A Volume-of-Fluid based simulation method for wave impact problems

K.M.T. Kleefsman [a,*], G. Fekken [a], A.E.P. Veldman [a],
B. Iwanowski [b], B. Buchner [c]

[a] *Department of Mathematics and Computing Science, University of Groningen, P.O. Box 800, 9700 AV Groningen, The Netherlands*
[b] *FORCE Technology Norway AS, Claude Monets allé 5, 1338 Sandvika, Norway*
[c] *Maritime Research Institute Netherlands, P.O. Box 28, 6700 AA Wageningen, The Netherlands*

Abstract

In this paper, some aspects of water impact and green water loading are considered by numerically investigating a dambreak problem and water entry problems. The numerical method is based on the Navier–Stokes equations that describe the flow of an incompressible viscous fluid. The equations are discretised on a fixed Cartesian grid using the finite volume method. Even though very small cut cells can appear when moving an object through the fixed grid, the method is stable. The free surface is displaced using the Volume-of-Fluid method together with a local height function, resulting in a strictly mass conserving method. The choice of boundary conditions at the free surface appears to be crucial for the accuracy and robustness of the method. For validation, results of a dambreak simulation are shown that can be compared with measurements. A box has been placed in the flow, as a model for a container on the deck of an offshore floater on which forces are calculated. The water entry problem has been investigated by dropping wedges with different dead-rise angles, a cylinder and a cone into calm water with a prescribed velocity. The resulting free surface dynamics, with the sideways jets, has been compared with photographs of experiments. Also a comparison of slamming coefficients with theory and experimental results has been made. Finally, a drop test with a free falling wedge has been simulated.
© 2005 Elsevier Inc. All rights reserved.

---

[*] Corresponding author. Tel.: +31 50 363 3972; fax: +31 50 363 3800.
E-mail address: theresa@math.rug.nl (K.M.T. Kleefsman).

1. Introduction

During the last decades increasing attention has been paid to the development of calculation methods for wave impact problems. In this paper a prediction method based on the Navier–Stokes equations is described. A special application of the method can be found in the maritime industry, where also the validation cases presented in this paper find their origin.

In the offshore industry a structure is placed in an oil or gas field for several years, often for over twenty years. These structures must survive all weather types, including heavy storms. In high or steep waves offshore structures face the problem of wave impact on their bow or bottom. One reported case is the Schiehallion FPSO where a rather steep wave had damaged the bow, which finally resulted in an evacuation of all personnel [17]. Furthermore, the water can exceed the deck level of a structure in high waves, resulting in the flow of a large amount of water on the deck. This is called green water and often causes damage to deck houses or other equipment on the deck [8,31]. A comprehensive research of green water on the deck of ship-type offshore structures has been conducted by Buchner [3]. There is a great need of simulation tools that can predict the impact loads of green water or steep waves and give more insight in the local impact phenomena.

A good prediction method is based on the Navier–Stokes equations, which describe the motion of an incompressible, viscous fluid, with a free liquid surface. Many methods for the treatment of the free surface are described in the literature. The most popular ones are the level-set method and the Volume-of-Fluid (VOF) method; see below. An overview of the various methods available can be found in [37].

In the level-set formulation a distance function $\phi(x,t)$ is introduced denoting the distance from x to the initial interface location at $t = 0$. The interface corresponds to the contour $\phi = 0$ at any instant [33]. In this method highly distorted interfaces can be treated and also topology changes are incorporated automatically. Although the interface is of finite thickness, the physical properties such as surface tension can be applied easily. A major problem of the level-set method is the lack of mass conservation. Several strategies have been studied to overcome this problem, e.g. a combination with a VOF method has been used [42] or a re-distancing algorithm [41]. In our application area Iafrati et al. [24] used the level-set technique for unsteady free-surface flows, where results of flow over a jump and flow inside a tank have been shown.

In the VOF method, first introduced by Hirt and Nichols [23], a VOF function F is introduced with values between zero and one, indicating the fractional volume of a cell that is filled with a certain fluid. The evolution of the VOF function is given by $DF/Dt = 0$. In every cell the VOF function is reconstructed. There to different methods can be applied; see [35] for an overview. The most commonly used methods are piecewise constant reconstruction, where the interface is said to be parallel to one of the coordinate axes, and piecewise linear reconstruction (PLIC, [44]). Based on the reconstructed interface and the velocity field fluxes are computed at cell faces and the fluid is moved from a donor cell to an adjacent acceptor cell. In the original VOF method, where a piecewise constant, stair-stepped reconstruction of the interface is used [35], a lot of 'flotsam' and 'jetsam' (small droplets disconnecting from the free surface) can occur. These droplets or small air-pockets, sometimes called 'wisps', are more present in the lower order methods and originate in the calculation of the fluxes [21,35]. In these methods also liquid can be lost or gained due to rounding the VOF function when $F < 0$ or $F > 1$.

In the method described in this paper the Navier–Stokes equations for an incompressible, viscous flow are solved. There to the flow domain is covered with a fixed Cartesian grid with staggered variables. The equations are discretised in space using the finite volume method. To overcome the problems described above, the VOF method is adapted with a local height function; see Gerrits [12] for a detailed description. The method has been incorporated in a simulation method called COMFLOW. The method was first developed for the simulation of sloshing on board spacecraft [13–15]. Another application has been found in medical science, where blood flow through elastic arteries has been studied [28–30]. The subject of this

paper is inspired by engineering problems from the maritime industry and builds on pilot studies of green water [11] and flow in anti-roll tanks [6,7].

Recently, the method has been extended with moving objects [10]. The objects are moving either with a prescribed motion or the motion is calculated from the interaction between the object and the liquid dynamics. A cut-cell method is used in which the boundaries of the object are kept sharp. Other popular methods exist in the literature. One is the immersed boundary method developed by Peskin [34] since the 1970s and used, for e.g. [26,27]. A more recent method is the fictitious domain method of Glowinsky et al. [16].

The VOF free-surface displacement method is tested using a well-known advection test for solid body rotation of a slotted disk. The overall numerical convergence of the simulation method has been checked by performing simulations of regular waves with increasing resolution in time and space.

For further validation two series of simulations are presented in this paper. The first is a simulation of a breaking dam flow with a box present in the domain. The computational results have been compared with experiments carried out by MARIN. The simulation can be seen as a model for the flow onto the deck of a vessel due to green water. The second series of simulations consists of drop tests, where wedges with different dead-rise angles, a circular cylinder and a cone are entering the water. Snapshots of the simulations have been compared with photographs of experiments by Greenhow and Lin [18]. The slamming coefficients of a surface penetrating cone have been computed for different entry velocities and compared to the theory of Schiffman and Spencer [38]. The experiments of Campbell and Weynberg [4] and the theory of Faltinsen [9] have been used to check the slamming coefficient for the entry of a circular cylinder. Results of a free falling wedge, where a full coupling between the liquid motion and the body motion has been established, have been compared with experimental results of Zhao et al. [45].

2. Governing equations

Flow of a homogeneous, incompressible, viscous fluid is described by the continuity equation and the Navier–Stokes equations, describing conservation of mass and momentum, respectively. In a conservative form they are given by:

$$\oint_{\partial V} \boldsymbol{u} \cdot \boldsymbol{n} \, \mathrm{d}S = 0, \tag{1}$$

$$\int_V \frac{\partial \boldsymbol{u}}{\partial t} \, \mathrm{d}V + \oint_{\partial V} \boldsymbol{u}\boldsymbol{u}^{\mathrm{T}} \cdot \boldsymbol{n} \, \mathrm{d}S = -\frac{1}{\rho} \oint_{\partial V} (p\boldsymbol{n} - \mu \nabla \boldsymbol{u} \cdot \boldsymbol{n}) \, \mathrm{d}S + \int_V \boldsymbol{F} \, \mathrm{d}V. \tag{2}$$

Here, ∂V is the boundary of volume V, $\boldsymbol{u} = (u,v,w)$ is the velocity in the three coordinate directions, $\boldsymbol{n}$ is the normal at the boundary ∂V, ρ is the density, p is the pressure and ∇ is the gradient operator. Further, μ denotes the dynamic viscosity and $\boldsymbol{F} = (F_x, F_y, F_z)$ is an external body force, for example gravity.

2.1. Boundary conditions

At the solid walls of the computational domain and at the objects inside the domain a no-slip boundary condition is used. This condition is described by $\boldsymbol{u} = 0$ for fixed boundaries and $\boldsymbol{u} = \boldsymbol{u}_b$ for moving objects with $\boldsymbol{u}_b$ being the object velocity.

Some of the domain boundaries may let fluid flow in or out of the domain. Especially, when performing wave simulations an inflow boundary is needed where the incoming wave is prescribed, and at the opposite boundary a non-reflecting outflow condition should be used. In our method the wave on the inflow

boundary can be prescribed as a regular linear wave or a regular 5th order Stokes wave [39]. Also a super-position of linear components can be used, which results in an irregular wave.

At the outflow boundary a non-reflecting boundary condition is needed to prevent the waves from reflecting from the boundary into the domain. A Sommerfeld condition (see, e.g. [40]) is very appropriate in cases where a regular wave is used. This condition is based on the wave equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0, \tag{3}$$

which is discretised to determine the velocities at the outflow boundary. The wave velocity c has to be pre-scribed (e.g. the wave velocity of an incoming regular wave). In the case of an irregular wave or a much deformed regular wave (e.g. due to the presence of an object in the flow) a damping zone is added at the end of the domain. Such a damping zone, as introduced by, e.g. [5,25] is a very effective and robust method for preventing wave reflections from the domain boundaries.

2.2. Free surface

When the position of the free surface is given by $s(\boldsymbol{x},t) = 0$, the displacement of the free surface is described by the following equation:

$$\frac{\mathrm{D}s}{\mathrm{D}t} = \frac{\partial s}{\partial t} + (\boldsymbol{u} \cdot \nabla)s = 0. \tag{4}$$

At the free surface boundary conditions are necessary for the pressure and the velocities. Continuity of normal and tangential stresses lead to the equations:

$$-p + 2\mu\frac{\partial u_{\mathrm{n}}}{\partial n} = -p_0 + \sigma\kappa, \tag{5}$$

$$\mu\left(\frac{\partial u_{\mathrm{n}}}{\partial t} + \frac{\partial u_{\mathrm{t}}}{\partial n}\right) = 0. \tag{6}$$

Here, u_{n} and u_{t} are the normal and tangential component of the velocity, respectively, p_0 is the atmospheric pressure, σ is the surface tension and κ denotes the total curvature of the free surface.

3. Numerical model

To solve the Navier–Stokes equations numerically the computational domain is covered with a fixed Cartesian grid. The variables are staggered as in the original MAC method [20], which means that the velocities are defined on cell faces, whereas the pressure is defined in cell centres.

The body geometry is piecewise linear and cuts through the fixed rectangular grid. Volume apertures (F^{b}) and edge apertures (A^x, A^y and A^z) are used to indicate for every cell which part of the cell and cell face, respectively, is open for fluid and which part is blocked. To track the free surface the VOF function F^{s} is used, which is 0 if no fluid is present in the cell, 1 if the open part of the cell is completely filled with fluid and between 0 and 1 if the cell is partly filled with fluid.

In every cell containing fluid the Navier–Stokes equations are solved. Cell labelling is introduced to distinguish between cells of different characters. First the cells that are completely blocked by geometry are called B(oundary) cells. These cells have volume aperture $F^{\mathrm{b}} = 0$. Then the cells that are empty but have the possibility of letting fluid flow in are labelled E(mpty). The adjacent cells containing fluid are S(urface) cells. The remaining cells are labelled as F(luid) cells. Note that these cells do not have to be completely

filled with fluid. In Fig. 1 an example of the cell labelling is given. The velocities that are positioned at the cell faces are labelled using the cell labels. For example, the velocity between an F-cell and an S-cell is called an FS-velocity.

3.1. Discretisation of the continuity equation

The continuity and momentum equations are discretised using the finite volume method. The natural form of the equations when using the finite volume method is the conservative formulation as given in Eqs. (1) and (2). In this paper the discretisation is explained in two dimensions; it can be extended to three dimensions in a straightforward manner. In Fig. 2, a computational cell is shown, which is cut by the body geometry. When applying conservation of mass in this cell, the discretisation results in

$$u_e A_e^x \delta y + v_n A_n^y \delta x - u_w A_w^x \delta y - v_s A_s^y \delta x + l(\boldsymbol{u}_b \cdot \boldsymbol{n}_b) = 0, \tag{7}$$

where the notation is explained in Fig. 2 and $\boldsymbol{u}_b = (u_b, v_b)$. The normal of the boundary is given by

$$\boldsymbol{n}_b = (\delta y(A_e^x - A_w^x), \delta x(A_n^y - A_s^y)) / \|\delta y(A_e^x - A_w^x), \delta x(A_n^y - A_s^y)\|. \tag{8}$$

Recognising l in the denominator of $\boldsymbol{n}_b$, the discrete continuity equation (Eq. (7)) can be written as

$$u_e A_e^x \delta y + v_n A_n^y \delta x - u_w A_w^x \delta y - v_s A_s^y \delta x + u_b(A_e^x - A_w^x)\delta y + v_b(A_n^y - A_s^y)\delta x = 0. \tag{9}$$

3.2. Discretisation of the Navier–Stokes equations

The Navier–Stokes equations are discretised in a control volume around a velocity that is defined at a cell face. In the case of uncut cells this simply means that the control volume consists of half of the neighbouring cells to the left and to the right of the velocity (see the left of Fig. 3). In the case of cut cells the control volumes are defined as half of the open part of the left neighbour cell and half of the open part of the right neighbour cell (see the right of Fig. 3 for an example). Now, the different terms of the momentum equation are discretised using these control volumes. The discretisation is explained for the momentum equation in x-direction.

