizontal layers (N_k), so that $N_t = N_i N_k$. Alternatively, the 3D mesh can also be thought of as N_i vertical columns of N_k elements per column. Any element in the mesh is identified with two indices, the first index i defines its position in the horizontal mesh ($i = 1, \ldots, N_h$), while the second index k defines the layer to which the element belongs ($k = 1, \ldots, N_k$).

Not all of the control volumes are always active throughout the computation. Depending on the position of the bed and free surfaces, certain control volumes might be completely out of the fluid domain, either because they are located completely below the bed surface or because they are completely above the free surface. Those free-of-fluid volumes do not need to be considered in the computations. The rest of control volumes can be completely or partially filled with water. The thickness of each active control volume is given by

$$\Delta z_{i,k} = \min(z_{k+1}, z_{\mathrm{s},i}) - \max(z_k, z_{\mathrm{b},i}) \tag{17}$$

where z_{b} and z_{s} define respectively the bed and free surface elevations. Due to the movement of the free surface in unsteady computations, at each time step the active elements are identified and their thickness is computed.

3.2. Discretization of the momentum equations

The 3D momentum equations are solved with an unsteady semi-implicit finite volume solver in a collocated grid, using Rhie–Chow interpolation [13] in order to stabilize both the dynamic as

DOI: 10.1002/fld

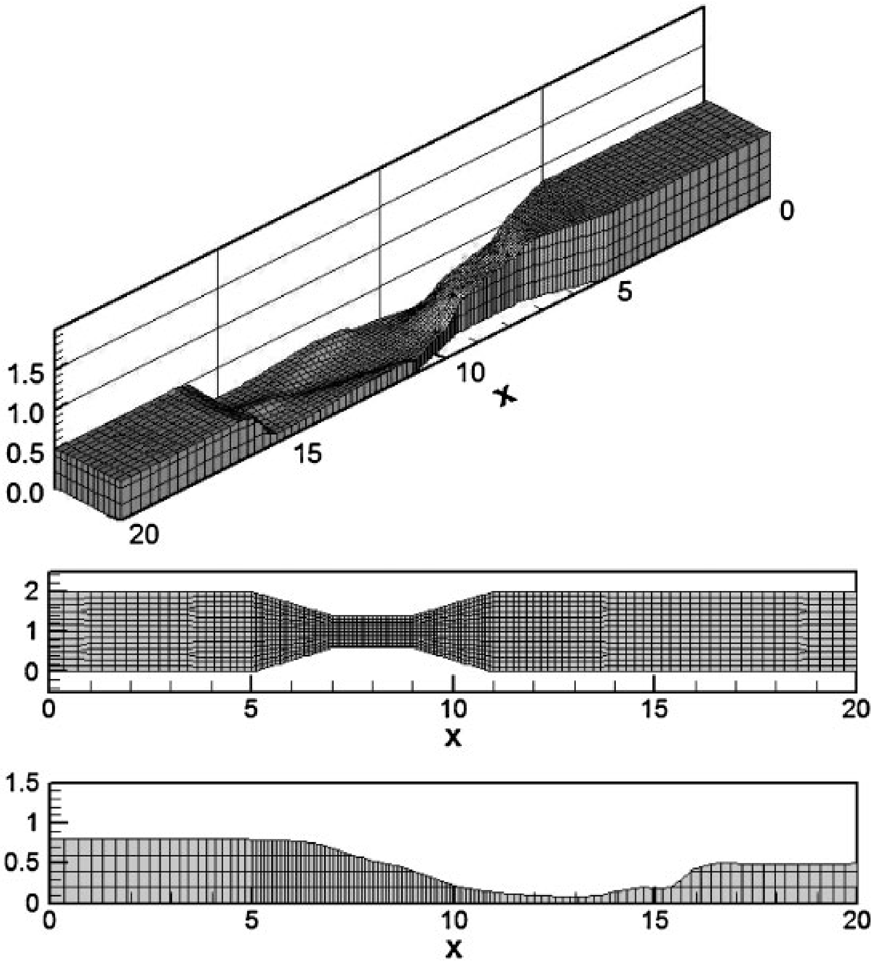

Figure 1. 3D, plan and section view of a typical numerical mesh.

well as the hydrostatic pressure. Integration of the momentum equations (11)–(12) over a control volume C_i gives:

$$\int_{V_i} \frac{\partial \Phi}{\partial t} \mathrm{d}V + F_i^C = F_i^P + F_i^Z \tag{18}$$

where Φ accounts for a given velocity component and F_i^C, F_i^P, F_i^Z represent, respectively, the numerical discretization of the following terms: convection, dynamic pressure and hydrostatic pressure. The subindex i in Equation (18) accounts for integration over the 3D control volume C_i. Convection and pressure terms might be discretized at time t^n, t^{n+1} or any other intermediate time, giving place to explicit or implicit schemes in time. In our case convection terms F_i^C are discretized at time $t^{n+1/2}$, the dynamic pressure source term F_i^P is computed implicitly at time

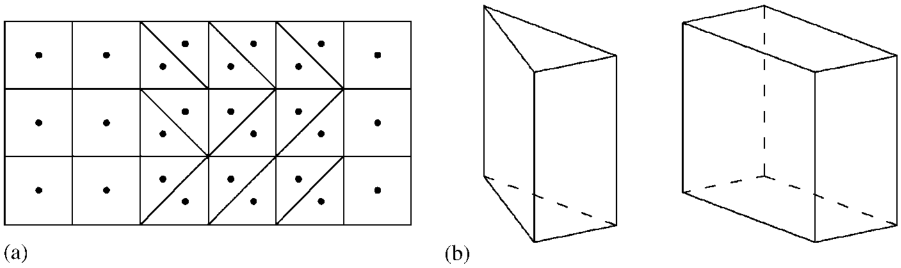

Figure 2. Numerical mesh construction: (a) plan view of an arbitrary numerical mesh and (b) control volumes generated from a triangle and a quadrilateral.

t^{n+1}, while the hydrostatic pressure term F_i^Z is computed explicitly in unsteady computations, but implicitly in steady computations for reasons that will be mentioned below. Discretization of the time derivative in Equation (18) gives

$$\frac{\Phi^{n+1}-\Phi^n}{\Delta t}V_i+F_i^{C,n+1/2}=F_i^{P,n+1}+F_i^{Z,n+\theta} \tag{19}$$

where θ varies between 0 for an explicit discretization of the hydrostatic pressure, and 1 for an implicit discretization of the hydrostatic pressure. The spatial discretization of each term in Equation (19) is specified in the following.

3.3. Discretization of the convective flux

Integration over a control volume of the convective flux in the momentum equation for the velocity component Φ is given by

$$\int_{V_i}\frac{\partial \Phi U_m}{\partial x_m}\,\mathrm{d}V=\int_{A_i}\Phi U_m\tilde{n}_m\,\mathrm{d}A=\int_{A_i}\Phi U_n\,\mathrm{d}A\approx\sum_{j\in K_i}\lambda_{ij}\Phi^*_{ij} \tag{20}$$

where K_i accounts for all the control volumes C_j, which share any face with the control volume C_i, U_n is the velocity component normal to the cell face, $\lambda_{ij}=U_{n,ij}A_{ij}$ is the outlet flux through the control volume face (which is computed by linear interpolation from the velocity value at the nodes N_i and N_j), A_{ij} is the area of the common face to the volumes C_i and C_j, and $\tilde{n}_m$ is the mth component of the unit normal vector to the control volume face. In Equation (20), Φ^*_{ij} accounts for the discretization of the variable Φ at the control volume face, which is computed with an adequate upwind scheme. In all the present applications the second-order Gamma scheme [14] has been used.

The discretization of the convective flux can be cast in standard form as:

$$F_i^{C,n+1/2}=\sum_{j\in K_i}\lambda_{ij}\Phi^*_{ij}=a_i\frac{\Phi_i^n+\Phi_i^{n+1}}{2}+\sum_{j\in K_i}a_{ij}\frac{\Phi_j^n+\Phi_j^{n+1}}{2} \tag{21}$$

The coefficients a_i, a_{ij} might be computed at time t^n or t^{n+1}. In the tests presented in this paper no significant differences were found with respect to the time discretization of the coefficients a_i, a_{ij}. The coefficients a_i and a_{ij} are given by the specific numerical scheme used, in our case the

Gamma scheme. Introducing Equation (21) into Equation (19), the standard form of the discretized momentum equations is obtained as:

$$\frac{V_i}{\Delta t}\Phi_i^{n+1}+a_i\frac{\Phi_i^{n+1}+\Phi_i^n}{2}+\sum_{j\in K_i}a_{ij}\frac{\Phi_j^{n+1}+\Phi_j^n}{2}=\frac{V_i}{\Delta t}\Phi_i^n+F_i^{P,n+1}+F_i^{Z,n+\theta} \tag{22}$$

3.3.1. Gamma scheme. The Gamma scheme was proposed by Jasak *et al.* in [14] as a second-order scheme specially suitable for unstructured meshes. The scheme belongs to the family of normalized variable diagram (NVD) differencing schemes [15, 16]. It is a bounded form of central differencing constructed on a compact computational molecule. This is of great appeal when dealing with unstructured meshes, since the scheme uses only the *close upwind node* to each face in order to compute the convective flux. This is not the case in other high-order schemes, which need information about the *far upwind nodes*, and those are difficult to identify in an arbitrarily unstructured mesh.

