INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **53**:229–256
Published online 19 June 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1279

An unsteady single-phase level set method for viscous free surface flows

P. M. Carrica*, †, R. V. Wilson‡ and F. Stern§

IIHR-Hydroscience and Engineering, The University of Iowa, Iowa City, IA 52242, U.S.A.

SUMMARY

The single-phase level set method for unsteady viscous free surface flows is presented. In contrast to the standard level set method for incompressible flows, the single-phase level set method is concerned with the solution of the flow field in the water (or the denser) phase only. Some of the advantages of such an approach are that the interface remains sharp, the computation is performed within a fluid with uniform properties and that only minor computations are needed in the air. The location of the interface is determined using a signed distance function, and appropriate interpolations at the fluid/fluid interface are used to enforce the jump conditions. A reinitialization procedure has been developed for non-orthogonal grids with large aspect ratios. A convective extension is used to obtain the velocities at previous time steps for the grid points in air, which allows a good estimation of the total derivatives. The method was applied to three unsteady tests: a plane progressive wave, sloshing in a two-dimensional tank, and the wave diffraction problem for a surface ship, and the results compared against analytical solutions or experimental data. The method can in principle be applied to any problem in which the standard level set method works, as long as the stress on the second phase can be specified (or neglected) and no bubbles appear in the flow during the computation. Copyright © 2006 John Wiley & Sons, Ltd.

Received 1 November 2005; Revised 4 April 2006; Accepted 20 April 2006

KEY WORDS: free surface; incompressible viscous flows; single-phase level set; unsteady

INTRODUCTION

Level set methods are becoming increasingly popular for the solution of fluid problems involving moving interfaces [1]. In the case of fluid/fluid interfaces, the level set methods can predict the

*Correspondence to: P. M. Carrica, IIHR-Hydroscience and Engineering, The University of Iowa, 300 South Riverside Dr., Iowa City, IW 52242-1585, U.S.A.
†E-mail: pablo-carrica@uiowa.edu
‡E-mail: robert-wilson@utc.edu
§E-mail: frederick-stern@uiowa.edu

Contract/grant sponsor: Office of Naval Research; contract/grant number: N00014-01-1-0073

evolution of complex free surface topologies including waves with largest slopes such as spilling and breaking waves, deforming bubbles and droplets, break up and coalescence, etc. Since the introduction of the level set method by Osher and Sethian [2], a large amount of bibliography on the subject has been published and several types of problems have been tackled with this method; see, for instance, the cited review by Sethian and Smereka [1] and the work by Osher and Fedkiw [3].

There is a class of fluid/fluid problems in which the interface between the fluids can be considered as a free-boundary, and therefore the computation can be limited to the more viscous and dense fluid. A most important set of problems of this class is the flow around surface-piercing bodies (like ship hulls) and around submerged bodies (as the flow past a submerged hydrofoil). The idea of solving only the water phase and use of appropriate boundary conditions at the free surface is not new, and has been used with almost any interface tracking method. Volume of fluid (VOF) methods solving only the water (or the denser) phase are common (see, for instance, Reference [4]).

Solving only the water phase in level set methods (here called the single-phase level set method) presents several advantages over the classic level set approach in which both fluids are solved (or two-phase level set method). One of them is that in the air phase only extension velocities are needed, which makes the problem considerably easier to converge. This results in a more robust computation, and in many cases faster. On one hand, the pressure does not need to be solved in air, and the velocities and turbulence quantities obey a linear pure convection equation, both resulting in a faster computation. On the other hand, the introduction of a boundary condition on the interface for the pressure and the interface detection has a cost. It is not clear if the single-phase level set method will be in general faster than an equivalent two-phase level set method. Limited testing on a two-dimensional flow over a bump (400 000 grid-points grid) resulted in speed-ups of over 40%, but this would be strongly dependent on the problem type and the ratio of grid points in air to the total number of grid points.

Perhaps the most important advantage of single-phase level set methods is the gain in robustness resulting from the fact that a problem of a single fluid with constant properties is solved. In this case, the computation of the pressure can be done in a standard way without pressure and velocity oscillations at the interface that are common in two-phase level set methods with large density ratios.