The volume integral of the time derivative is discretised in space using the midpoint rule. This results in

$$\int_V \frac{\partial u}{\partial t} \, dV \doteq \frac{\partial u_c}{\partial t} F_c^b \delta x_c \delta y. \tag{10}$$
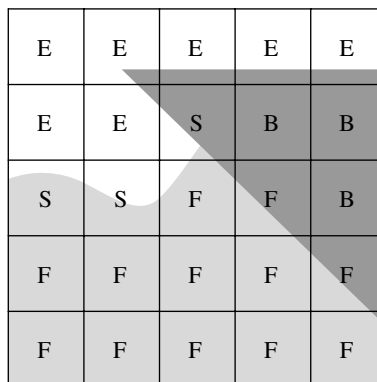

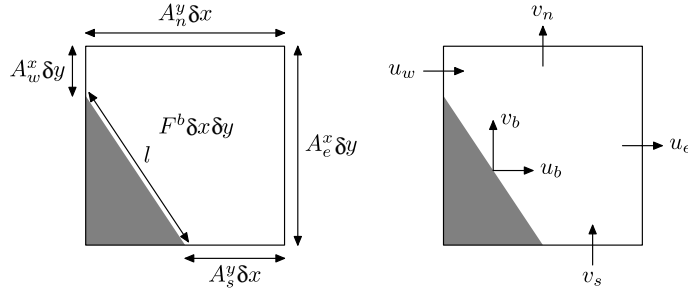Fig. 1. Cell labelling: dark grey denotes solid body, light grey is liquid.

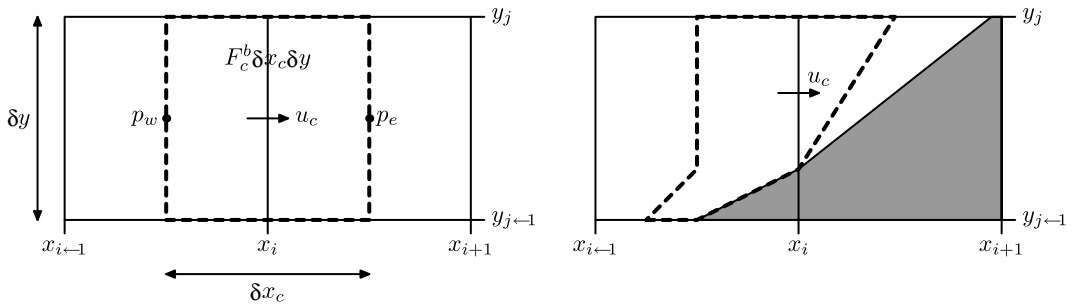Fig. 2. Conservation cell for the continuity equation.

Fig. 3. A control volume for the discretisation of the Navier–Stokes equation in x-direction in the case of an uncut cell (left) and a cut cell (right).

Here, u_c is the central velocity around which the control volume is placed and $F_c^b \delta x_c \delta y$ is the volume of the control volume (see Fig. 3 for the notation). The volume aperture F_c^b of the control volume is defined as $F_c^b = \frac{1}{2}(F_e^b + F_w^b)$ with F_e^b and F_w^b the volume apertures of the eastern and western cell, respectively.

The convective term is discretised directly from the boundary integral, which is given by

$$\oint_{\partial V} u\boldsymbol{u} \cdot \boldsymbol{n} \, dS. \tag{11}$$

Note that this integral has two different velocities in it: the scalar velocity u is advected with the velocity vector $\boldsymbol{u}$. This integral is evaluated at all boundaries of the control volume by multiplying the mass fluxes through the boundaries, m_r, m_d, m_l and m_u, which are mass fluxes through the right, lower, left and upper boundaries, respectively, with the scalar horizontal velocities at the different boundaries. Using the notation of Fig. 4, the discretisation is given by

$$\oint_{\partial V} u\boldsymbol{u} \cdot \boldsymbol{n} \, dS \doteq m_r u_r - m_d u_d - m_l u_l + m_u u_u = \frac{1}{2}m_r u_e - \frac{1}{2}m_d u_s - \frac{1}{2}m_l u_w + \frac{1}{2}m_u u_n + \left(\frac{1}{2}m_r - \frac{1}{2}m_d - \frac{1}{2}m_l + \frac{1}{2}m_u\right)u_c$$

$$\tag{12}$$

with the mass fluxes given by:

$$m_r = \frac{1}{2}\left(A_e^x u_e \delta y + A_c^x u_c \delta y + \left(A_c^x - A_e^x\right)u_r^b \delta y\right), \tag{13}$$

$$m_d = \frac{1}{2}\left(A_{se}^y v_{se} \delta x_e + \max\left(0, \left(A_{ne}^y - A_{se}^y\right)v_r^b \delta x_e\right) + A_{sw}^y v_{sw} \delta x_w + \max(0, \left(A_{nw}^y - A_{sw}^y\right)v_l^b \delta x_w)\right), \tag{14}$$
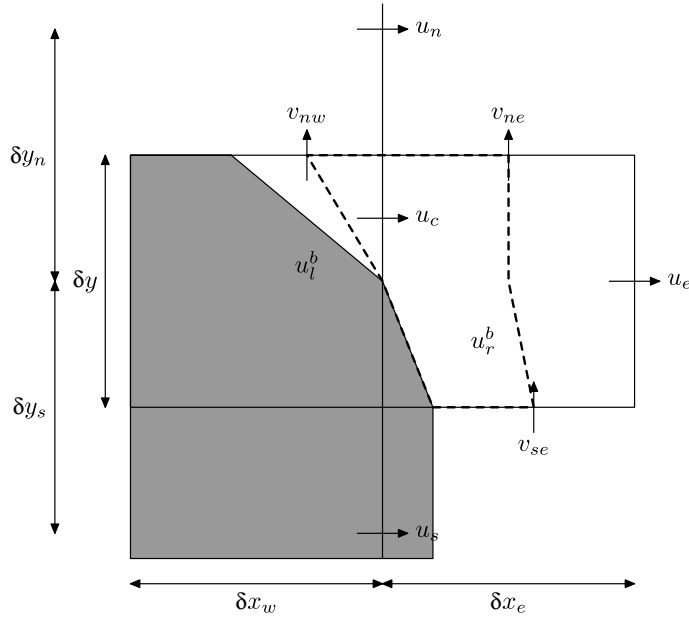
Fig. 4. Control volume for the discretisation of convective terms.

$$m_l = \tfrac{1}{2}\big(A^x_w u_w \delta y + A^x_c u_c \delta y + \big(A^x_c - A^x_w\big)u^b_l \delta y\big), \tag{15}$$

$$m_u = \tfrac{1}{2}\big(A^y_{ne} v_{ne} \delta x_e + \max\big(0, \big(A^y_{se} - A^y_{ne}\big)v^b_r \delta x_e\big) + A^y_{nw} v_{nw} \delta x_w + \max\big(0, \big(A^y_{sw} - A^y_{nw}\big)v^b_l \delta x_w\big)\big). \tag{16}$$

Here, A^x_e is the aperture belonging to the cell face at which u_e is defined, etc. The mass fluxes through the right and left boundary consist of a part of the fluid flow through the open boundaries and a part of the moving body. The mass flux due to the moving body is positive when the body is moving out of the cell. This results for m_r in the term $(A^x_c - A^x_e)u^b_r$, where u^b_r is the velocity of the body in the eastern cell. The part of the fluxes due to the moving body through the upper and lower cell boundaries contains a max-function that distinguishes between the situation that the body is moving into or out of the cell. In the coefficient of the central velocity u_c in Eq. (12) the contributions of the max-functions cancel each other, after which the discrete continuity equations (Eq. (9)) for the eastern and western cell can be recognised. Since both are zero, the total central coefficient is zero. When examining the coefficients of the velocities u_e, u_s, u_w and u_n, it can be clearly seen that the resulting matrix containing the convective coefficients is skew symmetric, which is also a property of the continuous convective operator [43].

For the diffusive term a discretisation is adopted in which the geometry is handled in a staircase way, so the cut cells are treated as if they were uncut. This method has been chosen to prevent instabilities, which can occur in the discretisation when a division is needed by the volume of a cut cell that can be arbitrary small. The diffusive term is rewritten to a volume integral, which for the Navier–Stokes equation in x-direction gives

$$\frac{1}{\rho} \oint_{\partial V} \mu \nabla u \cdot \boldsymbol{n} \, \mathrm{d}S = \frac{1}{\rho} \int_V \nabla \cdot \mu \nabla u \, \mathrm{d}V. \tag{17}$$

The midpoint rule has been used, which is also adopted for the spatial discretisation of the time derivative. In two dimensions the integrand can be written as the sum of the second order horizontal and vertical

derivatives of the horizontal velocity. Using the same notation as for the convective terms (see Fig. 4), the first order derivatives are calculated at the boundaries of the control volume. The first order derivatives are used to calculate the second order derivative, which is positioned at the location of u_c in an uncut cell. And to prevent division by short distances, the first order derivatives are positioned in the cell centres. This results in

$$\nabla \cdot \mu \nabla u \doteq \frac{1}{\delta x_c} \left(\mu_r \frac{u_e - u_c}{\delta x_e} - \mu_l \frac{u_c - u_w}{\delta x_w} \right) + \frac{1}{\delta y_c} \left(\mu_u \frac{u_n - u_c}{\delta y_n} - \mu_d \frac{u_c - u_s}{\delta y_s} \right) \tag{18}$$

with $\delta x_c = \frac{1}{2}(\delta x_e + \delta x_w)$ and $\delta y_c = \frac{1}{2}(\delta y_n + \delta y_s)$ and μ_r to μ_d defined at the same positions as the mass fluxes m_r to m_d. Now, the discretised diffusive term can be written as

$$\frac{1}{\rho} \int_V \nabla \cdot \mu \nabla u \, dV \doteq \frac{1}{\rho} F_c^b \delta x_c \delta y_c \left(\frac{\mu_r}{\delta x_c \delta x_e} u_e + \frac{\mu_l}{\delta x_c \delta x_w} u_w + \frac{\mu_u}{\delta y_c \delta y_n} u_n + \frac{\mu_d}{\delta y_c \delta y_s} u_s \right.$$
$$\left. - \left(\frac{\mu_r}{\delta x_c \delta x_e} + \frac{\mu_l}{\delta x_c \delta x_w} + \frac{\mu_u}{\delta y_c \delta y_n} + \frac{\mu_d}{\delta y_c \delta y_s} \right) u_c \right). \tag{19}$$

Using a staircase geometry makes the discretisation first order, but in the convection-dominated simulations studied in this paper the diffusive term is not really important. Eq. (19) reveals that the discretisation results in a symmetric matrix that is negative definite, which is also a property of the analytic diffusion operator [43].

Besides the physical kinematic viscosity μ/ρ, an amount of numerical viscosity is added in the diffusive term (μ^k/ρ). This numerical viscosity originates from the upwind discretisation of the convective term and is calculated every time step for every computational cell. E.g. when the horizontal velocity is positive, the numerical viscosity for the eastern velocity in Fig. 4 is equal to

$$\frac{\mu_r^k}{\rho} = \frac{1}{2} \frac{1}{F_c^b} \frac{m_r}{\delta y} \delta x_e = \frac{1}{4} \frac{1}{F_c^b} (A_e^x u_e + A_c^x u_c + (A_c^x - A_e^x) u_r^b) \delta x_e. \tag{20}$$

The pressure term in the x-momentum equation is discretised as a boundary integral resulting in

$$\oint_{\partial V} p n_x \, dS \doteq (p_e - p_w) A_c^x \delta y. \tag{21}$$

Here, p_e and p_w are the pressure in the eastern and western cells, respectively, (see Fig. 3), A_c^x is the edge aperture of the cell face where the central velocity is defined. Using this discretisation, the discrete gradient operator in the pressure term is the negative transpose of the discrete divergence operator Eq. (9), which is also an analytic property ($\nabla = -(\nabla \cdot)^T$) [43].

The external force F_g only consists of gravity in this study, so only a contribution of the external force is present in the momentum equation in z-direction (in three dimensions). The gravity term is discretised similar to the pressure term, wherefore it will be written as a boundary integral

$$\int_V F_g \, dV = \int_V \nabla g z \, dV = \oint_{\partial V} g z \boldsymbol{n} \, dS. \tag{22}$$

Evaluating this boundary integral in the xz-plane in the same way as the pressure term in the z-momentum equation results in

$$\oint_{\partial V} g z n_z \, dS = (g z_n - g z_s) A_c^z \delta x = g \delta z_c A_c^z \delta x, \tag{23}$$

where z_n and z_s are the coordinates of the centres of the cells north and south of the location where the z-momentum equation is discretised.

More details about the discretisations described in this section are given in [10,12].

3.3. Temporal discretisation

The equations of motion are discretised in time using the forward Euler method. This first order method is accurate enough, because the order of the overall accuracy is already determined by the first order accuracy of the free-surface displacement algorithm (see Section 3.5). Using superscript n for the time level, the temporal discretisation results in:

$$M^0 \boldsymbol{u}_h^{n+1} = -M^b \boldsymbol{u}_b^{n+1}, \tag{24}$$

$$\Omega \frac{\boldsymbol{u}_h^{n+1} - \boldsymbol{u}_h^n}{\delta t} = -C(\boldsymbol{u}_h^n, \boldsymbol{u}_b) \boldsymbol{u}_h^n - \frac{1}{\rho} \left((M^0)^{\mathrm{T}} p_h^{n+1} - \mu D \boldsymbol{u}_h^n \right) + \boldsymbol{F}_h^n. \tag{25}$$

The continuity equation is discretised at a new time level to ensure a divergence free velocity field. The spatial discretisation is written in matrix notation, where M is the divergence operator with M^0 working on the interior velocities and M^b on the boundary velocities, Ω contains cell volumes, C contains the convective coefficients (which depend on the velocity vector) and D contains diffusive coefficients.