The Gamma scheme uses second-order central differencing wherever this scheme satisfies the boundedness criterion in the NVD, and first-order upwind differencing otherwise. In order to avoid instabilities when switching between both schemes, a blending factor γ is introduced and a smooth transition between schemes is established. The scheme works as follows. First, the direction of the normal velocity is checked at every face, and the normalized variable $\tilde{\Phi}$ is computed at the centre of each control volume from the previous solution at time t^n as

$$\tilde{\Phi}_i=1-\frac{\Phi_j-\Phi_i}{2(\mathbf{\nabla\Phi})_i\mathbf{r}_{ij}} \tag{23}$$

where the gradient $(\mathbf{\nabla\Phi})_i$ is computed applying the Gauss theorem to the control volume, and the distance vector $\mathbf{r}_{ij}$ joins the nodes N_i and N_j. In Equation (23) the fluid flows through the control volume face from the node N_i to the node N_j. From the value of the normalized variable $\tilde{\Phi}_i$ at the centre of the control volume, the value of Φ at the cell face is computed as

$$\begin{aligned}\Phi_{ij}^{*}&=[1-\gamma_{ij}(1-f_{ij})]\Phi_i+\gamma_{ij}(1-f_{ij})\Phi_j\\ f_{ij}&=\frac{d_j}{d_i+d_j}=\frac{\Delta_j}{\Delta_i+\Delta_j}\end{aligned} \tag{24}$$

being f_{ij} the linear interpolation coefficient for the control volume face (Figure 3) and γ_{ij} the blending factor. For $\gamma_{ij}=0$ the first-order upwind scheme is recovered, while $\gamma_{ij}=1$ gives the second-order centred scheme. Using Equation (24) into Equation (20), the coefficients a_i and a_{ij} in Equation (21) are obtained as:

$$\begin{aligned}a_{ij}&=\gamma_{ij}(1-f_{ij})\lambda_{ij}\\ a_i&=-\sum_{j\in K_i}a_{ij}+\sum_{j\in K_i}\lambda_{ij}=-\sum_{j\in K_i}a_{ij}\end{aligned} \tag{25}$$

DOI: 10.1002/fld

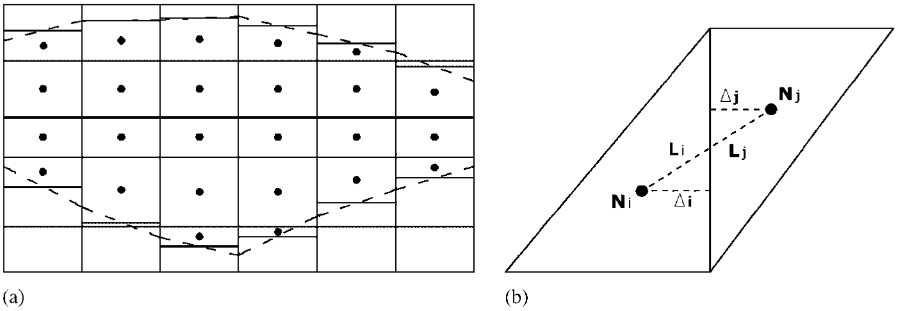

Figure 3. Numerical mesh: (a) vertical section of a numerical mesh and (b) interpolation coefficient for face values.

In order to obtain a stable NVD scheme the blending factor γ_{ij} is computed at each control volume face attending to the following cases:

$$\begin{aligned} \gamma_{ij} &= 0 && \text{if } 1 \leqslant \tilde{\Phi}_i \\ \gamma_{ij} &= 1 && \text{if } \beta_m \leqslant \tilde{\Phi}_i < 1 \\ \gamma_{ij} &= \tilde{\Phi}_i/\beta_m && \text{if } 0 < \tilde{\Phi}_i < \beta_m \\ \gamma_{ij} &= 0 && \text{if } \tilde{\Phi}_i \leqslant 0 \end{aligned} \tag{26}$$

The coefficient β_m controls the amount of blending between the first-order upwind scheme and the second-order central differencing scheme. Small values of β_m introduce less blending and therefore produce more accurate but also more unstable solutions, while large values of β_m give a more stable but less accurate scheme. In [14] a value of β_m between 0.1–0.5 is recommended.

3.4. Dynamic pressure

The dynamic pressure source term in the mth-momentum equation is discretized using a centred scheme as:

$$F_i^{P,n+1} = -\frac{1}{\rho}\int_{V_i} \frac{\partial P_{\mathrm{d}}^{n+1}}{\partial x_m}\,\mathrm{d}V = -\frac{1}{\rho}\int_{A_i} P_{\mathrm{d}}^{n+1}\tilde{n}_m\,\mathrm{d}A = -\frac{1}{\rho}\sum_{j\in K_i} P_{\mathrm{d},ij}^{n+1}\tilde{n}_{m,ij}A_{ij}, \quad m=1,3 \tag{27}$$

The value of the dynamic pressure at the control volume face ($P_{\mathrm{d},ij}^{n+1}$) is computed by linear interpolation from the node values as

$$P_{\mathrm{d},ij}^{n+1} = f_{ij}P_{\mathrm{d},i}^{n+1} + (1-f_{ij})P_{\mathrm{d},j}^{n+1} \tag{28}$$

being f_{ij} the interpolation coefficient defined in Equation (24).

DOI: 10.1002/fld

3.5. Free surface gradient

The free surface gradient source term accounts for the hydrostatic pressure force in the x- and y-directions. The centred discretization of the free surface gradient used in the solver is given by:

$$F_i^{Z,n+\theta} = -g \int_{V_i} \frac{\partial z_{\mathrm{s}}^{n+\theta}}{\partial x_m} \, \mathrm{d}V = -g \int_{A_i} z_{\mathrm{s}}^{n+\theta} \tilde{n}_m \, \mathrm{d}A = -g \sum_{j \in K_i} z_{\mathrm{s},ij}^{n+\theta} \tilde{n}_{m,ij} A_{ij}, \quad m = 1, 2 \tag{29}$$

In Equation (29) the free surface elevation z_{s} might be computed at time t^n (explicitly), at time t^{n+1} (implicitly), or using a combination of both values as:

$$z_{\mathrm{s}}^{n+\theta} = z_{\mathrm{s}}^{n+1} \theta + z_{\mathrm{s}}^{n} (1-\theta), \quad \theta \in [0, 1] \tag{30}$$

It is convenient to use an explicit discretization of the free surface gradient ($\theta = 0$) for unsteady problems, while a semi-implicit discretization, which is numerically more stable, is preferred for steady problems. The reason for doing so is that an implicit discretization introduces much damping in unsteady computations.

3.6. Rhie–Chow interpolation

As a collocated grid is used for the numerical discretization, it is necessary to introduce some kind of stabilization technique in order to avoid a checkerboard pressure field. In the present model, Rhie–Chow interpolation [13] has been used to stabilize both the dynamic as well as the hydrostatic pressure. In order to compute the mass residual for each control volume, the face velocity component in the x-direction is computed from the following expression:

$$\Phi_{ij} = \Phi_{ij}^0 - \frac{1}{\rho} \left. \frac{\partial P_{\mathrm{d}}}{\partial x} \right|_{ij} \frac{V_{ij}}{a_f} - g \left. \frac{\partial z_{\mathrm{s}}}{\partial x} \right|_{ij} \frac{V_{ij}}{a_f}, \quad \Phi_{ij}^0 = f_{ij} \Phi_i^0 + (1 - f_{ij}) \Phi_j^0 \tag{31}$$

with $1/a_f = f_{ij}/a_i + (1 - f_{ij})/a_j$. In Equation (31) the velocities Φ_i^0 are computed from the velocity at the nodes, after subtracting the dynamic pressure and free surface gradient source terms from them, which considering the steady state of Equation (22) gives:

$$\Phi_i^0 = \Phi_i + \frac{1}{\rho} \left. \frac{\partial P_{\mathrm{d}}}{\partial x} \right|_i \frac{V_i}{a_i} + g \left. \frac{\partial z_{\mathrm{s}}}{\partial x} \right|_i \frac{V_i}{a_i} = \Phi_i - \frac{F_i^P + F_i^Z}{a_i} \tag{32}$$

As Rhie–Chow interpolation is just a stabilization technique, in order to simplify the formulation the non-orthogonal contribution can be neglected without any major problem in the discretization of the pressure and free surface gradients at the control volume face in Equation (31). With this approximation the second and third addends on the right-hand side of Equation (31) are computed as

$$\frac{1}{\rho} \left. \frac{\partial P_{\mathrm{d}}}{\partial x} \right|_{ij} \frac{V_{ij}}{a_f} \approx \frac{1}{\rho} \frac{V_{ij}}{a_f} \frac{P_{\mathrm{d},j} - P_{\mathrm{d},i}}{d_{\perp,ij}} \tilde{n}_{x,ij} = \frac{1}{\rho} \frac{1}{a_f} (P_{\mathrm{d},j} - P_{\mathrm{d},i}) \tilde{n}_{x,ij} A_{ij} \tag{33}$$

$$\frac{1}{\rho} \left. \frac{\partial z_{\mathrm{s}}}{\partial x} \right|_{ij} \frac{V_{ij}}{a_f} \approx \frac{1}{\rho} \frac{V_{ij}}{a_f} \frac{z_{\mathrm{s},j} - z_{\mathrm{s},i}}{d_{\perp,ij}} \tilde{n}_{x,ij} = \frac{1}{\rho} \frac{1}{a_f} (z_{\mathrm{s},j} - z_{\mathrm{s},i}) \tilde{n}_{x,ij} A_{ij} \tag{34}$$

where $1/a_f$ is computed as $1/a_f = f_{ij}/a_i + (1 - f_{ij})/a_j$. Equations (31)–(34) are used in order to compute the velocity component Φ_{ij} at each cell face. Analog expressions are used for the velocity

DOI: 10.1002/fld

components in the y- and z-directions, just shifting $\tilde{n}_x$ for $\tilde{n}_y$ or $\tilde{n}_z$ depending on the velocity component considered.

3.7. SIMPLE algorithm

Mass continuity is achieved by pressure–velocity coupling using the SIMPLE algorithm [17]. At each time step the procedure is as follows. First, the three momentum equations are solved using the dynamic pressure field from the previous time step P_{d}^n. The solution of that system of equations gives a first guess for the velocity field at time t^{n+1}, which does not satisfy the continuity equation. The continuity residual at each control volume is computed, and a pressure correction equation is solved in order to obtain a new pressure field and a new divergence free velocity field. Then, all the coefficients a_i, a_{ij}, as well as the source terms F_i^Z, F_i^P are recomputed, the three momentum equations are solved again with the new pressure field, and the process is repeated until convergence, i.e. until the velocity field obtained from the momentum equations is divergence free. Notice that if an explicit discretization of the free surface gradient is used, then the free surface elevation source term F_i^Z is not updated within the SIMPLE iterations. In the former procedure the strongest implicit discretization is that of the convective flux. The implicit discretization of the dynamic pressure and free surface gradients depends on the number of iterations used in the SIMPLE algorithm, i.e. in the SIMPLE convergence criterion.