It is stressed that in principle the single-phase level set method can handle any problem that can be solved with two-phase level set methods, as long as the two conditions discussed next are met. The first condition arises from the fact that in single-phase level set methods (or any other method in which only the water phase is solved) the continuity condition will not be satisfied in the air phase, thus the method is not suitable for problems in which the air phase somehow gets pressurized during the computation. This means that the method will yield non-physical results if air is trapped or bubbles are formed inside the liquid during the calculation. The second condition, related to the first, is that the stresses caused on the liquid phase by the air phase must be negligible since, again, no computations are performed in the air and those stresses are imposed to be zero, or specified to some value if, for instance, breaking waves or wind-induced stresses are modelled. Other than these limitations, the method has no restrictions on surface topology, allowing for large-amplitude and/or very steep waves.

The main application pursued in the author's research programme is the computation of the viscous free surface flow in large surface-piercing bodies. This is a difficult problem that involves complicated three-dimensional geometries, which leads to non-orthogonal curvilinear grids with high aspect ratio, very high Reynolds numbers and complex free surface topologies. Both surface-tracking and surface-capturing methods have been used to tackle these types of problems.

DOI: 10.1002/fld

In surface-tracking methods, the computational grid is fitted to the free surface, and therefore is not fixed in time. This type of method can be higher-order accurate both in space and time and takes advantage of all the points in the grid since only the water phase is grided. The grid deformation process to fit the grid to the free surface works well and is robust as long as the free surface slope remains small. Examples of successful applications for free surface ship flows include resistance computation without [5] and with propellers [6], forward speed diffraction [7, 8], roll decay motion [9] and pitching and heaving motion in regular head seas [10]. Unfortunately, as the deformation of the free surface increases it is difficult to prevent grid quality deterioration and computation breakdown. One of the main conclusions drawn after the Gothenburg 2000 workshop on ship hydrodynamics [11] was that level set methods show promising results and further research for surface-capturing methods should be pursued to overcome the limitations of surface-tracking methods.

Besides level set methods, surface-capturing methods include volume of fluid (VOF) methods [12], front tracking methods [13], and other variations in which a colour or volume fraction function is computed. Among others, notable examples of the application of surface-capturing methods for flows around floating bodies are in Reference [14], where a surface-capturing method is used to simulate the flow around a surface-piercing blunt body, and in Sato *et al*. [15], who studied pitching and heaving in linear incident waves. Another surface-capturing method for ships, based on a gas volume fraction approach, is presented in Reference [16].

Level set methods have been applied for both submerged and surface-piercing body problems. Vogt and Larsson [17] used both single and two-phase level set methods to study the flow around a submerged two-dimensional hydrofoil in steady-state. In References [18–20], the authors solve more complicated three-dimensional flows around container ships using the two-phase level set approach with body-fitted coordinates. Three-dimensional flows around ships have been solved in steady-state using a single-phase level set method [21]. Note that all these single-phase methods are devised and used for steady-state applications only.

In this paper, an unsteady single-phase level set method for viscous, incompressible flow is presented. The computation of the total time derivative is a key issue to make the method time-accurate. The implementation of the overall scheme is discussed.

The method was tested against two two-dimensional and one three-dimensional unsteady cases: a linear progressive wave, the sloshing in a steady tank, and the wave diffraction by a ship. Results are compared against analytical or experimental data showing good agreement.

SINGLE- AND TWO-PHASE LEVEL SET METHODS

The standard level set method for incompressible free surface viscous flows originated about ten years ago [22] and has become very popular. This method is called the two-phase level set, since the solution is obtained in both fluids as discussed below.