3.4. Solution method

To solve the system of equations, the terms in the momentum equation are rearranged to

$$\boldsymbol{u}_h^{n+1} = \tilde{\boldsymbol{u}}_h^n + \delta t \Omega^{-1} \frac{1}{\rho} (M^0)^{\mathrm{T}} p_h^{n+1}, \tag{26}$$

where

$$\tilde{\boldsymbol{u}}_h^n = \boldsymbol{u}_h^n - \delta t \Omega^{-1} (C(\boldsymbol{u}_h^n) \boldsymbol{u}_h^n - \frac{\mu}{\rho} D \boldsymbol{u}_h^n - \boldsymbol{F}_h^n). \tag{27}$$

First an auxiliary vector field $\tilde{\boldsymbol{u}}_h^n$ is calculated from Eq. (27). Next Eq. (26) is substituted in Eq. (24), which results in a Poisson equation for the pressure [22]

$$M^0 \Omega^{-1} (M^0)^{\mathrm{T}} p_h^{n+1} = \frac{\rho}{\delta t} (M^0 \tilde{\boldsymbol{u}}_h^n + M^b \boldsymbol{u}_b^{n+1}). \tag{28}$$

From this equation, which does not require boundary conditions, the pressure is solved using the SOR (Successive Over Relaxation) method where the optimal relaxation parameter is determined during the iterations [2]. Once the pressure field is known, the new velocity field is calculated from Eq. (26) using the pressure gradient.

3.5. Free-surface displacement

After the new velocity field has been calculated, the free surface can be displaced. This is done using an adapted version of the VOF method first introduced in [23]. A piecewise constant reconstruction of the free surface is used, where the free surface is displaced by changing the VOF value in a cell using calculated fluxes through cell faces. The flux through a cell face is calculated as the velocity times the area of the cell face A and the time step δt

$$\delta F^s = \boldsymbol{u} \cdot \boldsymbol{n} A \delta t. \tag{29}$$

After all fluxes have been calculated, the VOF function is updated from time level n to $n+1$ using

$$(F^s)^{n+1} = (F^s)^n + \frac{\delta F_e^s + \delta F_n^s - \delta F_w^s - \delta F_s^s}{\delta x \delta y}, \tag{30}$$

where δF_e^s, δF_n^s, δF_w^s and δF_s^s are the fluxes through the eastern, northern, western and southern cell faces, respectively. Away from the free surface, this leads to a net flux of zero. In the neighbourhood of the free surface, the calculation of the fluxes is somewhat more complicated. The procedure explained by Hirt and Nichols in [23] is used to calculate these fluxes.

The original VOF method has two main drawbacks. The first is that flotsam and jetsam can appear [21,35], which are small droplets disconnecting from the free surface. The other drawback is the gain or loss of water due to rounding the VOF function when $F^s > 1$ or $F^s < 0$. By combining the VOF method with a local height function [12], these problems do not appear any more. The local height function is adopted in the following way. For every surface cell locally a function is defined that gives the height of the fluid in a column of three cells; see Fig. 5 and also Fig. 10, where the discretisation of the curvature is explained. The direction in which the function is defined is the direction of the coordinate axis that is most normal to the free surface (which is the positive z-direction in Fig. 5). Then, after calculating the fluxes across the cell boundaries of all three cells (the dashed-line region in Fig. 5) as in classical VOF, not the individual VOF values of the three column cells are updated, but the height function is updated. The individual VOF values of the three cells are then calculated from the height of the fluid in the column. When using this adapted fluid-displacement algorithm, the method is strictly mass conserving (because of the CFL-condition no overflow or underflow of the column can occur) and almost no flotsam and jetsam appear.

To assess the performance of the displacement method, the standard advection test of rotation of a slotted disk, defined by Zalesak [46], has been performed. This test has also been used by Rudman [36] and
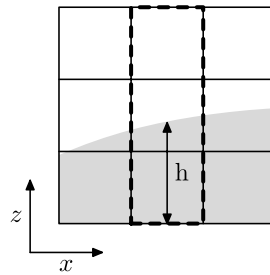

Fig. 5. The VOF function in cells near surface cells is updated using the local height function.
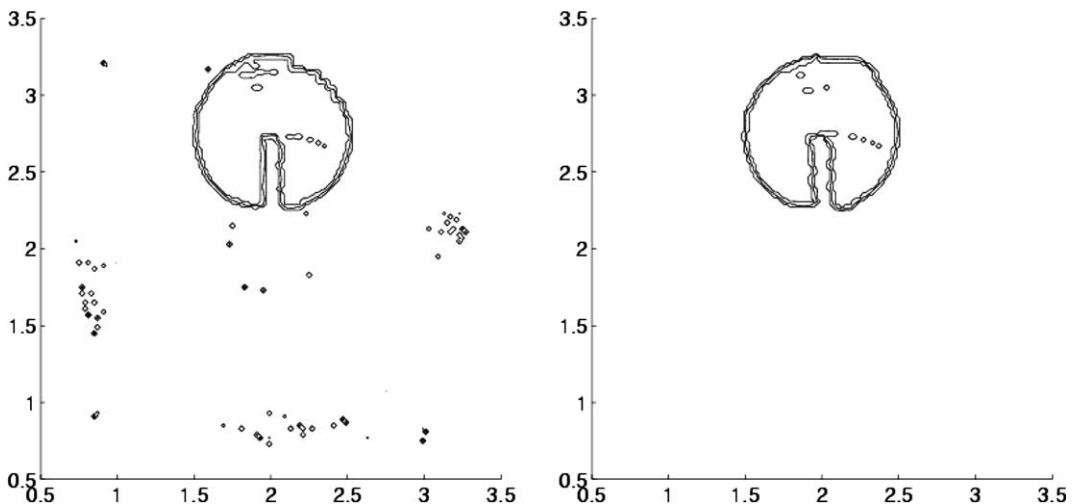

Fig. 6. Rotation of a slotted disk with original VOF (left) and VOF combined with a local height function (right).

Harvie and Fletcher [21] to compare different VOF methods. The domain is 4×4 m, the diameter of the slotted disk is 1 m, the width of the slot is 12 cm. The axis of rotation is put at (2,2), whereas the centre of the disk is at (2,2.75). The velocity field has a magnitude of 1 m/s at the middle of the domain edges. The mesh size is 200×200 and the CFL-number is 0.25. In Fig. 6 the resulting free surface after one rotation is shown with on the left standard VOF of Hirt–Nichols and on the right the current method. Three contour levels of the VOF function of 0.025, 0.5 and 0.975 are displayed. The standard VOF method generates much flotsam and jetsam, whereas in the adapted VOF method only some small holes in the fluid (of contour level 0.975) are present. Mass conservation is perfectly satisfied in the adapted method, whereas the fluid level in the standard VOF method has increased with 2%. In Fig. 7, a convergence study has been performed with grids of 100×100, 200×200 and 400×400 grid cells. The initial condition has also been plot. Only the contour level of 0.5 is displayed. The results show good convergence towards the exact solution.

Both methods, the original VOF method and the VOF method combined with a local height function, have been used in a dambreak simulation. In the left of Fig. 8 the result is shown of the free surface configuration of a dambreak calculation where original VOF is used. The originating flotsam and jetsam, small
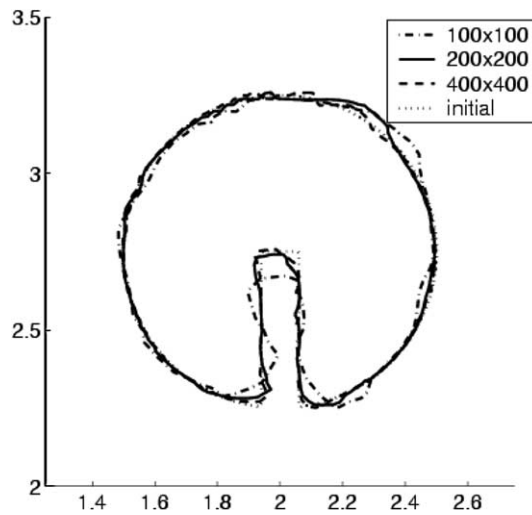

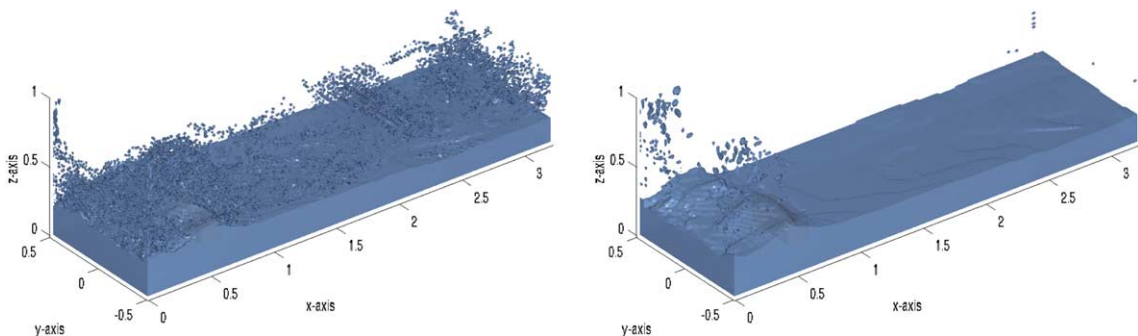Fig. 7. Grid refinement of the rotation of a slotted disk.

Fig. 8. Snapshots at the end of dambreak flow simulations with different algorithms for the displacement of the free surface: original VOF (left) and VOF combined with a local height function (right).

droplets disconnecting from the free surface, are very well visible. When combining VOF with a local height function, the amount of flotsam and jetsam has decreased considerably as can be seen in the right of Fig. 8. In Fig. 9, the change of the total water volume during the dambreak simulations is shown. The loss of water using standard VOF is considerable, about 7% after 6 s. In the adapted VOF method, the loss of water is only 0.02%, so mass is perfectly conserved.

3.6. Free-surface boundary conditions

3.6.1. Pressure at the free surface

A boundary condition for the pressure is needed in surface cells. The pressure in surface cells can be calculated from interpolation between the pressure at the free surface and the pressure in an adjacent fluid cell. The boundary condition that defines the pressure at the free surface is given by Eq. (5), which describes the continuity of normal stresses. The term containing the viscosity is neglected, which leaves

$$p = p_0 - \sigma \kappa. \tag{31}$$

Although surface tension is not the driving force in the simulations studied in this paper, it cannot always be neglected. To calculate the contribution of the surface tension in the pressure at the free surface, the total curvature of the free surface has to be determined in every S-cell.

When the free surface is given by a level-set function $s(x,y,t) = 0$, the mean total curvature is given by $\kappa = \nabla \cdot \mathbf{n}$, where $\mathbf{n} = \nabla s / |\nabla s|$ is the normal at the free surface. The local height function earlier introduced is also used to calculate the mean total curvature. The local height function is defined based on the orientation of the free surface. If the orientation of the free surface is more vertical than horizontal as in Fig. 10, the local height function is defined parallel with the x-axis. When it is defined by $h(y,t) = x$, in terms of the level-set function this would correspond to $s(x,y,t) = x - h(y,t)$. The values of the local height function are
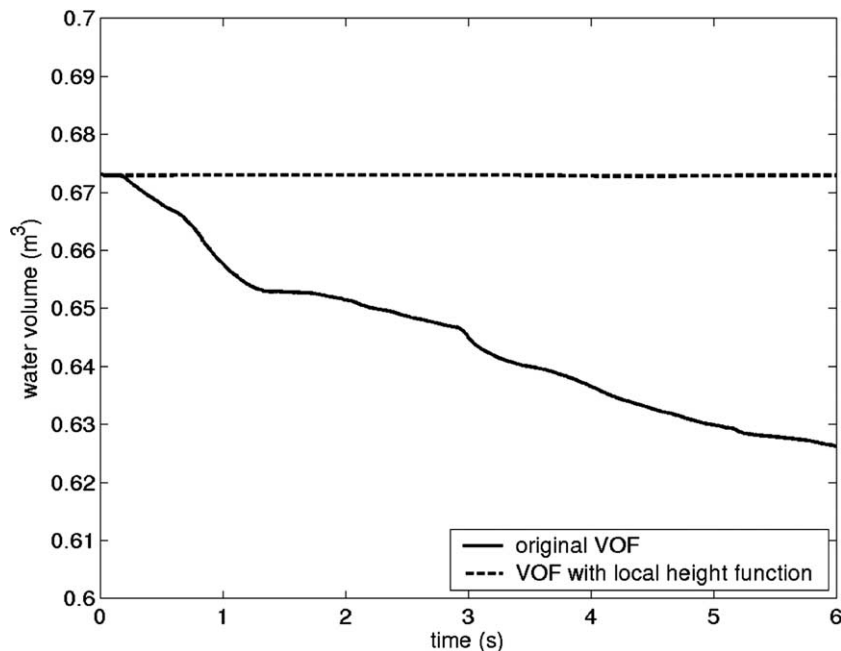

Fig. 9. Change in water volume during a dambreak simulation using original VOF and VOF combined with a local height function.
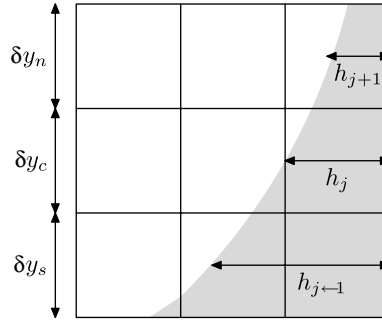
Fig. 10. Local height function to calculate the total curvature κ.

calculated using VOF fractions of three adjacent computational cells (see Fig. 10 for a visual explanation). In this case the mean curvature of the free surface is given by

$$\kappa = \frac{\partial}{\partial y}\left(\frac{\partial h/\partial y}{\sqrt{1+(\partial h/\partial y)^2}}\right). \tag{32}$$

The derivative of the local height function needed at positions $y_{j+\frac{1}{2}}$ and $y_{j-\frac{1}{2}}$ are discretised using standard finite differences. Then the discrete mean curvature in the central S-cell of Fig. 10 is given by

$$\kappa = \frac{1}{\delta y_c}\left(\frac{h'_n}{\sqrt{1+h'^2_n}} - \frac{h'_s}{\sqrt{1+h'^2_s}}\right) \tag{33}$$

with

$$h'_n = \frac{h_{j+1}-h_j}{\frac{1}{2}(\delta y_n + \delta y_c)} \quad \text{and} \quad h'_s = \frac{h_j - h_{j-1}}{\frac{1}{2}(\delta y_c + \delta y_s)}. \tag{34}$$

In three dimensions, the procedure for the calculation of the mean curvature follows a similar approach, but now using a cube of nine columns. Details can be found in [12].