At each time step, the continuity residual for the control volume C_i is computed as

$$R_i = \sum_{j\in K_i} U_{n,ij}^{\mathrm{RC}} A_{ij} = \sum_{j\in K_i} \lambda_{ij}^{\mathrm{RC}} \tag{35}$$

where $U_{n,ij}^{\mathrm{RC}}$ is the normal outlet velocity to the cell face computed from Rhie–Chow interpolation. As the residual is not zero, a velocity correction u' is needed. In order to achieve mass conservation, the velocity correction must fulfil the following equation:

$$R_i = -\sum_{j\in K_i} u'_{n,ij} A_{ij} \tag{36}$$

The velocity correction $u'_{n,ij}$ can be related to the pressure gradient using a simplification of Equation (22), applied to the control volume face:

$$u'_{n,ij} = -\frac{1}{\rho}\left.\frac{\partial P'_{\mathrm{d}}}{\partial n}\right|_{ij} \frac{V_{ij}}{a_f} \tag{37}$$

where $1/a_f = f_{ij}/a_i + (1-f_{ij})/a_{ij}$ is computed from the coefficients used in the momentum equations. Combining Equations (36) and (37), and neglecting the non-orthogonal pressure gradient at the cell face, gives the following equation for the pressure correction:

$$a_i^* P'_{\mathrm{d},i} + \sum_{j\in K_i} a_{ij}^* P'_{\mathrm{d},j} = R_i \tag{38}$$

with:

$$a_{ij}^* = \frac{1}{\rho}\frac{1}{a_f}\frac{V_{ij}}{d_{\perp,ij}} A_{ij} = \frac{1}{\rho}\frac{A_{ij}^2}{a_f}, \quad a_i^* = -\sum_{j\in K_i} a_{ij}^* \tag{39}$$

Equation (38) is solved in order to obtain the pressure correction $P'_{\mathrm{d},i}$ at each control volume in the mesh. The new pressure field is used in the momentum equation in order to obtain a new velocity field, and an iterative procedure is followed until convergence. Convergence is achieved when the following criterion is fulfilled:

$$\frac{\sum_i \sum_{j\in K_i} \lambda_{ij}^{\mathrm{RC}}}{\sum_i \sum_{j\in K_i} |\lambda_{ij}^{\mathrm{RC}}|} < \varepsilon \tag{40}$$

where ε is a small number ($\varepsilon = 10^{-3}$–10^{-6}). The iterative process is stopped either when the convergence criterion is fulfilled or when a maximum number of iterations is achieved (normally between 5 and 10 iterations are allowed for unsteady problems).

When solving the pressure equation, at the free surface nodes the pressure is computed using a linear interpolation between the free surface and the inner nodes as

$$P_{kwse} = P_{kwse-1} \frac{z_{kwse} - z_{\mathrm{s}}}{z_{kwse-1} - z_{\mathrm{s}}} \tag{41}$$

where the subindex $kwse$ refers to the nodes in the free surface layer (free surface nodes) and z_{s} is the free surface elevation. In Equation (41) it has been considered that the free surface pressure is zero (the atmospheric pressure is the reference pressure). As it will be shown in the results, using Equation (41) gives a much more accurate prediction of the wave celerity than imposing the zero pressure directly at the free surface nodes ($P_{kwse} = 0$). In fact the latter is equivalent to imposing an additional zero vertical gradient boundary condition for the dynamic pressure, which is not physically correct and would require a very fine vertical discretization near the free surface in order to compute the correct wave celerity.

3.8. Free surface computation

In order to update the water surface elevation at each time step, the control volumes in the free surface layer (free surface volumes) are considered. The evolution of the free surface in the model is computed by a mass continuity equation, using a height-function method. This is an adequate and efficient approach for problems with a single-valued free surface function $z_{\mathrm{s}} = f(x, y)$, which is often the case in environmental flows. When the free surface of the fluid is not a single-valued function this approach cannot be used. The free surface equation reads:

$$\frac{\partial z_{\mathrm{s}}}{\partial t} = W_s - \frac{\partial z_{\mathrm{s}}}{\partial x} U_s - \frac{\partial z_{\mathrm{s}}}{\partial y} V_s \tag{42}$$

where U_s, V_s, W_s are the three components of the free surface velocity. Instead of using Equation (42) directly, a mass conservation equation averaged in the vertical direction is used. Integration of the 3D mass conservation equation from an arbitrary elevation z to the free surface elevation z_{s} gives:

$$\begin{aligned} 0 &= \int_z^{z_{\mathrm{s}}} \left(\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z} \right) \mathrm{d}z \\ &= \frac{\partial}{\partial x} \int_z^{z_{\mathrm{s}}} U \,\mathrm{d}z - U_s \frac{\partial z_{\mathrm{s}}}{\partial x} + \frac{\partial}{\partial y} \int_z^{z_{\mathrm{s}}} V \,\mathrm{d}z - V_s \frac{\partial z_{\mathrm{s}}}{\partial y} + W_s - W_z \end{aligned} \tag{43}$$

Combining Equations (42) and (43) gives the following equation for the free surface evolution:

$$\frac{\partial z_{\mathrm{s}}}{\partial t} = W_z - \frac{\partial}{\partial x}\int_z^{z_{\mathrm{s}}} U \,\mathrm{d}z - \frac{\partial}{\partial y}\int_z^{z_{\mathrm{s}}} V \,\mathrm{d}z \tag{44}$$

Equation (44) is solved in a fluid layer near the free surface at each time step. Actually, the integration range is only extended to the free surface volumes, and so the right-hand side of Equation (44) is nothing but the mass residual in the free surface volumes. Hence, the free surface is moved upward or downward depending on the sign of the mass residual at the free surface nodes, using the following equation:

$$z_{\mathrm{s}}^{n+1} - z_{\mathrm{s}}^{n} = -\frac{\Delta t}{A_{h,i}} \sum_j U_{n,ij}^{n+1} A_{ij} \tag{45}$$

where the summation index extends to all the control volume faces except to the top face (the face that defines the free surface) and $A_{h,i}$ is the area of the top face horizontal projection.

3.9. Solver for the system of equations

During the iterative procedure described in this section, four systems of equations need to be solved: three for the momentum equations (Equations (22)) and one for the pressure correction equation (Equation (38)). The number of equations of each system is equal to the number of active volumes in the mesh. Each system of equations is solved iteratively by layers. For each layer it is necessary to solve a sparse linear system, which can be efficiently done by a preconditioned GMRES [18] method. An iterative procedure can be followed, sweeping the horizontal layers several times in order to increase the implicit discretization of the solver, but this is not necessary since the SIMPLE algorithm is itself an iterative method.

4. TEST CASES

Several test cases have been computed in order to verify the capabilities of the model to handle short wave transformation. In all the cases the numerical results are compared either with analytical solutions or with experimental results.

The first test case is a standing wave in a closed and deep rectangular basin. This test case is commonly used to verify the accuracy of wave propagation models when computing linear dispersive waves [4, 5, 19, 20]. The influence on the solution of some numerical issues, as the number of horizontal layers and the hydrostatic pressure semi-implicit time discretization, is analysed. The numerical results are compared with the inviscid analytical solution, with special attention to the wave period prediction and to the numerical damping of the method. The second test case consists in a radially symmetric wave. The results computed with both a structured and an unstructured mesh are compared. In the third test case the experimental data provided for the Benchmark problem number 3 in the Second International Workshop on Long-Wave Runup Models are compared with the numerical results for the shoaling and runup in a vertical wall of solitary waves. The ability of the model to compute dispersive wave transformation over a submerged bar is investigated in the fourth test case, using the experimental results of Beji and Battjes [21] for comparison with the model predictions. Finally, in the fifth test case the propagation of a short wave over a 2D shoal is computed. This problem involves refraction, diffraction, and shoaling

DOI: 10.1002/fld

of dispersive waves. The numerical results are compared with the experimental data of Berkhoff *et al*. [22].

All the test cases have been modelled with a three horizontal layer mesh, using the Gamma scheme and a second-order Crank–Nicholson time discretization. It was found that with this number of layers a good compromise between accuracy and computational cost is met in short wave computations with the present model. One single horizontal layer (depth-averaged SWE) produces completely erroneous results when dealing with short waves. Using two horizontal layers improves considerably the results, but still some errors are introduced in highly dispersive waves. A mesh with three horizontal layers was able to produce reasonably good results in all the test cases studied in this paper. Increasing the number of layers did not improve significantly the results, while increasing the computational cost. Nonetheless, it should be noticed that the thickness of the horizontal layers has some influence on the results. In our computations we have observed that good results are often obtained using a bottom layer with a thickness of approximately half the water depth, a middle layer with a thickness of approximately 30% of the water depth, and a top layer thickness of 20% the water depth. Of course this guideline does not intend to be universal and may vary for other test cases.

4.1. Standing short wave in a closed rectangular basin

This test case is commonly used in order to verify the accuracy of numerical methods in the computation of the wave celerity. It can also be used to evaluate the numerical damping introduced by the discretization scheme. We will consider the same initial conditions as in [4, 19], which produce a highly dispersive wave. The rectangular basin is 20 m long and 10 m deep, and it is initially excited by a sinusoidal perturbation with a wave length of $L = 20$ m and a wave height of $H = 0.2$ m, and the initial conditions are given by:

$$z_s = 10 + 0.1 * \cos\left(\frac{\pi x}{10}\right), \quad u = v = w = 0 \tag{46}$$

In this case linear wave theory applies since the ratios $H/L = 0.01$ and $H/d = 0.02$ are both much lower than 1, and the oscillation period T is given by the linear dispersion relation as:

$$c^2 = \frac{gL}{2\pi}\tanh\frac{2\pi d}{L} = 5.57\,\mathrm{m/s}, \quad T = \frac{L}{c} = 3.59\,\mathrm{s} \tag{47}$$

For the numerical computations, the time step is set to $\Delta t = 0.1\,\mathrm{s} = T/35.9$ and the finite volume size to $\Delta x = 1\,\mathrm{m} = L/20$. The computation is run for 36 s, which approximately comprises 10 wave periods. Figure 4 shows the numerical results obtained using a mesh with three horizontal layers. The thickness of each layer from the bed to the top is: 5, 2.5, and 2.5 m. It is clearly shown in Figure 4 the importance of the correct implementation of the free surface boundary condition for the dynamic pressure. If the zero pressure boundary condition is imposed at the free surface using Equation (41), the oscillation frequency is very accurately resolved. On the other hand, if the zero pressure boundary condition is imposed at the free surface nodes, then the oscillation period is underestimated by the numerical model.