In a two-phase flow, the instantaneous local equations of motion within each fluid can be written as [23]

$$\frac{\partial \mathbf{v}_k}{\partial t} + \mathbf{v}_k \cdot \nabla \mathbf{v}_k = -\frac{1}{\rho_k} \nabla p_k + 2 \frac{\mu_k}{\rho_k} \nabla \cdot \mathbf{D}_k + \mathbf{g} \tag{1}$$

$$\nabla \cdot \mathbf{v}_k = 0 \tag{2}$$

DOI: 10.1002/fld

where the subscript $k = l$ or g indicates the phase present at a given point in space, which in the case presented herein can be either liquid or gas. $\mathbf{v}$, p and $\mathbf{D}$ are the velocity, pressure and rate of deformation tensor, ρ_k and μ_k are the density and viscosity of fluid k, and $\mathbf{g}$ is the gravity acceleration. At the interface, the interfacial boundary conditions or jump conditions apply. In the case of immiscible fluids:

$$[-p\mathbf{I} + 2\mu\mathbf{D}] \cdot \mathbf{n} = -(\sigma\kappa\mathbf{n} + \nabla_i\sigma) \tag{3}$$

where the bracket means $l - g$ and the normal $\mathbf{n}$ is taken from the liquid to the gas. The second term on the right-hand side of Equation (3) is the stress due to gradients of surface tension or Marangoni effect [24], usually important when large gradients of temperature are present, as in boiling flow. ∇_i denotes the gradient in the local free surface coordinates. The interfacial curvature κ is computed from

$$\kappa = \nabla \cdot \mathbf{n} \tag{4}$$

The jump conditions (3) can be integrated into the equations of motion (1) and (2), resulting in a body force concentrated in the interface of a single incompressible fluid with variable properties [25]:

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla\mathbf{v} = -\frac{1}{\rho}\nabla p + 2\frac{\mu}{\rho}\nabla \cdot \mathbf{D} + \mathbf{g} + (\sigma\kappa\mathbf{n} + \nabla_i\sigma)\delta(\phi) \tag{5}$$

where ϕ is a distance to the interface function, positive in liquid and negative in gas, and δ is a smoothed delta function defined later in this section. The location of the interface is then given by the zero level set of the function ϕ, known as the level set function. Since the free surface is a material interface (in the absence of interfacial mass transfer such as evaporation or condensation), then the equation for the level set function is

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla\phi = 0 \tag{6}$$

and from the level set function the normal can be computed as

$$\mathbf{n} = -\frac{\nabla\phi}{|\nabla\phi|} \tag{7a}$$

Since the fluid properties in Equation (5) change discontinuously across the interface and the concentrated surface tension force also becomes infinite in an infinitesimal volume, direct solution of Equation (5) is naturally difficult. Two approaches are usually followed to overcome these difficulties.

In the standard level set method, the interface is smoothed across a finite thickness region, usually a few grid points thick. The fluid properties and the delta function are thus modified as [21]

$$\rho(\phi) = \rho_g + (\rho_l - \rho_g)H(\phi) \tag{7b}$$

$$\mu(\phi) = \mu_g + (\mu_l - \mu_g)H(\phi) \tag{7c}$$

$$\delta(\phi) = \frac{dH(\phi)}{d\phi} \tag{7d}$$

DOI: 10.1002/fld

where the smoothed Heaviside function is usually expressed as

$$H(\phi) = \begin{cases} 0 & (\phi < -\alpha) \\ 0.5[1 + \phi/\alpha + \sin(\pi\phi/\alpha)/\pi] & (|\phi| \leqslant \alpha) \\ 1 & (\phi > \alpha) \end{cases} \tag{8}$$

with α the half-thickness of the property transition region. An important step is to maintain the level set function a distance function within the transition region at all times. This is achieved by the reinitialization step, discussed later in this paper.

One drawback of the level set method is the introduction of the transition region. This results in smearing of the flow properties and variables, forcing them to be continuous at the interface regardless of the appropriate jump conditions. This problem is solved by the Ghost Fluid Method, in which the jump conditions are introduced implicitly in the formulation by solving for a 'ghost' fluid across the interface [26, 27]. This approach, though easy to implement, forces the solution of two fluid fields at each grid node, one for each fluid. This results in additional computational cost that can be very demanding in large three-dimensional computations.