At the intersection of the free surface and the solid body a boundary condition is needed for computing the mean curvature. This condition is given by a static contact angle, which is the angle between the normal of the free surface and the normal of the solid body. In the current calculations it is set at 90°.

Once the mean total curvature has been calculated, the pressure at the free surface is determined using Eq. (31). The pressure in surface cells is now calculated from the pressure at the free surface and the pressure in an adjacent fluid cell as in the method of Hirt and Nichols [23]. The fluid cell that is used for the interpolation is chosen based on the orientation of the free surface. If, for example, the orientation of the free surface is mainly horizontal (as in Fig. 11) with the fluid below the free surface, the fluid cell below the surface cell is used for the interpolation. The pressure in the centre of the surface cell is calculated by linear interpolation as

$$p_S = \eta p_{fs} + (1-\eta)p_F, \tag{35}$$

using the notation in Fig. 11 and $\eta = h/d$. If no fluid cell is found as neighbour of the surface cell, the pressure in the surface cell is set equal to the atmospheric pressure, corrected with a hydrostatic pressure contribution based on the local height of the fluid.
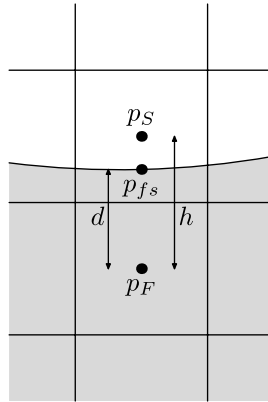
Fig. 11. Pressure interpolation in surface cells.

3.6.2. Velocities at the free surface

Velocities in the neighbourhood of the free surface can be grouped in different classes (see Fig. 12). The first class contains the velocities between two F-cells, between two S-cells and between an S- and F-cell. These velocities are determined by solving the momentum equation, so they are called momentum velocities. The second class consists of the velocities between an S- and an E-cell. These velocities are determined using boundary conditions, which will be described below. The last class consists of velocities between two E-cells that are sometimes needed to solve the momentum equation. These are determined using the tangential free-surface condition, Eq. (6), as described in [12].

3.6.3. SE-velocities

The choice for the method to determine the velocities at the cell faces between surface and empty cells (SE-velocities) turns out very important for the robustness and the accuracy of the flow model. Furthermore, it influences the occurrence of numerical spikes in the pressure signal (see Fig. 20, where numerical spikes can be seen): when a surface (or empty) cell of which the divergence is not zero changes to a fluid cell, the pressure responds with a spike to restore $\nabla \cdot \boldsymbol{u} = 0$.

Two methods can be distinguished:

- Method 1: the divergence of every S-cell is set to zero [20]. In the case that only one SE-velocity is present in the S-cell, this velocity is uniquely defined. In the case that more E-cells surround the S-cell, the net mass flux through FS-boundaries is divided over the SE-boundaries such that $\nabla \cdot \boldsymbol{u} = 0$ is satisfied.
- Method 2: the SE-velocities are determined by extrapolating interior velocities. The velocities used for the extrapolation are taken from the direction of the main body of the fluid. In this study both constant and linear extrapolation are used with their specific (dis)advantages.

Fig. 12. Different classes of velocities near the free surface.

Table 1
Summary of the performance of the two methods for SE-velocities

	Accuracy wave	Robustness	Smooth p
$\nabla \cdot \boldsymbol{u} = 0$	−	−	+
Extrapolation	+	Linear −	−
		Constant +	

Below, the performance of the methods with respect to accuracy in wave simulations, robustness and smoothness of the pressure field is explained. This is also summarised in Table 1, where a '+' sign indicates good performance and a '−' sign less satisfactory performance.

The first column of Table 1 indicates that Method 1 gives less accurate results in wave simulations than Method 2. This difference can be seen in Fig. 13, where a snapshot of the wave elevation in a steep wave event is shown. Method 2, indicated by the solid line, gives a good prediction of the wave elevation compared to the measurements. The predicted elevation using Method 1 is worse. The explanation of the results is that when using extrapolation from the interior velocity field the velocities are much better estimated than when using the criterion $\nabla \cdot \boldsymbol{u} = 0$.

In the second column of Table 1 the robustness of the methods is indicated. The low robustness of the '$\nabla \cdot u = 0$' method originates from instabilities in cut cells. E.g. in the situation as sketched in Fig. 14 the SE-velocity is calculated using

$$u_{\text{SE}} = \frac{1}{A_{\text{SE}}^x} (A_{\text{SS}}^z w_{\text{SS}} - A_{\text{FS}}^x u_{\text{FS}} - 0) \tag{36}$$

to satisfy $\nabla \cdot \boldsymbol{u} = 0$. Because of the large ratio between the SE-aperture and the SS-apertures, the SE-velocity will become very large. If this configuration is stationary during a number of consecutive time steps, the
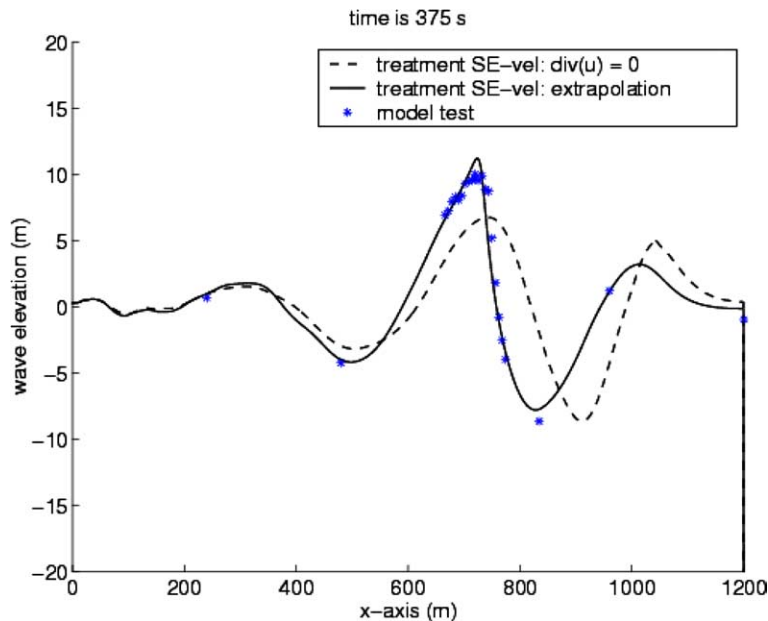

Fig. 13. Different methods for free-surface velocities in a steep wave simulation: $\nabla \cdot \boldsymbol{u} = 0$ and linear extrapolation.
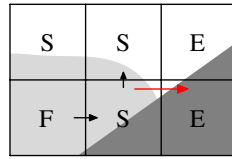
Fig. 14. Very large SE-velocity when using Method 1.

time integration will become unstable. Also the linear extrapolation method can lead to instabilities when the velocity field is not smooth. When using constant extrapolation, the method is much more robust.

The smoothness of the pressure field in the two methods is described in the third column of Table 1. As mentioned before, pressure spikes can occur when a surface cell with $\nabla \cdot \boldsymbol{u} \neq 0$ changes to a fluid cell. By construction, these spikes do not appear in the first method, where always $\nabla \cdot \boldsymbol{u} = 0$ in surface cells. But using constant or linear extrapolation for the SE-velocities can result in spikes, because then the continuity equation is not automatically satisfied in surface cells. When such a surface cell changes to a fluid cell, the pressure responds with a spike.

To overcome the problems described above, in practice a method should be chosen that is an engineering mix between the two methods. The extrapolation method with constant extrapolation performs best and has been chosen in our numerical method. But to prevent spikes in the pressure signal, $\nabla \cdot \boldsymbol{u} = 0$ is enforced during the time step when a cell changes label from surface cell to fluid cell. The constant extrapolation is changed to linear extrapolation when greater accuracy in wave simulations is needed. This combination results in a highly accurate and very robust method.

4. Stability of the method

In the case of uncut cells with fixed objects the stability of the equation containing the time integration term and the convective term is given by the CFL-restriction, which in one dimension reads $\delta t |u|/h \leqslant 1$ (h is the size of the uncut cell). When cut cells are present, for the chosen convective discretisation this criterion is not changed. This result is not directly straightforward when looking at the equation containing the time derivative and the convective term

$$\frac{\partial \boldsymbol{u}}{\partial t} = -\Omega^{-1} C(\boldsymbol{u}, \boldsymbol{u}_{\mathrm{b}}) \boldsymbol{u}. \tag{37}$$

The matrix Ω is a diagonal matrix containing volumes of the cells, so these entries can become arbitrarily small for cut cells, hence the elements in Ω^{-1} can become arbitrarily large. To examine stability, the eigenvalues of the convective matrix C, generated by Eq. (12), have to be determined. Using Gerschgorin circles, these eigenvalues can be estimated as being of order $O(\Omega u/h)$. The Ω in this estimation cancels the contribution of Ω^{-1} in $\Omega^{-1} C$, leaving the stability criterion for cut cells the same as for uncut cells [10].

When moving objects are present, the story becomes somewhat different. A distinction can be made between the object moving normal to its boundary and tangential to its boundary (as shown in the left and right of Fig. 15, respectively). When the object is moving tangential to its boundary, the eigenvalues of the matrix $C(\boldsymbol{u}, \boldsymbol{u}_{\mathrm{b}})$ can again be estimated by $O(\Omega u/h)$, which means that stability is guaranteed when the CFL-restriction is used. But when the object is moving normal to itself, the eigenvalues of $C(\boldsymbol{u}, \boldsymbol{u}_{\mathrm{b}})$ are of order $O(h\,u_{\mathrm{b}})$. Now stability is not guaranteed anymore, with eigenvalues of $\Omega^{-1} C(\boldsymbol{u}, \boldsymbol{u}_{\mathrm{b}})$ in Eq. (37) of order $O(\Omega^{-1} h\,u_{\mathrm{b}})$. They can become arbitrary large due to the factor Ω^{-1}.
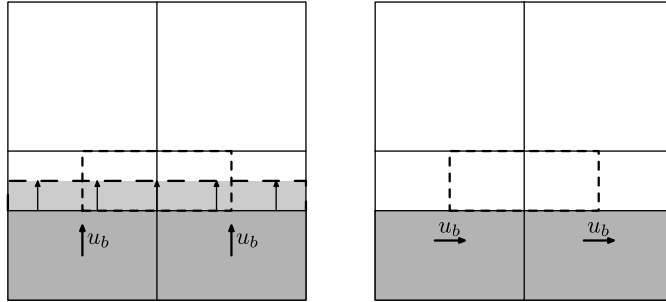
Fig. 15. (Left) Boundary moving normal to itself: maximum stabilisation is required; (right) boundary moving tangential to itself: no stabilisation is required.

To cancel the effect of Ω^{-1}, a formulation based on a weighted average of the fluid velocity and the boundary velocity is applied in the cells cut by the moving object. To avoid smearing of the interface in cases where it is not necessary to stabilise the convective term (namely when the object is moving tangentially to its boundary), the following discretisation is used:

$$\boldsymbol{u}^{n+1} = \Lambda(\boldsymbol{u}^n - \delta t(\Omega^{n+1})^{-1}C^n\boldsymbol{u}^n) + (I - \Lambda)\boldsymbol{u}_{\mathrm{b}}^{n+1} \tag{38}$$

with weight factor $\Lambda = \Omega^{n+1}(\Omega^{n+1} + |\Delta\Omega|)^{-1}$, where $\Delta\Omega = \Omega^{n+1} - \Omega^n$ is the difference between cell volumes at two different time steps. The weight factor Λ has been chosen such that the stabilising term is only used when the body is moving; note that it equals unity for fixed objects. Furthermore, maximum stabilisation is established when the object is moving normal to its boundary, whereas no stabilisation is used when the object is moving tangential to its boundary (as $\Delta\Omega = 0$ then).