The numerical results obtained with a two layer mesh are shown in Figure 5. The thickness of each layer in this case is 5 m. With just two layers the model underpredicts the wave period. It is also shown in Figure 5 the importance of the hydrostatic pressure time discretization. An implicit discretization of the free surface gradient introduces too much damping in the solution, while using an explicit discretization almost does not damp the numerical solution.

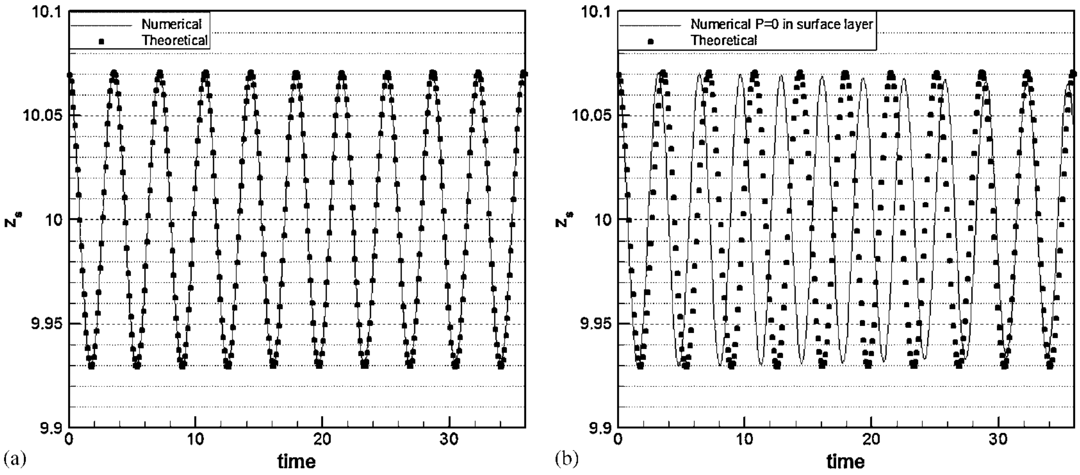

Figure 4. Rectangular closed basin computed with three horizontal layers. Numerical results (solid line) and analytical solution (circles): (a) $P=0$ imposed at the free surface interface and (b) $P=0$ imposed at the free surface nodes.

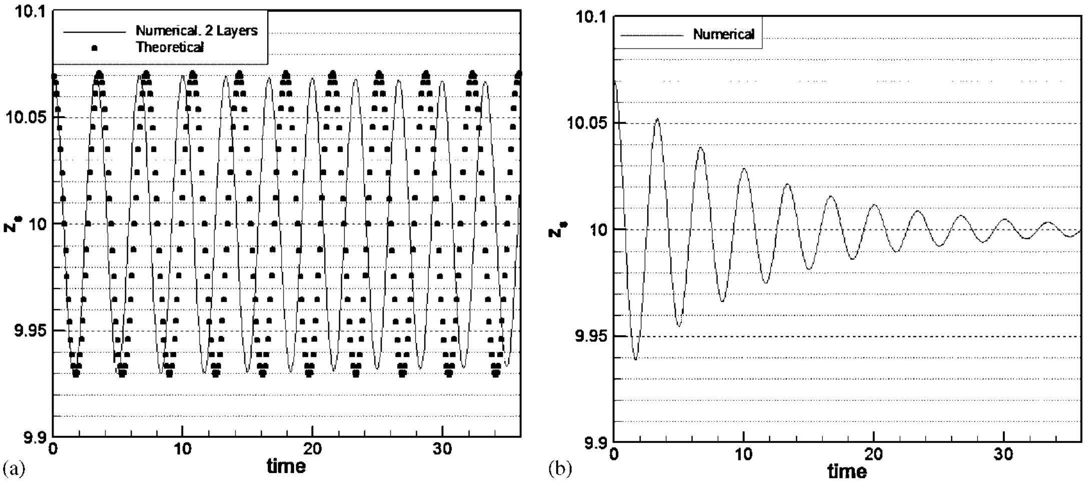

Figure 5. Rectangular closed basin computed with two horizontal layers. Numerical results (solid line) and analytical solution (circles): (a) explicit discretization of the hydrostatic pressure and (b) implicit discretization of the hydrostatic pressure.

Although this problem is just two dimensional in the vertical direction, it has also been computed with a fully 3D mesh, in order to check the unstructured discretization method presented in Section 3. For this purpose, the unstructured horizontal mesh shown in Figure 6 has been used. In this case the nodes (which are placed at the geometric centre of the triangles in Figure 6) are not aligned and the distance between nodes is not uniform. The number of nodes in the longitudinal direction is 40, which gives an average distance between nodes of $\Delta x = 0.5\,\mathrm{m}$. The results computed

DOI: 10.1002/fld

with the unstructured mesh are almost identical to those obtained with the uniform mesh. The amount of damping is virtually identical. There is a small difference in the computed oscillation periods (Table I), but this is not significant considering that a rather coarse discretization is being used.

4.2. *Radially symmetric wave*

A radially symmetric small amplitude inviscid wave was modelled in order to check the ability of the model to compute two-dimensional wave propagation in an unstructured grid. This test case was also modelled in [23, 24]. The initial conditions are given by zero velocity and the following free surface profile:

$$z_s = 0.7 + 0.01\exp(-(x^2+y^2)/10) \tag{48}$$

Two different meshes, a structured mesh built from quadrilaterals as well as an unstructured mesh built from triangles, have been used in order to verify that the same results are obtained in both cases (Figure 7). The structured mesh has 64×64 cells with $\Delta x = \Delta y = 0.5\,\text{m}$. A similar element size was used for the triangular elements of the unstructured mesh. Three horizontal layers have been used in the computations, although in this case almost the same results are obtained with one single layer, since the ratio between the water depth and the wave length d/L is small. The numerical results (Figure 7) agree very well with the theoretical predictions of the wave crest given in [24]. No significant differences appear between the results obtained with both meshes.

4.3. *Solitary wave runup*

This test case is the Benchmark problem number 3 proposed in the Second International Workshop on Long-Wave Runup Models, which was held in Washington in 1995. Solitary wave experiments were conducted in the U.S. Army Engineer Waterways Experiment Station [25], in a 23.2 m long and 0.45 m wide wave flume. The first part of the bed flume (15 m) is horizontal, being followed by three planes with different slopes: a first plane with a slope of 1:53 over 4.36 m, a second plane with a slope of 1:150 over 2.93 m, and a third plane with a slope of 1:13 over 0.9 m. At the end of the flume there is a vertical wall, which produces a complete reflection of the incident wave. The still water depth is 0.218 m in the horizontal part of the flume.

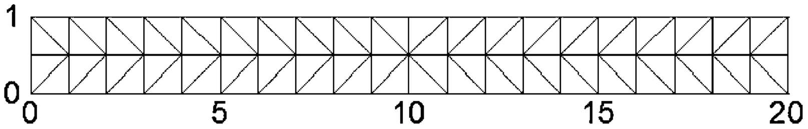

Figure 6. Rectangular closed basin. Horizontal view of the unstructured mesh.

Table I. Rectangular closed basin. Free oscillation periods obtained using different number of layers for the vertical discretization. The analytical solution is $T = 3.59\,\text{s}$.

	1 layer (s)	2 layer (s)	3 layer (s)	4 layer (s)	10 layer (s)
Structured mesh	2.05	3.33	3.58	3.59	3.61
Unstructured mesh	2.03	3.31	3.56	3.57	3.58

DOI: 10.1002/fld

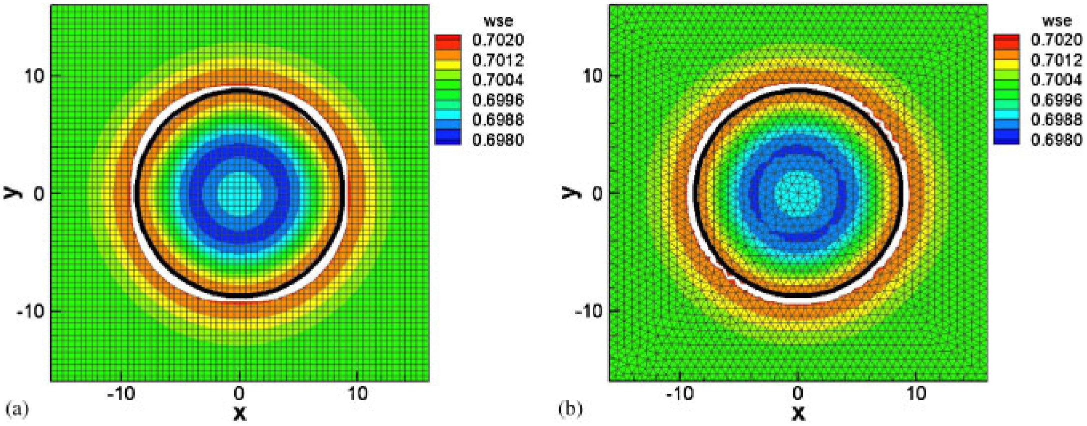

Figure 7. Radially symmetric wave. Numerical (white circle) and theoretical (black circle) predictions of the wave crest: (a) structured mesh and (b) unstructured mesh.

Table II. Solitary wave runup on a vertical wall. Position of the wave gauges.

Gage number	S4	S5	S7	S9
Gage position (m)	12.64 – 14.06	15.04	19.40	22.33

The numerical results are compared with experimental data provided in the Second International Workshop on Long-Wave Runup Models [25]. Three solitary waves were generated in the laboratory experiments using a piston wavemaker. Only the first two waves (Case A and Case B) have been modelled here, since the third wave breaks before reaching the vertical wall. The wave runup at the vertical wall and the time evolution of the free surface at several wave stations were registered in the experiments (Table II). The accuracy of the water depth gauges used in the experiments was around 1 mm, while the runup at the vertical wall was measured by visual observation.

The time step used for the computations is $\Delta t = 0.01\,\mathrm{s}$ and the spatial discretization size $\Delta x = 0.01\,\mathrm{m}$. The vertical discretization is done with three horizontal layers (Figure 8). The thickness of each layer is 12 cm (bed layer), 6 cm (middle layer), and 4 cm (surface layer). The spatial domain covers all the wave flume (23.2 m). The number of finite volumes in the numerical mesh is 2320 per layer (6960 control volumes in the three layers). Further refinement of the spatial and time discretization did not produce significant improvement in the results.

At the inlet boundary the normal pressure and velocity gradients are set to zero, and the free surface level is specified from the following expression, which was proposed for this problem in [26]:

$$z_{\mathrm{s}} = 0.218 + c_0 \exp - \frac{(t-c_1)^2}{c_2} \tag{49}$$

The value of the constants c_0, c_1, and c_2 in Equation (49) is specified in Table III. The constant c_0 defines the amplitude of the incident solitary wave, the constant c_1 defines the time at which

Copyright © 2008 John Wiley & Sons, Ltd.

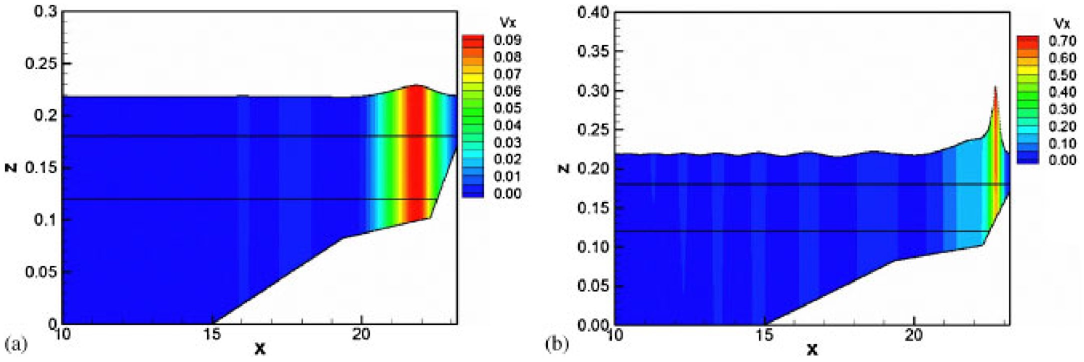

Figure 8. Solitary wave runup. Vertical discretization and longitudinal profile: (a) Case A and (b) Case B.

Table III. Solitary wave runup. Constants used for the boundary condition specification (Equation (49)).

	c_0 (m)	c_1	c_2
Case A	0.0083	1.5	0.5775
Case B	0.0560	1.5	0.1173

the crest of the wave enters the domain (but it does not affect its shape), while the constant c_2 defines the shape of the wave. The value of the constant c_0 has been specified in such a way that the numerical and experimental wave amplitude at wave gauge S4 are the same. In Case B some spurious oscillations of the free surface are generated behind the solitary wave in the offshore boundary. These are probably due to the way in which the wave is generated, imposing only the water depth value, and not the velocity, at the inlet boundary.

The numerical results are compared with the experimental data at wave gauges S5, S7, and S9 in Figure 9. The observed runup at the vertical wall was 2.7 cm for Case A and 45.7 cm for Case B. The computed runup at the wall for Case A was 2.5 cm. Nonetheless, it should be noticed that the runup at the vertical wall is very sensitive to the incoming wave height. In our numerical results for Case A, an increase of just 1 mm (which is of the order of the gauge accuracy) in the incoming wave height at station S4 would produce an increase of approximately 2 mm in the computed runup at the wall and thus, it would produce a perfect match with the experimental data. In the numerical results reported by the participants in the workshop the runup for Case A varies between 1.9 and 2.8 cm depending on the numerical model used and on the offshore boundary condition. In particular we should cite the results of Grilli, who obtained a 2.5 cm runup with a fully non-linear potential flow model (2D in the vertical plane). As we are neglecting in the present simulations the effects of turbulence and bed friction, our numerical results should be very similar to those of Grilli. This is confirmed by the time series of the water depth at gauges S7 and S9, which are extremely similar and consistent with the results of Grilli [27] and also by the runup at the vertical wall.

In Case B the observed runup at the vertical wall (45.7 cm) is extremely high in comparison with the incident wave, and also significantly higher than the computed runup (29 cm). It was

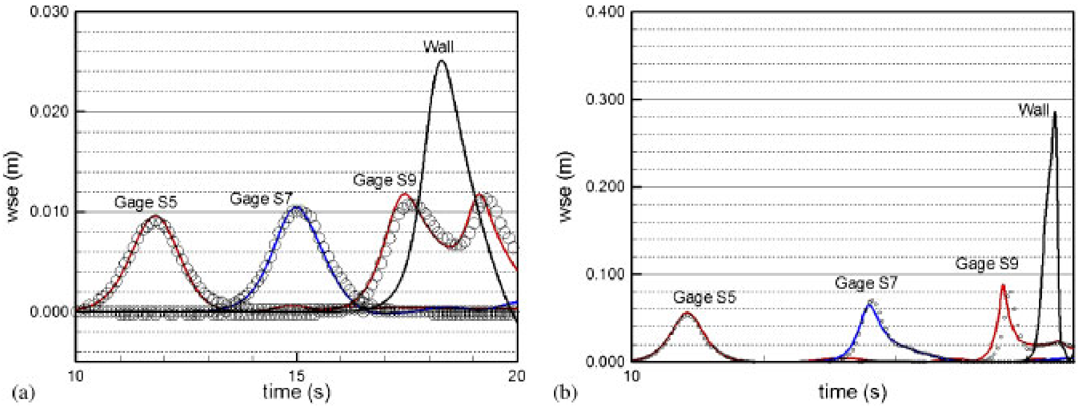

Figure 9. Solitary wave runup. Numerical results (solid line) and experimental data (circles) at wave gauges S5, S7, S9, and wall. The size of the circles is proportional to the experimental uncertainty: (a) Case A and (b) Case B.

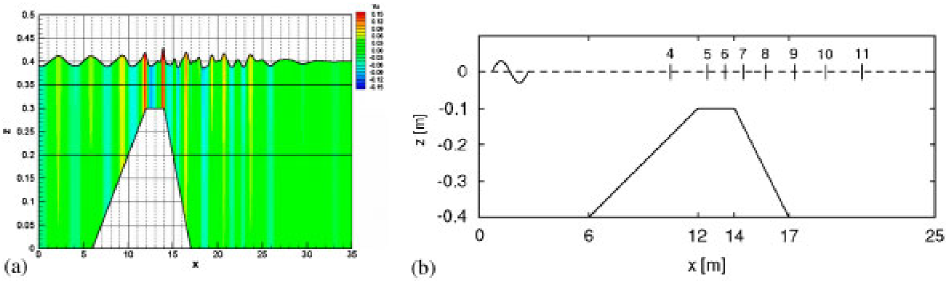

Figure 10. Submerged bar test case: (a) vertical discretization and (b) location of the experimental stations.

suggested by Kanoglu and Synolakis [28] that this drastic increase in the runup happens because the wave breaks right on the wall, causing the collapse of a trapped air bubble under the plunging wave [29]. This process cannot be predicted by a single phase model and therefore, the computed runup at the wall for Case B is lower than the measured one.

4.4. Short wave propagation over a 1D bar

In this test case a non-linear dispersive wave is propagated over a 1D submerged bar. The numerical results are compared with the experimental data obtained by Beji and Battjes [21] in laboratory experiments (Figure 10). The experiments of Beji and Battjes have also been modelled by Stelling and Zijlema [4], Li and Fleming [6], and Lin and Li [7] among others. Two different waves are considered, which correspond to Case A and Case C in [21]. In both cases the water depth is 40 cm. In Case A the wave height is 2 cm and the wave period 2.02 s, which gives a ratio $H/L = 0.0053$. In Case C the wave height is 4.1 cm and the wave period 1.01 s, with a ratio $H/L = 0.027$. The shoaling

DOI: 10.1002/fld

produced by the bar increases the wave height and reduces the wave length, increasing in such a way the ratio H/L and therefore the non-linear effects, boosting the higher harmonics. After the bar the harmonics are dispersed at different wave celerities. This process can only be computed with a non-hydrostatic model.

The time step used in the computations is $\Delta t = 0.01\,\mathrm{s}$ and the spatial discretization size $\Delta x = 0.0125\,\mathrm{m}$. The vertical discretization is done with three horizontal layers (Figure 10). The thickness of each layer is 20 cm (bed layer), 15 cm (middle layer), and 5 cm (surface layer). The spatial domain covers 35 m, including a sponge layer of 10 m located next to the outlet boundary. The number of finite volumes in the numerical mesh is 8400.

Regarding the boundary conditions, at the inlet boundary the free surface level is specified from the sinusoidal wave shape, and the normal pressure and velocity gradients are set to zero. In order to minimize wave reflections at the outlet boundary a sponge layer is used to damp the wave energy, and the free surface elevation is set to 0.4 m. The sponge layer used was proposed by

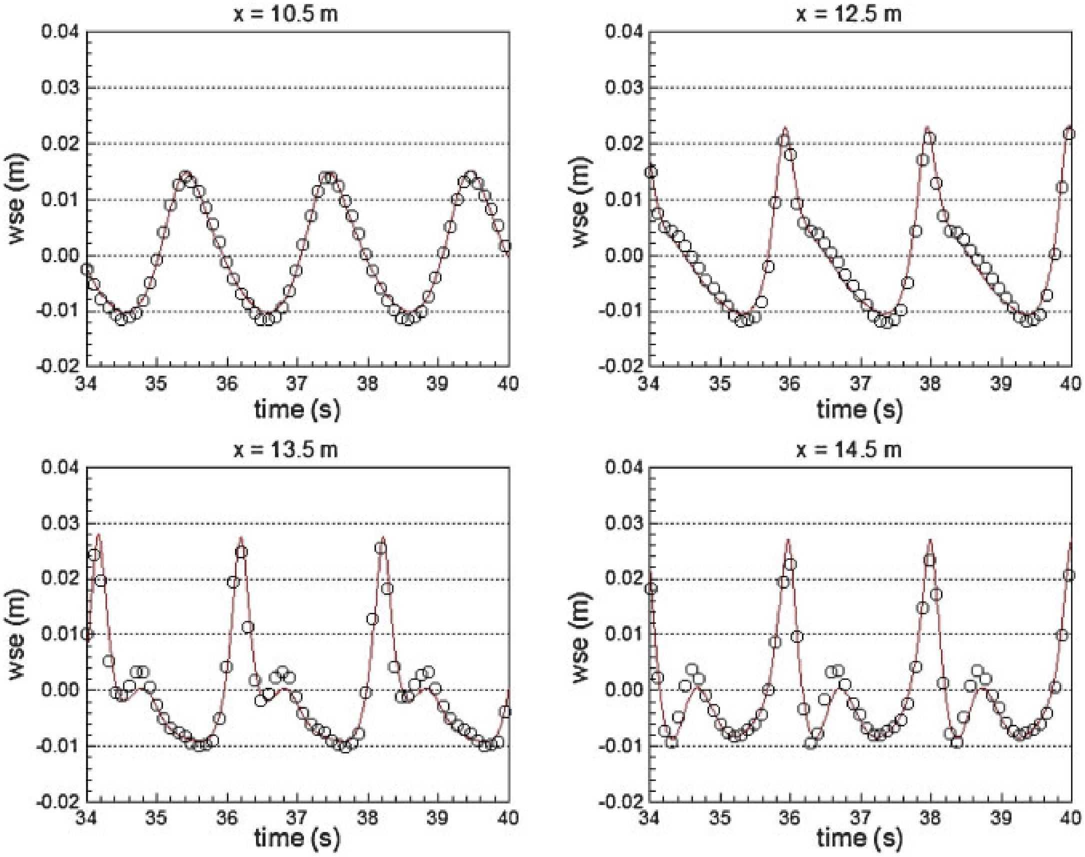

Figure 11. Submerged bar. Numerical results (solid line) and experimental data (circles). Case A. Stations S4, S5, S6 and S7.