Also from the diffusive term, a stability criterion follows with a restriction on the time step. In the case of uncut cells, this criterion is given by $\delta t \leqslant \rho h^2/2\mu$. The diffusive term is discretised as if all cells were uncut ('staircase' approach), so the above criterion is also valid in the cut-cell model.

5. Convergence of the method

To check the numerical convergence of the method, which is first order in space and time, simulations have been performed of a regular wave. The wave has a period of 14.44 s, the wave height is 10.14 m and the water depth is 600 m. The domain consists of four wavelengths and is cut off at a depth of 300 m. Half of the domain is used as damping zone to minimise wave reflections. The waves are generated using 5th order Stokes theory. The error in the wave elevation after four wave periods is shown in Fig. 16 for a convergence study in time (left) and space (right). The error $E(x)$ is calculated by

$$E(x) = \frac{|\eta(x) - \eta_{\mathrm{th}}(x)|}{H} \tag{39}$$

with $\eta(x)$ and $\eta_{\mathrm{th}}(x)$ the calculated and theoretical wave elevation, respectively, and H the wave height. For grid refinement, three different grids have been used with 120×25, 240×50 and 480×100 grid points. The number of time steps has been varied for a convergence study in time, more specifically, 63, 125 and 250 time steps per period have been used. The upper part of Fig. 16 shows the wave elevation at the same moment in time as when the error is calculated. From the plots convergence with increasing resolution is clear, especially in the spatial grid refinement, but not a very clear conclusion can be drawn about the order of
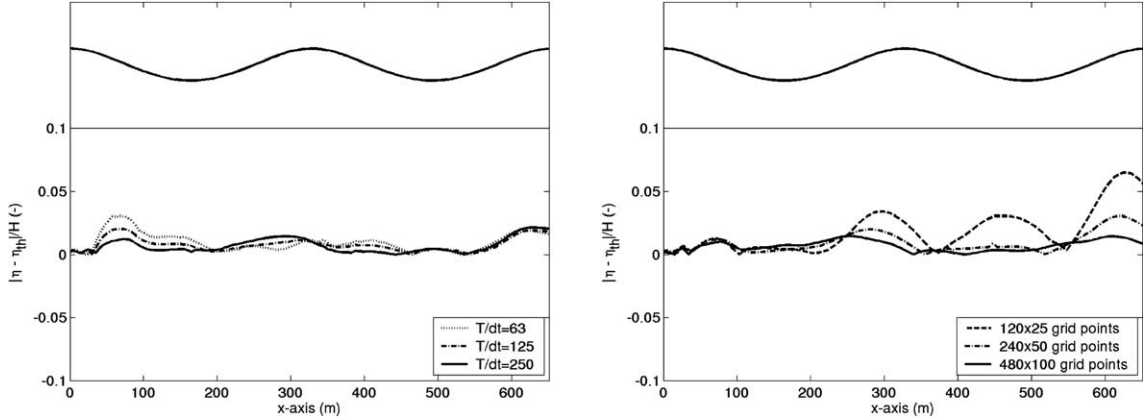
Fig. 16. Convergence of the method in time (left) and space (right).

convergence. This is partly due to wave reflections from the damping zone, which still have an influence on the accuracy of the simulations.

6. Interactive body–fluid motion

In the current method the objects are either moving with a prescribed motion, or the motion is calculated from the interaction between the object and the liquid dynamics. Simulations with prescribed moving objects are only an approximation of reality, since the motion of the body usually is unknown, as it is affected by the forces the fluid exerts on the body. In this section a full coupling between the liquid motion and the body motion is described. The acceleration of the body can be computed by Newton's second law

$$A_{\text{body}} = I^{-1}F_{\text{body}}, \tag{40}$$

where I is the inertia matrix and F_{body} contains the forces and moments exerted by the fluid.

An explicit integration of Eq. (40), where the body motion is following from the forces exerted by the liquid $(A_{\text{body}}^{n+1} = I^{-1}F_{\text{body}}^{n})$, is not guaranteed to be stable as we will show. When assuming a simple hydrodynamic model from added mass potential theory (one-dimensional, without rotations, hence $I = m$ with m the mass of the body), the force on the body can be calculated as $F_{\text{body}} = -m_{\text{a}}A$, with m_{a} being the added mass of the body. Inserting this into the explicit integration gives

$$A^{n+1} = -\frac{m_{\text{a}}}{m}A^{n}. \tag{41}$$

This system is only stable when $m_{\text{a}}/m \leqslant 1$. In other words, it is required that the mass of the moving body is larger than its added mass. For floating bodies this will not always be true, implying that this explicit integration cannot be used in that case.

To ensure stability in all cases, per time step an iterative method is used to solve the problem fully implicit. Introducing sub-iteration index k and rewriting $A_{\text{body}}^{n+1} = (V_{\text{body}}^{n+1} - V_{\text{body}}^{n})/\delta t$ this leads to

$$(V_{\text{body}}^{n+1})^{k+1} = V_{\text{body}}^{n} + \delta t I^{-1}F_{\text{body}}^{k} \tag{42}$$

with V_{body} the body velocity. The force F_{body} is computed by an integration of the pressure over the boundary of the moving body (viscous forces are neglected in this study). Therefore, for every sub-iteration a new

pressure field has to be computed, which corresponds to the new velocity of the body by solving the Poisson equation for the pressure. To prevent divergence of the iteration process, a relaxation parameter ω is introduced

$$(V_{body}^{n+1})^{k+1} = (V_{body}^{n+1})^k + \omega((V_{body}^{n+1})^* - (V_{body}^{n+1})^k), \tag{43}$$

where $(V_{body}^{n+1})^*$ is given by Eq. (42). Substituting $(V_{body}^{n+1})^*$ and rewriting leads to

$$(V_{body}^{n+1})^{k+1} = \omega(V_{body}^n + \delta t I^{-1} F_{body}^k) + (1 - \omega)(V_{body}^{n+1})^k. \tag{44}$$

Here, ω depends on the stability of the system; typically it has to compensate the factor m_a/m in Eq. (41). For problems with moving bodies that are heavy, the coupling will be stable with just a bit under-relaxation resulting in faster convergence. About 1–3 iteration steps are needed in that case. However, for lighter bodies more under-relaxation is needed, resulting in slower convergence. Typically, the number of sub-iterations is between 1 and 20, depending on the body mass and the magnitude of the fluid forces acting on the body. This slows down the method, because in every sub-iteration step the Poisson equation for the pressure has to be solved. However, the method is stable and performs well.

7. Dambreak simulation

At the Maritime Research Institute Netherlands (MARIN) experiments have been performed for breaking dam flows. These experiments can be seen as a simple model of green water flow on the deck of a ship. The dambreak is a very popular validation case, because the set-up is easy: no special in- or outflow conditions are needed. A large tank of $3.22 \times 1 \times 1$ m is used with an open roof. The right part of the tank is first closed by a door. Behind the door 0.55 m of water is waiting to flow into the tank when the door is opened. This is done by releasing a weight, which almost instantaneously pulls the door up. In the tank a box has been placed that represents a scale model of a container on the deck of a ship.

During the experiment measurements have been performed of water heights, pressures and forces. In Fig. 17 the positions of the measured quantities are shown. Four vertical height probes have been used; one in the reservoir and the other three in the tank. The box was covered by eight pressure sensors, four on the front of the box and four on the top. The forces on the box were also measured.

As initial configuration of the simulation with COMFLOW, the water in the right part of the domain is at rest. When the simulation is started, due to gravity the water starts to flow into the empty part of the tank. A fine grid of $236 \times 76 \times 68$ grid cells has been used with some stretching towards the bottom of the tank. The simulation is continued for 6 s with an automatically adapted time step using maximum CFL-numbers around 0.75, resulting in a time step of the order of 0.001 s. In Fig. 18 two snapshots of the early stages of the simulation are shown together with some images of the video from the experiment (at the same instants of time). The smaller pictures inside the snapshots show the water in the reservoir. There is a very good
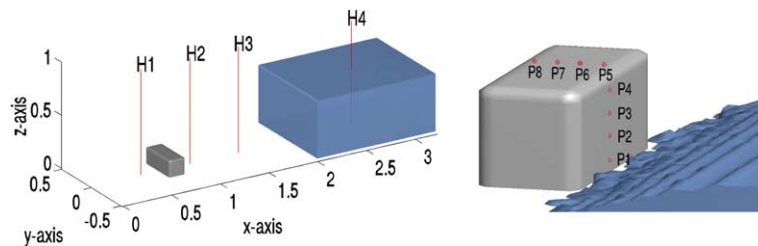

Fig. 17. Measurement positions for water heights and pressures in the dambreak experiment.
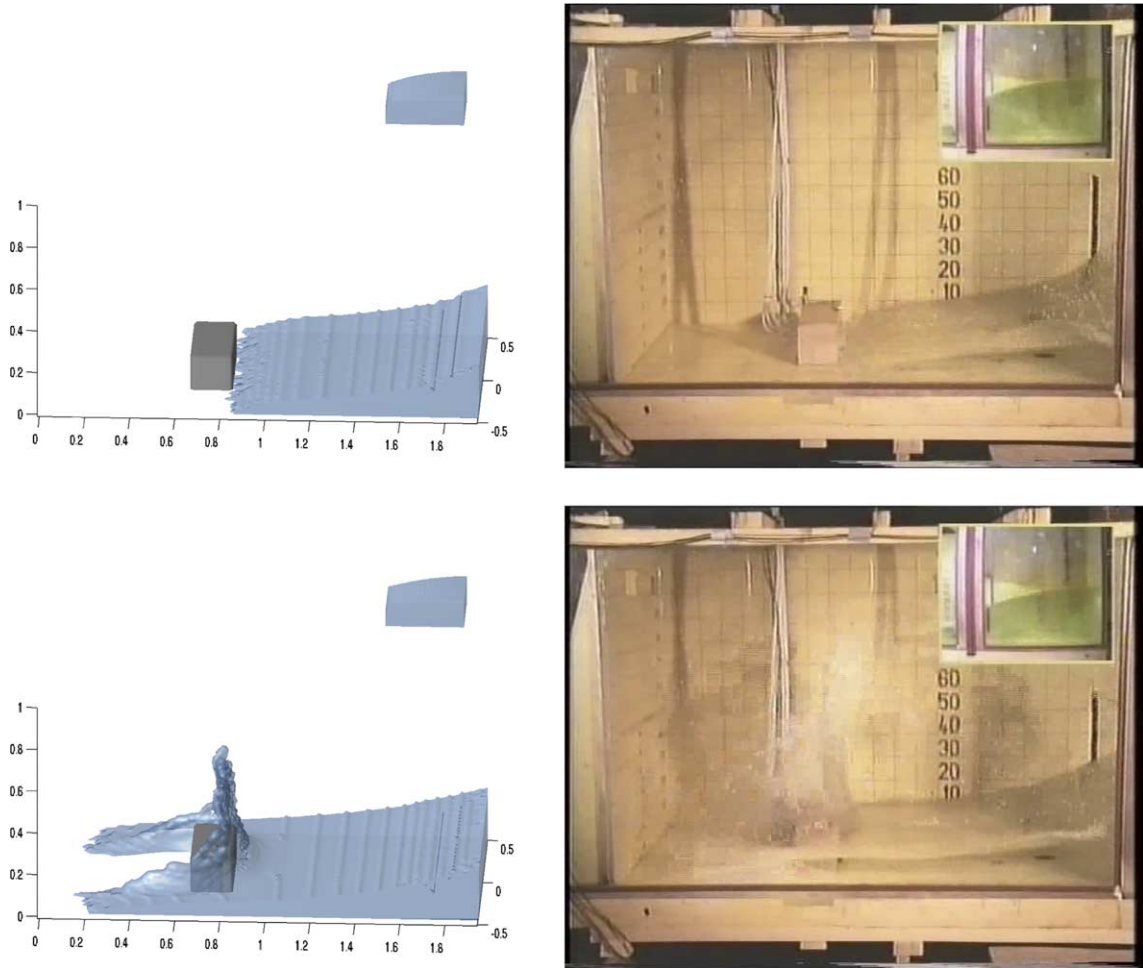
Fig. 18. Snapshots of a dambreak simulation with a box in the flow compared with experiment at time 0.4 and 0.56 s.

agreement between the snapshots of simulation and experiment. The time instant when the water is first hitting the box is the same. The shape of the free surface, bending a bit forwards in the second picture, is seen in both experiment and simulation. In the simulation the free surface has some ripples, which can be suppressed by using a piece-wise linear reconstruction of the free surface [44] instead of the reconstruction aligned with the coordinate axes used in this study.

In Fig. 19 time histories of the water height at two locations are shown: in the reservoir, and in the tank just in front of the box. The agreement in both pictures is very good until the water has returned from the back wall (after about 1.8 s). After that some differences occur, but the global behaviour is still the same. After the water has returned from the wall, the fluid height at probe H2 is the largest. The water flows back to the reservoir, where it turns over again after about 4 s. The moment that this second wave meets the height probe at H2 again (after about 5 s) is almost exactly the same in simulation and experiment.