DOI: 10.1002/fld

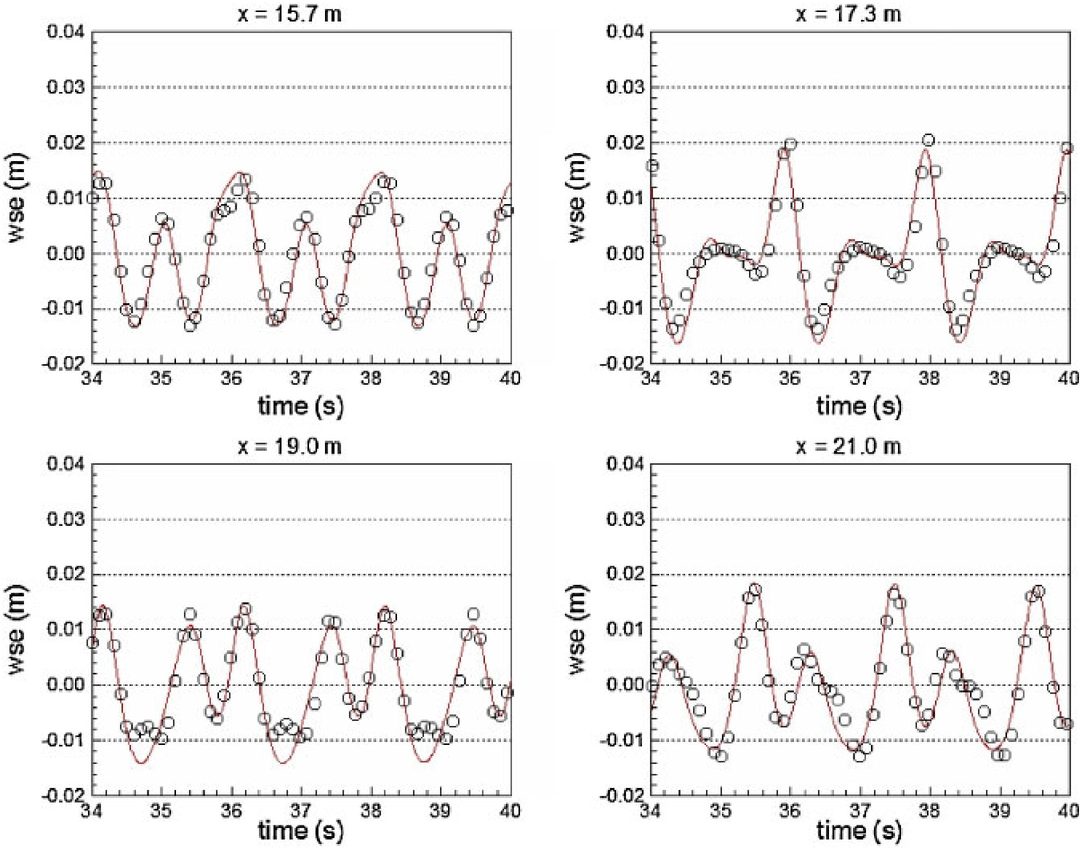

Figure 12. Submerged bar. Numerical results (solid line) and experimental data (circles). Case A. Stations S8, S9, S10 and S11.

Dingemans in [30], being defined by

$$v=\frac{1}{2}\left(\tanh\left(\frac{\sin(\pi\hat{x}/2)}{1-\hat{x}^2}\right)+1\right)$$
$$\hat{x}=4\frac{x-L_0}{L}-1 \quad \text{if } 0<\frac{x-L_0}{L}<0.5 \tag{50}$$
$$\hat{x}=3-4\frac{x-L_0}{L} \quad \text{if } 0.5<\frac{x-L_0}{L}<1$$

where v is a damping coefficient that defines a damping term in the momentum equations given by vu_i, L_0 is the start of the sponge layer, and L is its length. The damping term vu_i enters on the left-hand side of Equations (10)–(12), and it is discretized implicitly.

The agreement between numerical and experimental data is very satisfactory at all the experimental stations for Case A (Figures 11 and 12). The largest disagreements appear at station S10 ($x=19$ m), where a small undulation in the tough of the wave is not captured by the model. Apart

DOI: 10.1002/fld

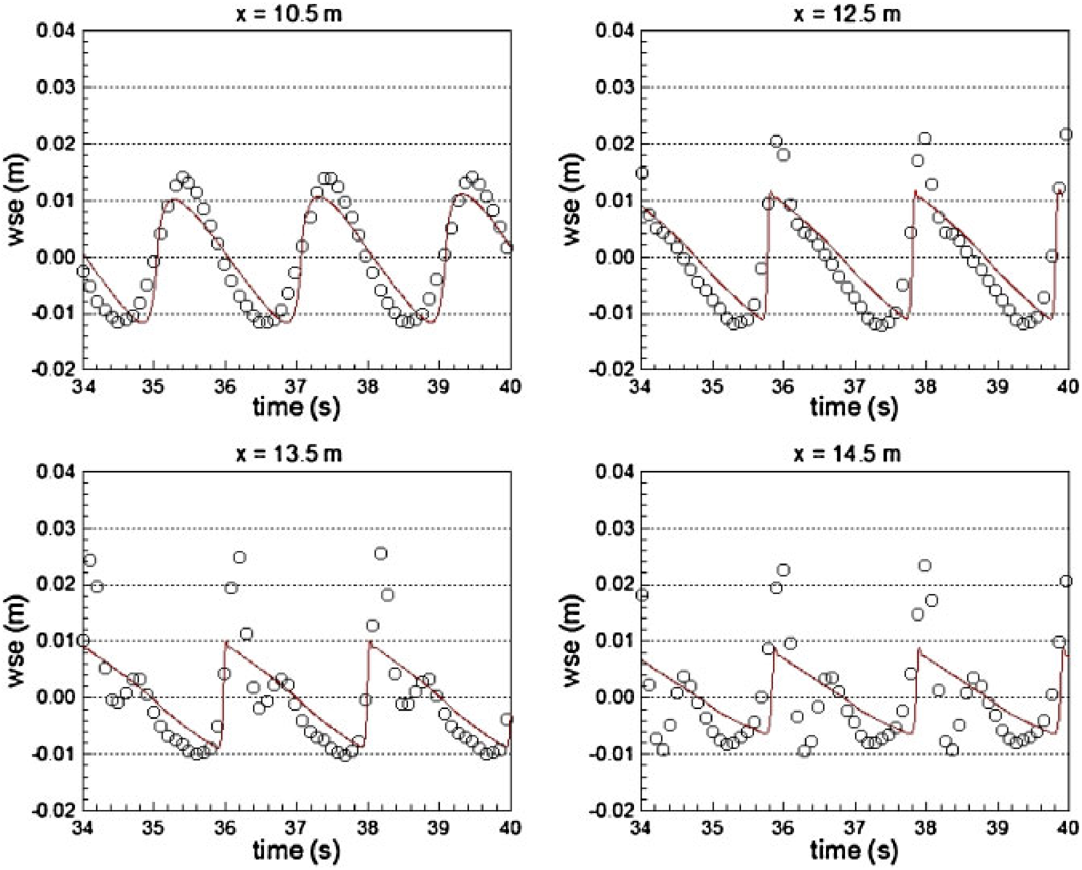

Figure 13. Submerged bar. Numerical results computed with an hydrostatic shallow water model (solid line) and experimental data (circles). Case A. Stations S4, S5, S6 and S7.

from that, all the high-order harmonics generated in the shoaling process are also generated by the model.

As it has been previously said, the propagation of the short wave and the shoaling produced by the bar cannot be computed with an hydrostatic model, because wave dispersion, which is a 3D non-hydrostatic process, plays a very important role in this process. The limitations of a hydrostatic shallow water model are shown in Figure 13. Details of the shallow water model used to compute the numerical results presented in Figure 13 can be found in Reference [31].