The instant when the wave hits the box is perfectly captured by the simulation as can be seen from Fig. 20. Here the pressure at point P1 and P3 at the front of the box and at the top of the box, P5 and P7 (see Fig. 17), are shown. The magnitude of the impact pressure is the same for simulation and experiment at
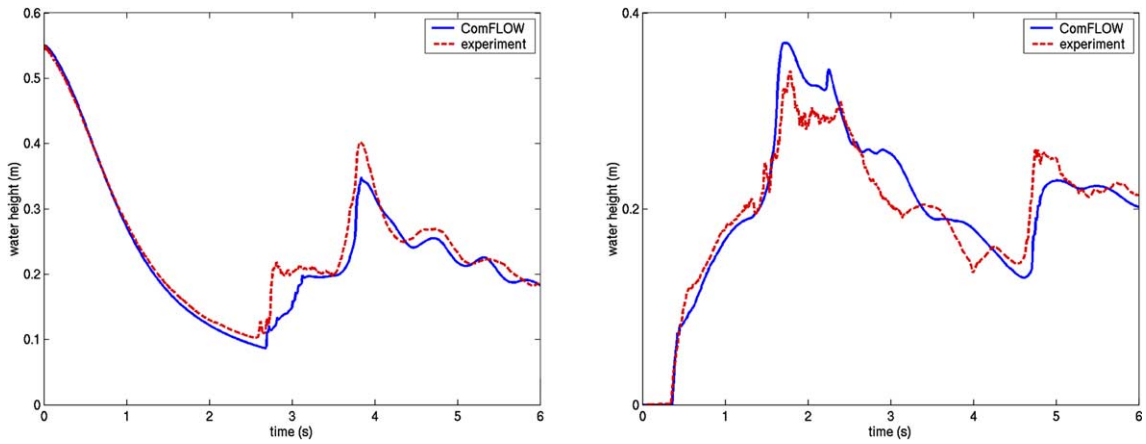
Fig. 19. Vertical water heights in the reservoir H4 (left) and the tank H2 (right).
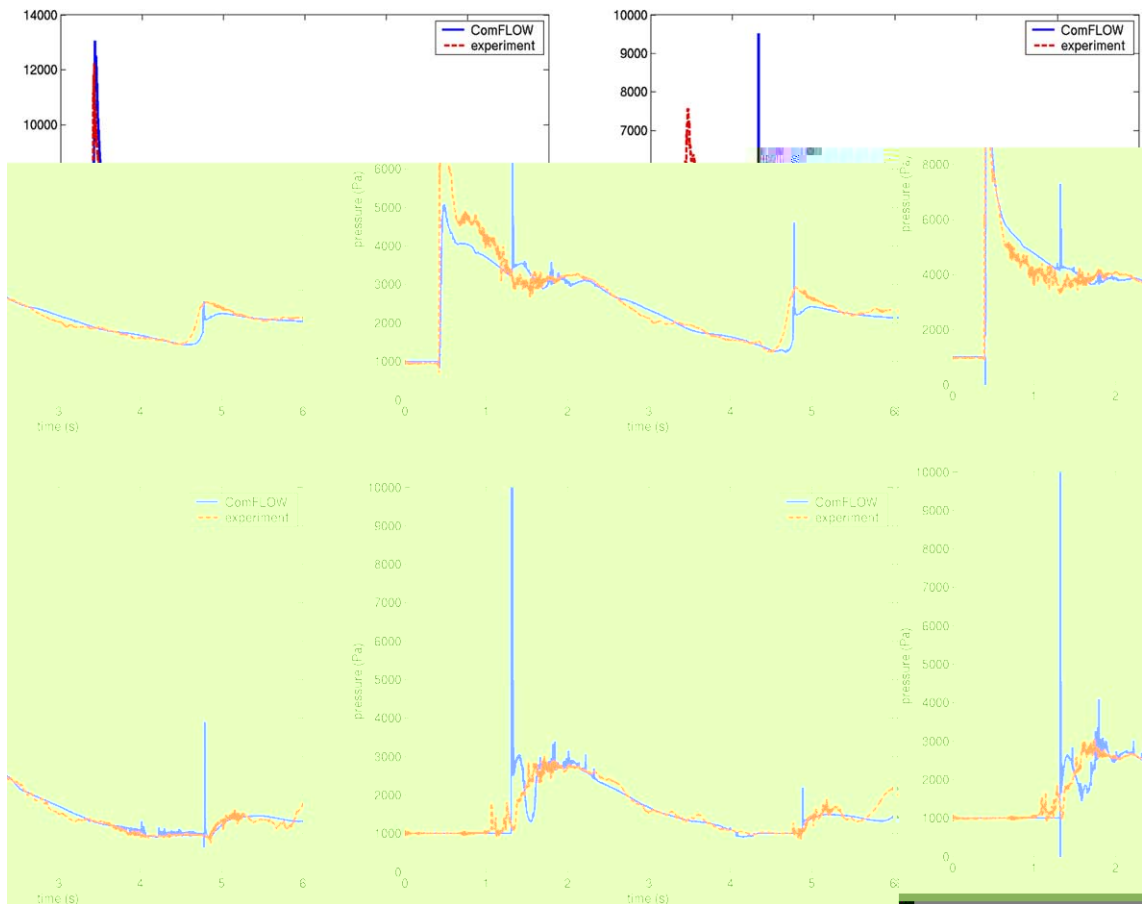

Fig. 20. Pressure time histories at P1 (picture upper left), P3 (upper right), P5 (lower left) and P7 (lower right).

pressure point P1 (the lowest on the box), but is under predicted by the simulation at point P3. The moment the return wave hits the box again (at about 4.7 s) is again visible in the graphs. In the bottom graphs of Fig. 20, where the time history of pressure transducers at the top of the box are shown, a clear difference occurs between simulation and experiment. After about 1.3 s there is a wiggle in the simulation with a duration of 0.5 s, which is not present in the experiment. Before this point the water hits the top of the box when the wave coming back from the wall is overturning. This is a real difference, which cannot be explained properly at the moment.

Several spikes appear in the pressure signals that are visible in all the graphs at the same moment (for example at 1.3 s). These spikes occur, because some water enters an empty cell that is completely surrounded by cells with fluid. When the water enters the E-cell, there is no empty cell left in the neighbourhood, so this cell changes to a fluid cell in one time step without being a surface cell in between. This discontinuous change in label and the corresponding restoration of $\nabla \cdot \mathbf{u} = 0$, results in a pressure peak over the whole pressure field.

In Fig. 21 a grid refinement study of the dambreak simulation is shown. Three different grids have been used with, in increasing order, $59 \times 19 \times 17$ grid points, $118 \times 38 \times 34$ grid points and $236 \times 76 \times 68$ grid points. The finest grid has also been used in the previous figures. In the figure the pressure along the lower part of the front side of the box is shown. The overall flow of the water is pretty much the same in all three grids, but when zooming in on the pressure peak (in the right of the figure) differences become visible. The coarsest grid is clearly not good enough. The pressure peak is over predicted and the water reaches the box too late. Although the water reaches the box earlier in the finer grids, there is still a small difference between simulation and experiment. The magnitude of the impact is better predicted on the finer grids.

The results of the dambreak simulation are in good agreement with the experiment. The global behaviour of the fluid is the same and the impact peak of the pressure agrees well, especially along the lower part of the box.

8. Water entry of wedge, cone and circular cylinder

In this section results are presented from water entry of two-dimensional wedges, circular cylinders and of a three-dimensional cone. The tests have been performed with prescribed constant entry velocities.
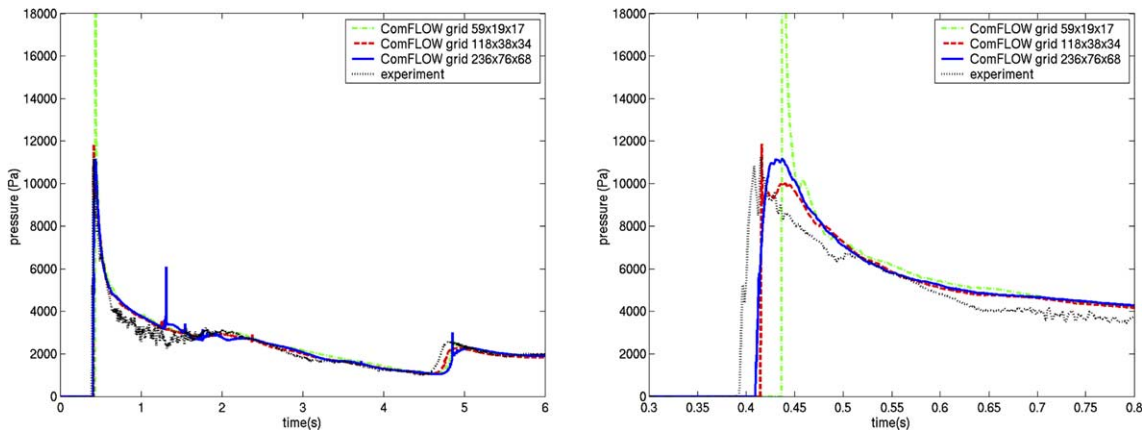

Fig. 21. Grid refinement in the dambreak simulation: pressure at the lower part of the box (left), zoomed picture (right).

8.1. Wedge entry

Fig. 22 presents free-surface profiles for the entry of two two-dimensional wedges. The wedges have dead-rise angles of 30° and 45°, respectively. The simulation results are compared with photographs of experiments by Greenhow and Lin [18]. The visual comparison between experiments and simulations is relatively good. However, not all the droplets and small details are resolved by the simulation, because of the displacement algorithm and the grid resolution. The simulations have been performed on a grid of 300×280 computational cells. Using such a fine grid, ComFLOW is able to resolve the jets at the side of the wedges. The angle under which the jets are formed and bend away is well predicted by ComFLOW. However, these jets do not have a large influence on the impact loads during the penetration [19]. The flow is not perfectly symmetric, because the numerical algorithm in ComFLOW is not symmetric (for example the marching direction in the iteration process of the Poisson equation has some influence on the symmetry).

In Fig. 23 the results of a grid refinement have been shown. Three different grids have been used: 75×70, 150×140 and 300×280 grid cells. The large difference between the results of the free surface profiles is in the formation of the jets: the finer the grid, the better the jets are resolved.
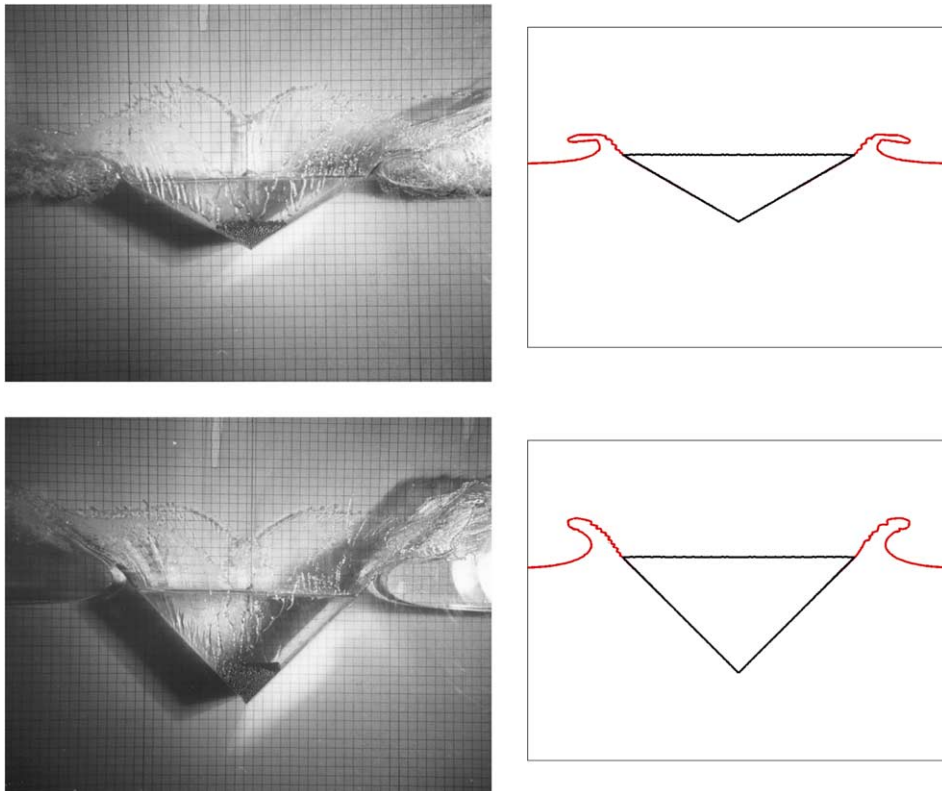

Fig. 22. Snapshots of wedge entry with dead-rise angles 30° (up) and 45° (down), experimental photographs of Greenhow and Lin [18].

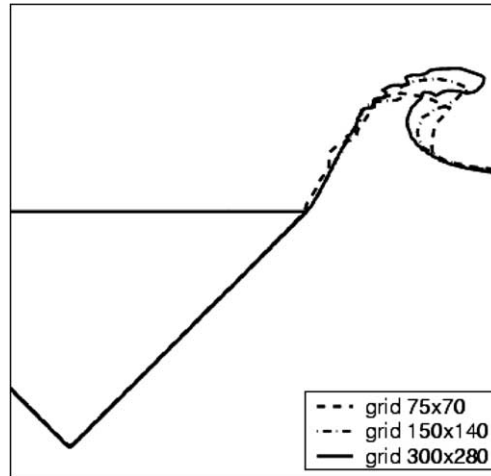Fig. 23. Free-surface profile of a falling wedge simulation with dead-rise angle 45° using three different grids.

8.2. Cone entry

As the three-dimensional equivalent of the wedge, the entry of a cone has also been studied. The cone has a dead-rise angle of $\beta = 30°$. The slamming coefficient, as a non-dimensional measure of the total hydrodynamic force, has been compared with the theoretical prediction by Schiffman and Spencer made in 1951 [38]

$$\frac{F}{\rho} = 3k(\beta)\tan^3\left(\frac{\pi}{2} - \beta\right)V^4 t^2, \tag{45}$$

where $t = 0$ is the moment the cone hits the free surface first. The non-dimensional parameter $k(\beta)$ is considered to give most accurate results at value 1.6 (see [1]). In the top of Fig. 24 the slamming coefficient as calculated by COMFLOW has been plotted against the penetration depth Vt. Three different entry velocities have been chosen, leading to the same results in a suitable set of scaled variables.