Case C is more demanding since the non-linearities are stronger (the ratio H/L is much higher than in Case A). The model gives reasonable results at all the stations (Figures 14 and 15), although some small features of the wave shape are not well captured at the stations S7 ($x=14.5$ m), S8 ($x=15.7$ m), and S9 ($x=17.3$ m). This is the region where the depth increases, producing a dispersion of the different wave harmonics, which have been produced by the shoaling effect of the bar. The largest disagreements appear at station S9. At station S10 ($x=19.0$ m) the wave

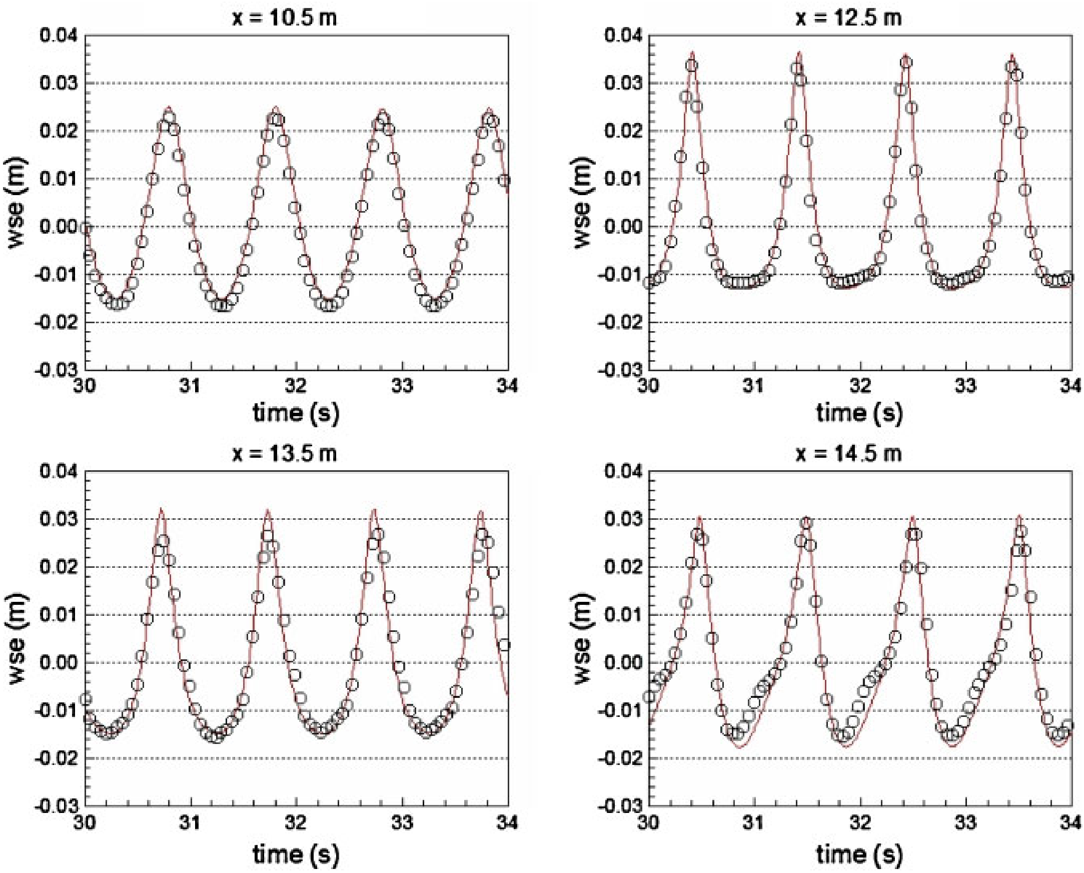

Figure 14. Submerged bar. Numerical results (solid line) and experimental data (circles). Case C. Stations S4, S5, S6 and S7.

amplitude is slightly overpredicted by the model. Despite those small discrepancies the overall agreement with the experimental data is good.

4.5. Short wave propagation over a 2D shoal

This is a fully three-dimensional case in which the capabilities of the model to compute short wave refraction and diffraction are tested. A regular wave with a period of $T = 1\,\mathrm{s}$ and a wave height of $H_0 = 4.64\,\mathrm{cm}$ is propagated over a sloping bed with an elliptical shoal (Figure 16). Similar test cases have also been used to validate wave models by Stelling and Zijlema [4], Zijlema and Stelling [5], Li and Fleming [6], and Saied and Tsanis [32].

The spatial domain is a rectangle of 20 m in the x-direction and 35 m in the y-direction (including a 5 m sponge layer next to the outlet boundary). The bed slope is $i = 0.02$ and the bathymetric lines form an angle of 20° with the x-direction. The bed elevation is given by

$$z_b = -\min[0.45, \max(0.10, 0.45 - 0.02(5.84 + y'))] + d \tag{51}$$

DOI: 10.1002/fld

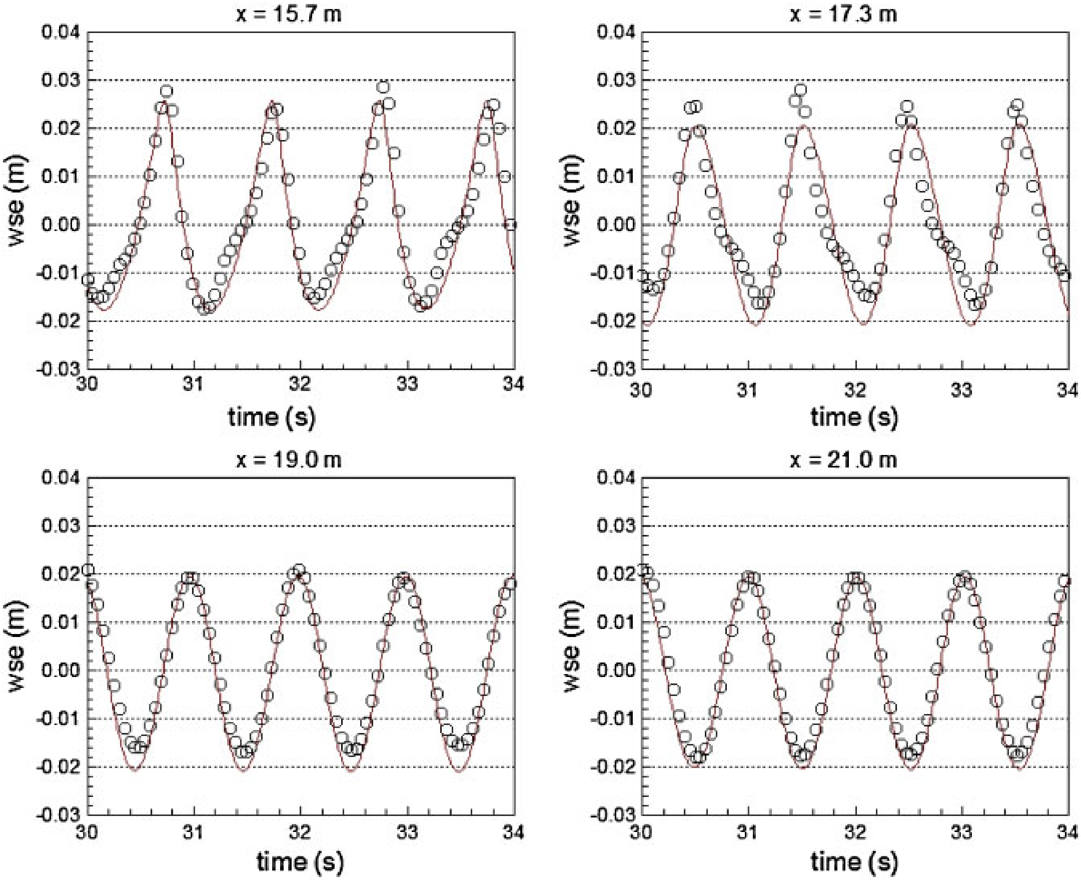

Figure 15. Submerged bar. Numerical results (solid line) and experimental data (circles). Case C. Stations S8, S9, S10 and S11.

where d is the shoal thickness. The coordinates x' and y' are defined by a 20° clockwise rotation of the x, y coordinates:

$$x' = x\cos 20° - y\sin 20°, \quad y' = x\sin 20° + y\cos 20° \tag{52}$$

The shoal has an elliptical shape given by

$$\left(\frac{x'}{4}\right)^2 + \left(\frac{y'}{3}\right)^2 = 1 \tag{53}$$

and a thickness defined by:

$$d = -0.30 + 0.50\sqrt{1 - \left(\frac{x'}{5}\right)^2 - \left(\frac{y'}{3.75}\right)^2} \tag{54}$$

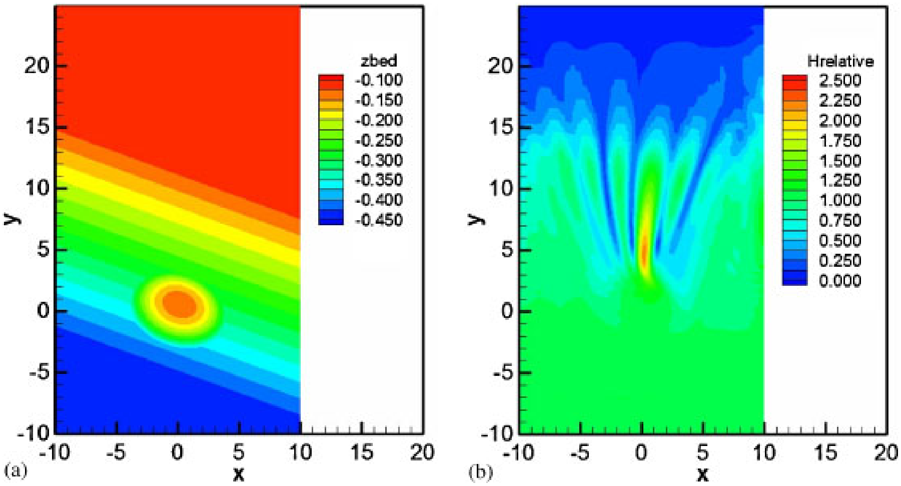

Figure 16. 2D elliptical shoal test case: (a) bed elevation (m) and (b) relative wave height.

A time step of $\Delta t = 0.005$ s and a cell size of $\Delta x = \Delta y = 0.05$ m (280 000 rectangular cells on each layer) have been used in the computations. The vertical discretization is done with 3 horizontal layers with a thickness of 25 cm (bed layer), 12 cm (middle layer), and 8 cm (surface layer). This gives a total number of 840 000 finite volumes. The Gamma scheme with a second-order time discretization was used in the computations. Boundary conditions are specified in a similar manner as in the submerged bar test case.

The computation is started from a hydrostatic situation (still water) and run over 50 seconds, after which a steady situation has been reached. The relative wave height field is shown in Figure 16, where the relative wave height H_{r} is computed as the difference between the maximum water surface elevation and the minimum wave elevation, divided by the incident wave height ($H_{\mathrm{r}} = (z_{\mathrm{s,max}} - z_{\mathrm{s,min}})/H_0$). The effect of wave refraction as well as the shadow rays caused by diffraction can be clearly observed in Figure 16. The comparison between numerical and experimental relative wave height at different sections is quite satisfactory (Figure 17), being the maximum wave height as well as the interference patterns caused by diffraction well captured by the model.

5. CONCLUSIONS

In this paper a 3D free surface non-hydrostatic layer-structured finite volume model has been presented. The model, which solves the 3D Euler equations for incompressible flow, has been applied to several test cases that involve non-linear short wave propagation, shoaling, refraction, and diffraction processes. The flow equations are discretized in a collocated mesh, which is unstructured in the horizontal plane, but structured in the vertical direction. Pressure–velocity coupling is achieved by the SIMPLE algorithm, using Rhie–Chow interpolation to stabilize the pressure field. The model reduces to a 2D depth-averaged shallow water model when one single layer is defined for the vertical discretization.

DOI: 10.1002/fld

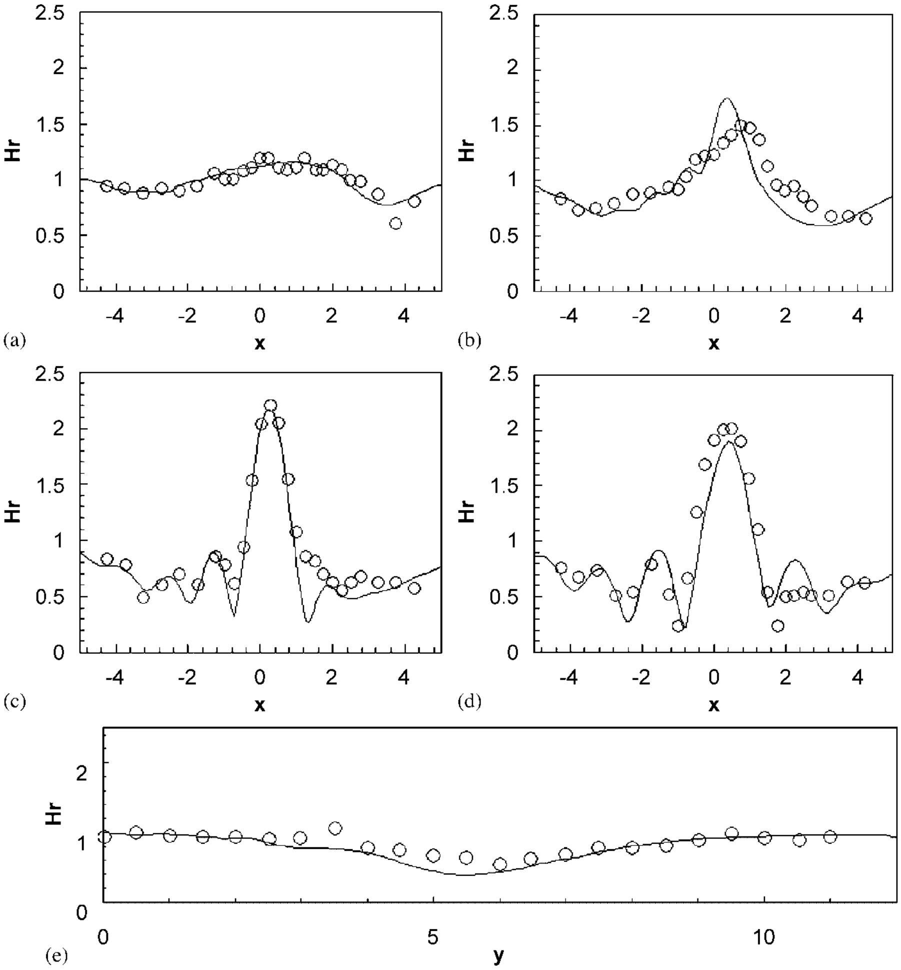

Figure 17. 2D elliptical shoal. Experimental validation: (a) section $y=1\,\mathrm{m}$; (b) section $y=3\,\mathrm{m}$; (c) section $y=5\,\mathrm{m}$; (d) section $y=7\,\mathrm{m}$; and (e) section $x=-2\,\mathrm{m}$.

All the numerical results have been compared with either analytical solutions or with experimental data, showing in all cases a very satisfactory agreement with only three horizontal layers in the numerical mesh. This means that the model can take into account the effects of non-hydrostatic pressure distribution with a rather coarse vertical discretization. Vertical refinement might be needed for applications in which it is necessary to resolve flow patterns in the vertical plane, but it is not a requirement otherwise, even if the pressure distribution is non-hydrostatic.

It has been shown by the numerical results that the correct implementation of the pressure boundary condition at the free surface is vital in order to obtain an accurate wave celerity with just a few horizontal layers. The zero pressure boundary condition must be imposed at the face of the free surface volumes, and not at the volume node. Otherwise the wave celerity is computed erroneously unless a very fine vertical discretization is used, which increases considerably the computational cost. In the numerical results it has also been noticed the importance of using an explicit discretization of the free surface gradient (hydrostatic pressure source term). An implicit discretization of that term introduces too much damping in the solution, which gives too small wave heights.

Although shear stresses can be neglected in all the situations studied in this paper, they can be easily incorporated in the model by just adding a diffusion term to the momentum equations. Wave–current iteration is also considered naturally by the equations. Those are great advantages of these kind of wave models.

Finally, it should be noticed that even if most of the test cases presented in this paper have been computed with a structured mesh, the code can handle accurately unstructured meshes made of triangular prisms. This has been shown in test cases number 1 and 2, which involve the computation of a standing short wave in a closed basin and the computation of a radially symmetric wave. Therefore, the model is suitable for computing short wave propagation over irregular geometries.

REFERENCES

1. Olsen NRB, Kjellesvig HM. Three-dimensional numerical flow modelling for estimation of spillway capacity. *Journal of Hydraulic Research* 1998; **36**(5):775–784.
2. Wu W, Rodi W, Wenka T. 3D numerical modeling of flow and sediment transport in open channels. *Journal of Hydraulic Engineering* 2000; **126**(1):4–15.
3. Casulli V, Zanolli P. Semi-implicit numerical modeling of nonhydrostatic free-surface flows for environmental problems. *Mathematical and Computer Modelling* 2002; **36**:1131–1149.
4. 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.
5. Zijlema M, Stelling G. Further experiences with computing non-hydrostatic free-surface flows involving water waves. *International Journal for Numerical Methods in Fluids* 2005; **48**:167–197.
6. Li B, Fleming CA. Three-dimensional model of Navier–Stokes equations for water waves. *Journal of the Waterway*, *Port*, *Coastal*, *and Ocean Engineering* 2001; **127**:16–25.
7. 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.
8. Monaghan JJ. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics* 1992; **30**: 543–574.
9. Idelsohn SR, Storti MA, Oñate E. Lagrangian formulations to solve free surface incompressible inviscid fluid flows. *Computer Methods in Applied Mechanics and Engineering* 2001; **191**:583–593.
10. Tryggvason G, Bunner B, Esmaeeli A, Juric D, Al-Rawahi N, Tauber W, Nas JHS, Yan YJ. A front tracking method for the computations of multiphase flow. *Journal of Computational Physics* 2001; **169**:708–759.
11. Hirt CW, Nichols BD. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
12. Nichols BD, Hirt CW. Calculating three-dimensional free surface flows in the vicinity of submerged and exposed structures. *Journal of Computational Physics* 1973; **12**:234–246.
13. Rhie CM, Chow WL. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal* 1983; **21**(11):1525–1532.
14. Jasak H, Weller HG, Gosman AD. High resolution NVD differencing scheme for arbitrarily unstructured meshes. *International Journal for Numerical Methods in Fluids* 1999; **31**:431–449.
15. Leonard BP. Simple high-accuracy resolution program for convective modelling of discontinuities. *International Journal for Numerical Methods in Fluids* 1988; **8**:1291–1318.
16. Gaskell PH, Lau AKC. Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm. *International Journal for Numerical Methods in Fluids* 1988; **8**:617–641.
17. Patankar SV. *Numerical Heat Transfer and Fluid Flow*. McGraw-Hill: New York, 1980.
18. Vuik C, van der Vorst HA. A comparison of some GMRES-like methods. *Linear Algebra and its Applications* 1992; **160**:131–162.

DOI: 10.1002/fld

19. Casulli V, Stelling GS. Numerical simulation of 3D quasi-hydrostatic free-surface flows. *Journal of Hydraulic Engineering* 1998; **124**(7):678–686.
20. 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.
21. Beji S, Battjes JA. Experimental investigation of wave propagation over a bar. *Coastal Engineering* 1993; **19**:151–162.
22. Berkhoff CW, Booy N, Radder AC. Verification of numerical wave propagation models for simple harmonic. *Coastal Engineering* 1982; **6**:255–279.
23. Williams JJR. Free-surface simulations using an interface-tracking finite-volume method with 3D mesh movement. *Engineering Applications of Computational Fluid Mechanics* 2007; **1**(1):49–56.
24. Thomas TG, Leslie DC, Williams JJR. Free surface simulations using a conservative 3D code. *Journal of Computational Physics* 1995; **116**(1):52–68.
25. Briggs MJ, Synolakis CE, Kanoglu U, Green DR. Benchmark problem 3: runup of solitary waves on a vertical wall. In *Long-wave Runup Models*, Yeh H, Liu P, Synolakis C (eds). World Scientific Publishing: Singapore, 1997; 375–383.
26. Beck BC, Baptista AM. WET2: an Eulerian–Lagrangian shallow water FEM model. In *Long-wave Runup Models*, Yeh H, Liu P, Synolakis C (eds). World Scientific Publishing: Singapore, 1997; 265–271.
27. Grilli S. Fully nonlinear potential flow models used for long wave runup prediction. In *Long-wave Runup Models*, Yeh H, Liu P, Synolakis C (eds). World Scientific Publishing: Singapore, 1997; 116–180.
28. Kanoglu U, Synolakis CE. Long wave runup on piecewise linear topographies. *Journal of Fluid Mechanics* 1998; **374**:1–28.
29. Topliss ME, Cooker MJ, Peregrine DH. Pressure oscillations during wave impact on vertical walls. In *Proceedings of the Twenty-Third International Coastal Engineering Conference*, Edge BL (ed.), vol. 2. ASCE: New York, 1992; 1639–1650.
30. Dingemans MW. *Water Wave Propagation Over Uneven Bottoms*. World Scientific: Singapore, 1997.
31. Cea L, Ferreiro A, Vázquez-Cendón ME, Puertas J. Experimental and numerical analysis of solitary waves generated by bed and boundary movements. *International Journal for Numerical Methods in Fluids* 2004; **46**(8):793–813.
32. Saied UM, Tsanis IK. Improved parabolic water wave transformation model. *Coastal Engineering* 2005; **52**: 139–149.