The results of the slamming coefficients for the impact during the entry of a cone are in very good agreement with the classical theory of Schiffman and Spencer [38]. In the bottom of Fig. 24 a cross-section is shown of the free-surface profile during the cone entry. From this it can be seen that the jets at the side of the cone, which are clearly present in the entry of a wedge (see Fig. 22), are not well resolved. This is caused by the difference in two- and three-dimensions and due to the much coarser grid used in the cross-section of the cone entry. However, it does not have a large influence on the total slamming force, because pressures in the jets are very small and do not contribute much.

8.3. Entry of a circular cylinder

The entry of a circular cylinder has also been studied. Snapshots of two different instants in time are shown in Fig. 25 and are compared with photographs of experiments by Greenhow and Lin [18]. The free-surface shape observed in the experiment is very well resolved by COMFLOW. Not all the details of the droplets of the splash are captured by the simulation, but the jets that appear at the sides of the cylinder are well predicted.

The total vertical hydrodynamic force on the cylinder during the first stage of the impact has been calculated and compared with experimental results of Campbell and Weynberg [4], also reported in [1]. In Fig. 26 the slamming coefficients of the cylinder entry with different entry velocities have been plotted
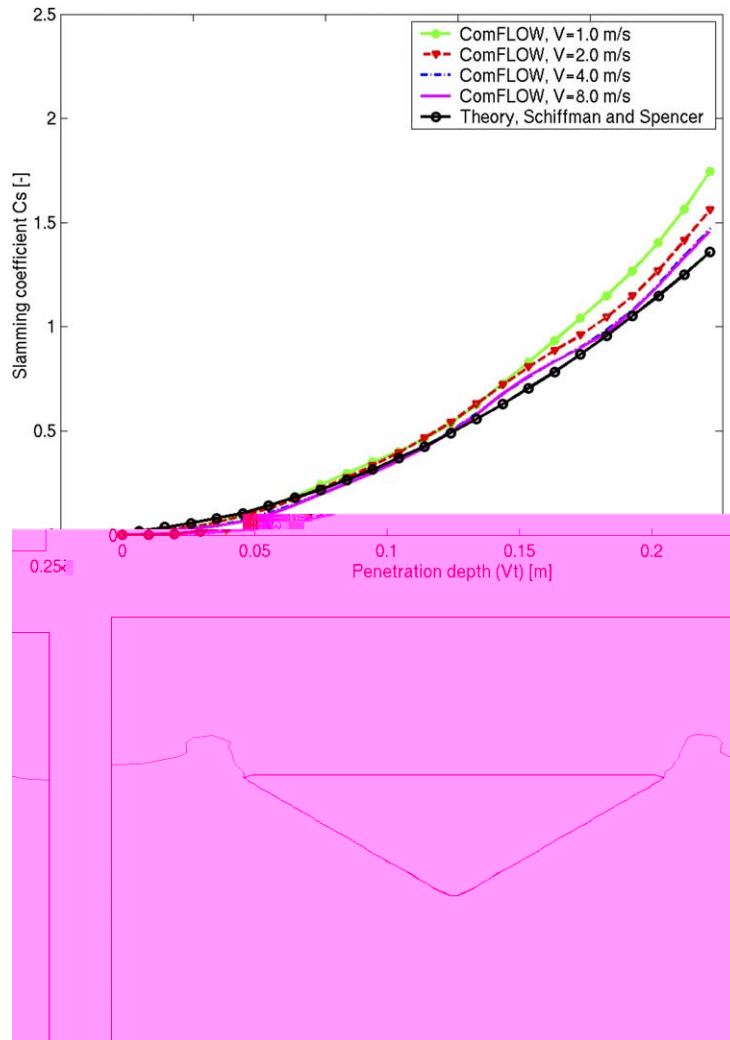
Fig. 24. (Top) slamming coefficient for the impact of a cone, ComFLOW compared with theory of Schiffman and Spencer [38]; (bottom) cross-section during the entry of a cone showing the free-surface profile.

versus the non-dimensional penetration depth. The slamming coefficient is given by $Cs = F/\rho R V^2$, with F the total vertical hydrodynamic force, R the radius of the circular cylinder and V the entry velocity. Besides the experimental result of Campbell and Weynberg, also the theory of Von Karman (1929), reported by Faltinsen in [9], has been included. This theory is based on potential flow theory. For the initial state of the entry of a circular cylinder, the hydrodynamic slamming force can be estimated by

$$F = V\rho\frac{\pi}{2}(2VR - 2V^2t),$$ (46)

where t denotes time with $t = 0$ the moment of first impact.

The comparison between the experiments of Campbell and Weynberg and the simulations is relatively good. It can be seen that the initial impact is a bit under predicted by ComFLOW for all entry velocities. The initial impact is more in agreement with the theory of Von Karman. In a later stage the results are in
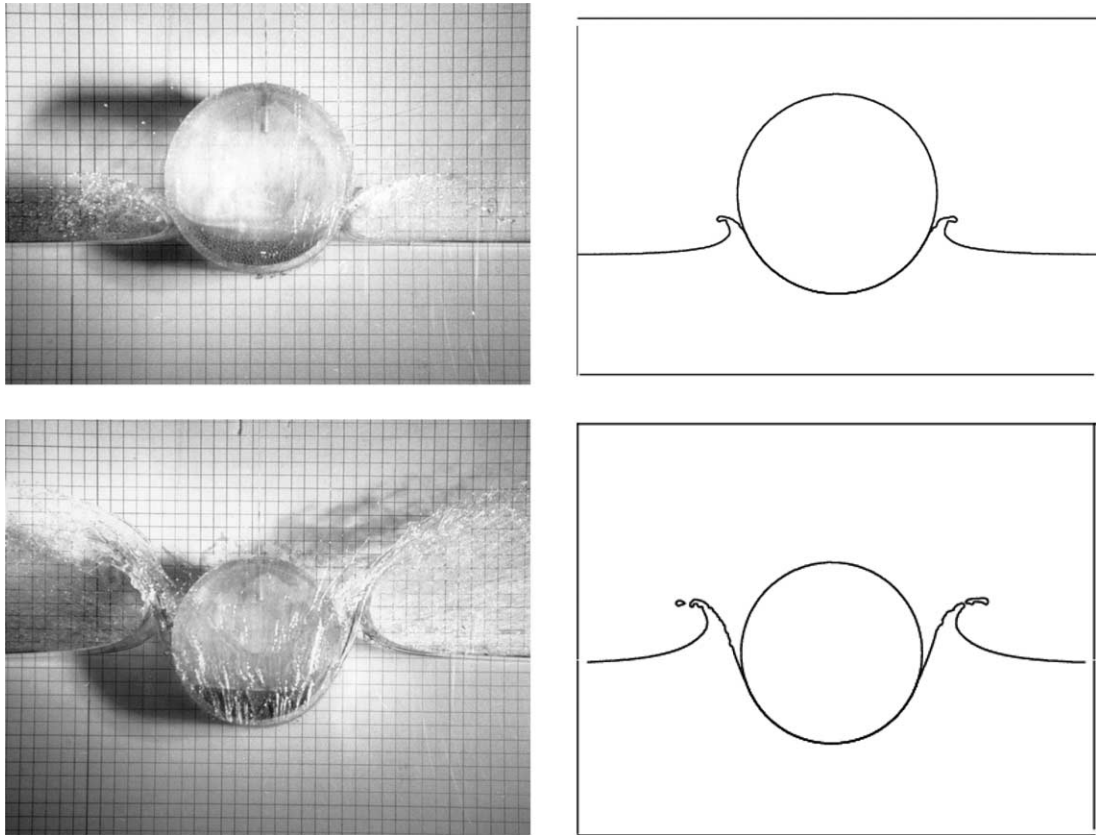
Fig. 25. Snapshots of cylinder entry, experimental photographs of Greenhow and Lin [18].

very good agreement with the experiments of Campbell and Weynberg. The results of CoMFLOW are similar for different entry velocities, confirming near-perfect scaling with V^2.

To investigate convergence of the method under grid refinement, the circular cylinder entry simulations have also been run with different grids. The results are presented in Fig. 27. It can be seen that the coarseness of the grid has a very large influence on the formation of the jets aside the cylinder. This was also concluded from the simulation of the cone entry. A very fine grid is needed to capture the jets. However, the formation of the jets does not have a large influence on the total hydrodynamic force. The force can be predicted quite accurately on the coarsest grid.

9. Free falling wedge

Finally, results are shown of a wedge falling freely into initially calm water. The velocity of the wedge is not prescribed, but follows from the interaction with the liquid dynamics. It is calculated every time step as explained in Section 6.

Zhao et al. [45] published some experimental results for a falling wedge. The wedge is 0.5 m wide and 0.29 m high with 30° dead-rise angle (see Fig. 28). The total length of the wedge is 1 m of which the measuring section is 0.2 m. The total weight of the wedge and the drop mechanism is 241 kg. During the exper-
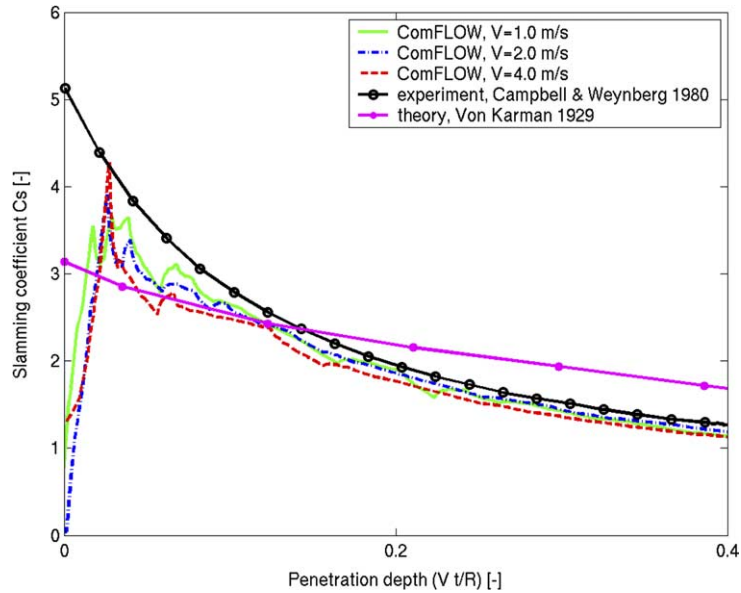
Fig. 26. Slamming coefficient of the entry of a circular cylinder compared to the experiments of Campbell and Weynberg and the theory of Von Karman.

iment the vertical drop velocity and the vertical force are measured. In Zhao et al., a fully non-linear solution is presented using an initial value problem solved by potential flow theory. The solution is compared to the experimental results. The measurements have also been used by Muzaferija et al. [32] to compare their method with. They use a finite volume method with polyhedral control volumes incorporating a free-surface capturing model. The motion of the object is prescribed using the measurements. In both numerical methods of Zhao and Muzaferija the vertical force was over predicted, which is due to two-dimensional effects. Zhao et al. successfully adapted their method to account for the three-dimensional effects. Muzaferija et al. presented some three-dimensional simulations, which also showed a decrease in the vertical force and better agreement with the experiment.
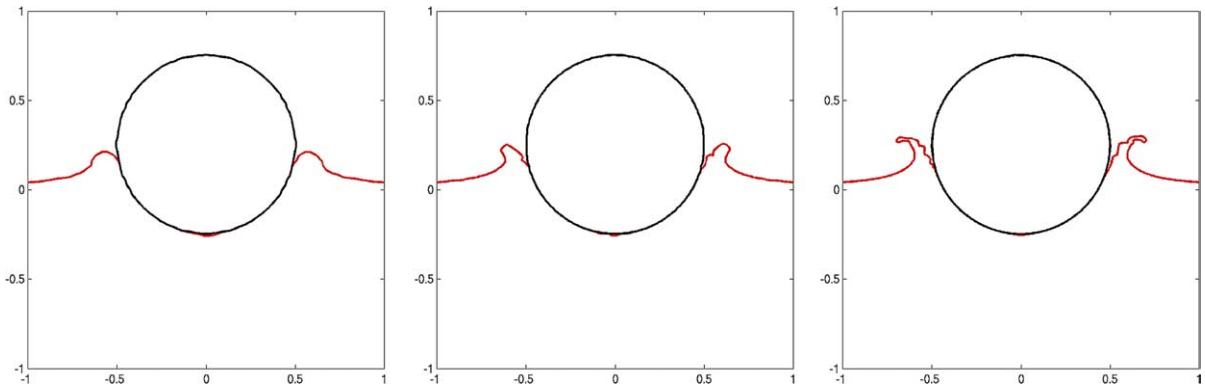

Fig. 27. Effect of grid refinement on the jets formed during water entry of a circular cylinder: from left to right 100×100, 200×200 and 400×400 grid cells.
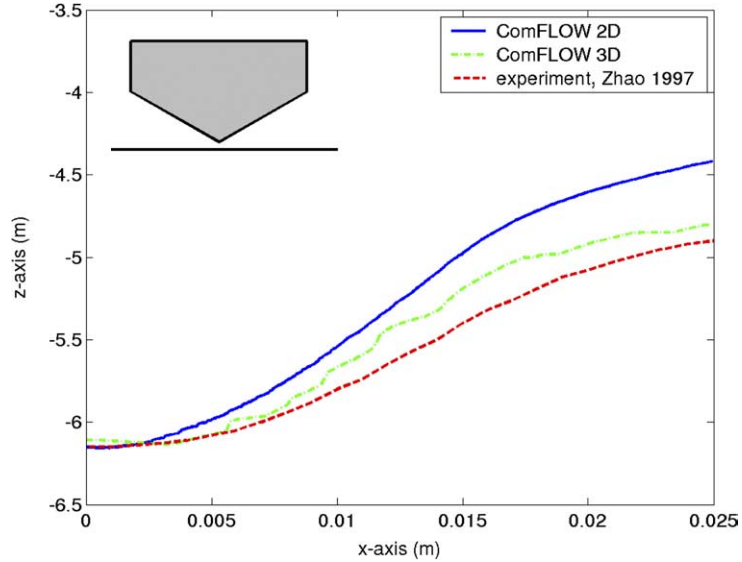
Fig. 28. Vertical velocity of the free falling wedge: experiment compared with two- and three-dimensional simulations.

In Fig. 28 the vertical velocity of the falling wedge is shown as computed by ComFLOW compared with the measurements. An initial velocity of −6.15 m/s is prescribed after which the wedge is falling freely. In the two-dimensional simulation where a grid of 200×200 cells has been used, the absolute value of the vertical velocity is a bit under predicted. In Fig. 29 the calculated vertical force is shown compared with the measurement. The vertical force is over predicted, resulting in a slower motion of the object. To perform a three-dimensional simulation, the gap between the wedge and the side walls has to be determined. In Muz-
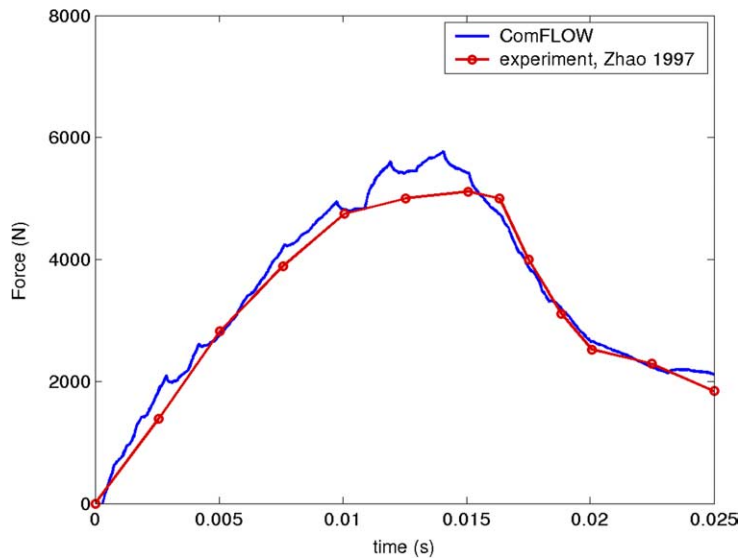

Fig. 29. Vertical force on the free falling wedge in a two-dimensional simulation.

aferija et al. the size of the gap has been altered resulting in good results for a gap of 0.25 m. The same gap size has been used in a three-dimensional simulation with ComFLOW. A grid of $60 \times 60 \times 60$ cells has been used, which is much coarser than the two-dimensional grid. The vertical drop velocity has been plotted in Fig. 28, which shows indeed a better agreement with the experiment. The three-dimensional effect of a smaller vertical force is also present in the simulations using ComFLOW.

10. Conclusions

In the present paper results are shown of the simulation of engineering problems using a Navier–Stokes solver with a VOF-based free-surface displacement. The adopted finite volume discretisation with a cut-cell method on a fixed Cartesian grid results in a stable method, although the size of the fluid cells can get arbitrary small. The method has shown to converge in time and space by performing regular wave simulations with increasing resolution. The combination of the original VOF method with a local height function has shown to improve the free-surface treatment by examining the rotation of a slotted disk. By choosing good numerical boundary conditions for the velocities at the free surface, the method can accurately simulate steep waves and is very robust.

The dambreak simulation has been performed with a box in the flow on which pressures and forces have been calculated. The comparison of the present method with experimental results is very good. The method is well able to predict the impact loads resulting from a highly complex flow.

To validate the impact of a moving object, water entry tests have been performed for wedges with different dead-rise angles, a cone and a circular cylinder. The visual agreement between snapshots of the simulations and photographs of experiments by Greenhow and Lin [18] is very good. On a fine grid the thin jets at the sides of the entering object are resolved fairly well. For the cone and the circular cylinder slamming coefficients have been calculated and compared with available experiments and theoretical estimates, which gives a satisfactory agreement. Finally, a free falling wedge has been simulated, where a full coupling between the liquid motion and the body motion has been established. The resulting vertical velocity and vertical force have been compared with an experiment by Zhao [45]. The two-dimensional effect that slows down the wedge too much has been made visible by performing a three-dimensional simulation.

The results of the simulations give much confidence in the performance of the method. The method will be developed further in the coming years by extending it towards two-phase flows. A major advantage of a two-phase model is that the boundary conditions for the velocities at the free surface, which have a large influence on the robustness and accuracy of the method, are not needed anymore. The method will also be extended with a coupling to an outer domain, where waves are generated using a computationally much cheaper potential flow diffraction code. In this way the ComFLOW domain can be limited to the close surroundings of the places of impact. For validation, simulations of a moving vessel in high waves resulting in green water on the deck will be performed and compared with experiments. Besides green water and bow impact simulations, the focus will also be on sloshing, e.g. on board LNG tankers.

References

[1] D. Battistin, A. Iafrati, Hydrodynamic loads during water entry of two-dimensional and axisymmetric bodies, J. Fluid Struct. 17 (2003) 643–664.

[2] E.F.F. Botta, M.H.M. Ellenbroek, A modified SOR method for the poisson equation in unsteady free-surface flow calculations, J. Comput. Phys. 60 (1985) 119–134.

[3] B. Buchner, Green water on ship-type offshore structures, Ph.D. Thesis, Delft University, 2002.

[4] I.M.C. Campbell, P.A. Weynberg, Measurement of parameters affecting slamming, Report No. 440, Wolfson Unit of Marine Technology, Tech. Rep. Centre No. TO-R-8042, Southampton, 1980.

[5] A. Clément, Coupling of two absorbing boundary conditions for 2D time-domain simulations of free surface gravity waves, J. Comput. Phys. 126 (1996) 139–151.

[6] E.F.G. van Daalen, J. Gerrits, G.E. Loots, A.E.P. Veldman, Free surface anti-roll tank simulations with a volume of fluid based Navier–Stokes solver, in: Proceedings of the 15th International Workshop on Water Waves and Floating Bodies, Ceasarea, Israel, 2000.

[7] E.F.G. van Daalen, K.M.T. Kleefsman, J. Gerrits, H.R. Luth, A.E.P. Veldman, Anti-roll tank simulation with a volume of fluid (VOF) based Navier–Stokes solver, in: 23rd Symposium on Naval Hydrodynamics, Val de Rueil, France, 2001, pp. 457–473.

[8] G. Ersdal, A. Kvitrud, Green water on Norwegian production ships, in: Proceedings of the 10th ISOPE Conference, Seattle, 2000.

[9] O.M. Faltinsen, Sea Loads on Ships and Offshore Structures, Cambridge University Press, Cambridge, 1990.

[10] G. Fekken, Numerical simulation of free-surface flow with moving rigid bodies, Ph.D. Thesis, University of Groningen, 2004. Available from: www.ub.rug.nl/eldoc/dis/science/g.fekken.

[11] G. Fekken, A.E.P. Veldman, B. Buchner, Simulation of green-water loading using the Navier–Stokes equations, in: J. Piquet (Ed.), Proceedings of the Seventh International Conference on Numerical Ship Hydrodynamics, Nantes, France, 1999, pp. 6.3-1–6.3-12.

[12] J. Gerrits, Dynamics of liquid-filled spacecraft, Ph.D. Thesis, University of Groningen, The Netherlands, 2001. Available from: www.ub.rug.nl/eldoc/dis/science/j.gerrits.

[13] J. Gerrits, A.E.P. Veldman, Numerical simulation of coupled liquid–solid dynamics, in: Proceedings of the ECCOMAS 2000, Barcelona, Spain, 2000.

[14] J. Gerrits, A.E.P. Veldman, Transient dynamics of containers partially filled with liquid, in: B. Sarler, C.A. Brebbia (Eds.), Moving Boundaries, vol. VI, 2001, pp. 63–72.

[15] J. Gerrits, A.E.P. Veldman, Dynamics of liquid-filled spacecraft, J. Eng. Math. 45 (2003) 21–38.

[16] R. Glowinsky, T.W. Pan, J. Périaux, A Lagrange multiplier/fictitious domain method for the numerical simulation of incompressible viscous flow around moving rigid bodies: (I) case where the rigid body motions are known a priori, C.R. Academic Science Paris t. 324, Série 1, Mathematical problems in mechanics, 1997.

[17] P. Gorf, N. Barltrop, B. Okan, T. Hodson, R. Rainey, FPSO bow damage in steep waves, in: Proceedings of the Rogue Waves, Brest, 2000.

[18] M. Greenhow, W.-M. Lin, Non linear free surface effects: experiments and theory, Report No. 83-19, Department of Ocean Engineering, MIT, Cambridge, MA, 1983.

[19] M. Greenhow, Wedge entry into initially calm water, Appl. Ocean Res. 9 (1987) 214–223.

[20] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluid 8 (1965) 2182–2189.

[21] D.J.E. Harvie, D.F. Fletcher, A new volume of fluid advection algorithm: the stream scheme, J. Comput. Phys. 162 (2000) 1–32.

[22] C. Hirsch, Numerical Computation of Internal and External Flow, Section 23.3, Wiley, New York, 1990.

[23] C.R. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.

[24] A. Iafrati, A. Di Mascio, E.F. Campana, A level set technique applied to unsteady free surface flows, Int. J. Num. Meth. Fluid 35 (2001) 281–297.

[25] M. Israeli, S.A. Orszag, Approximation of radiation boundary conditions, J. Comput. Phys. 41 (1981) 115–135.

[26] D. Juric, G. Tryggvason, A front tracking method for dendritic solidification, J. Comput. Phys. 123 (1996) 127–148.

[27] H.-C. Kan, H.S. Udaykumar, W. Shyy, R. Tran-Son-Tay, Hydrodynamics of a compound drop with application to leukocyte modeling, Phys. Fluid 10 (1998) 760–774.

[28] G.E. Loots, Fluid–structure interaction in hemodynamics, Ph.D. Thesis, University of Groningen, The Netherlands, 2003. Available from: www.ub.rug.nl/eldoc/dis/science/g.e.loots.

[29] G.E. Loots, B. Hillen, A.E.P. Veldman, The role of hemodynamics in the development of the outflow tract of the heart, J. Eng. Math. 45 (2003) 91–104.

[30] E. Loots, B. Hillen, H. Hoogstraten, A. Veldman, Fluid–structure interaction in the basilar artery, in: R. Herbin, D. Kroener (Eds.), Finite Volumes for Complex Applications III, 2002, pp. 607–614.

[31] W.D.M. Morris, J. Millar, B. Buchner, Green water susceptibility of North Sea FPSO/FSUs, in: Proceedings of the 15th Conference on Floating Production Systems (FPS), London, 2000.

[32] S. Muzaferija, M. Peric, P. Sames, T. Schellin, A two-fluid Navier–Stokes solver to simulate water entry, in: 22nd Symposium on Naval Hydrodynamics, 2000.

[33] S. Osher, J.A. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.

[34] C.S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (1977) 220–252.

[35] W.J. Rider, D.B. Kothe, Reconstructing volume tracking, J. Comput. Phys. 141 (1998) 112–152.

[36] M. Rudman, Volume-tracking methods for interfacial flow calculations, Int. J. Num. Meth. Fluid 24 (1997) 671–691.

[37] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, Annu. Rev. Fluid Mech. 31 (1999) 567–603.

[38] M. Schiffman, D.C. Spencer, The force of impact on a cone striking a water surface vertical entry, Comm. Pure Appl. Math. 4 (1951) 379–417.

[39] L. Skjelbreia, J.A. Hendrickson, Fifth order gravity wave theory, in: Proceedings of the Seventh Coastal Engineering Conference, The Hague, 1960.

[40] I. Orlanski, A simple boundary condition for unbounded hyperbolic flows, J. Comput. Phys. 21 (1976) 251–269.

[41] M. Sussman, E. Fatemi, An efficient, interface preserving level set re-distancing algorithm and its application to interfacial incompressible fluid flow, SIAM J. Sci. Comput. 20 (1999) 1165–1191.

[42] M. Sussman, E.G. Puckett, A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows, J. Comput. Phys. 162 (2000) 301–337.

[43] R.W.C.P. Verstappen, A.E.P. Veldman, Symmetry-preserving discretization of turbulent flow, J. Comput. Phys. 187 (2003) 343–368.

[44] D.L. Youngs, An interface tracking method for a 3D Eulerian hydrodynamics code, Technical Report AWRE/44/92/35, Atomic Weapons Research Establishment, 1987.

[45] R. Zhao, O. Faltinsen, J. Aarsnes, Water entry of arbitrary two-dimensional sections with and without flow separation, in: 21st Symposium on Naval Hydrodynamics, 1997.

[46] S.T. Zalesak, Fully multi-dimensional flux corrected transport algorithms for fluid flow, J. Comput. Phys. 31 (1979) 335–362.