It is then desirable for many applications to be able to solve a single-phase problem in a fixed grid, capturing the interface with appropriate enforcement of the jump conditions and still retaining the advantages of level set methods. These applications generally involve air–water flows in which the density and viscosity ratios are about 1000 and 75, respectively. Under these conditions, the interface can be taken as shear stress free for most applications, as frequently done with interface tracking algorithms. In this way, the computational domain to solve the RANS equations is restricted to the grid points in water plus a few nodes in air to enforce the jump conditions, with the consequent economy of resources. A second advantage is that the continuity equation is enforced always in a single fluid, thus allowing the use of standard collocated methods without the usual pressure and velocity oscillations that occur at the interface between fluids with a large density ratio [1, p. 355]. Such a method is the single-phase level set method [17, 21]. In the next sections an extension to the single-phase level set method is presented, in order to make it appropriate for unsteady problems.

UNSTEADY SINGLE-PHASE LEVEL SET METHOD DETAILS

In this section, the details of the single-phase level set method are described, including derivation and implementation of the jump conditions, reinitialization of the level set as a distance function and computation of the total time derivatives.

Governing equations

Since the problem is solved in a single fluid, the RANS equations can be non-dimensionalized as usual to obtain

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = -\nabla p + \nabla \cdot \left[\frac{1}{Re_{\text{eff}}} \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T \right) \right] + \mathbf{S} \tag{9}$$

DOI: 10.1002/fld

where the source **S** includes any volumetric source with the exception of the gravity, which is lumped with the pressure to define the non-dimensional piezometric pressure:

$$p = \frac{p_{\text{abs}}}{\rho U_0^2} + \frac{z}{Fr^2} \tag{10}$$

In Equations (9) and (10) p_{abs} is the absolute pressure, Re_{eff} is the effective Reynolds number and Fr is the Froude number, defined as:

$$Re_{\text{eff}} = \frac{U_0 L}{\nu + \nu_t} \tag{11}$$

$$Fr = \frac{U_0}{\sqrt{gL}} \tag{12}$$

where U_0 is the free-stream velocity and L is the characteristic length. ν_t is the turbulent viscosity, which in this work is obtained after solving the blended k–ω model of turbulence [28]. Since an incompressible fluid is considered, the continuity equation reads:

$$\nabla \cdot \mathbf{v} = 0 \tag{13}$$

To capture the location of the interface, the already introduced level set function is solved:

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0 \tag{6}$$

Enforcement of the jump conditions

In contrast to the two-phase level set method, in which the interfacial jump conditions are embedded naturally on the formulation, the jump conditions at the interface between two fluids, Equation (3), must be treated as a boundary condition enforced explicitly in a single-phase level set approach. The jump condition in any direction tangential to the free surface is [27]

$$[\mu(\nabla \mathbf{v} \cdot \mathbf{n}) \cdot \mathbf{t} + \mu(\nabla \mathbf{v} \cdot \mathbf{t}) \cdot \mathbf{n}] = 0 \tag{14}$$

Neglecting the viscosity in air, and since **t** is an arbitrary vector perpendicular to the normal to the interface, the boundary conditions for the velocity at the interface are obtained:

$$\nabla \mathbf{v} \cdot \mathbf{n} = 0 \tag{15}$$

In the direction normal to the interface, the jump condition in dimensional form can be written as:

$$[p_{\text{abs}} - 2\mu(\nabla \mathbf{v} \cdot \mathbf{n}) \cdot \mathbf{n}] = \sigma\kappa + \nabla_i \sigma \cdot \mathbf{n} \tag{16}$$

As a good approximation in water/air interfaces, the pressure can be taken as constant in the air. Also, because of Equation (15), the second term on the left-hand side of Equation (16) is zero. Thus, the jump condition reduces to:

$$p_{\text{abs}} = \sigma\kappa + \nabla_i \sigma \cdot \mathbf{n} \tag{17}$$

DOI: 10.1002/fld

At this point, the surface tension effects are neglected, because for the class of problems of interest in this paper the curvature of the free surface is small. This is not a limitation of the model and surface tension can be easily included, as done by Di Mascio *et al*. [21]. Introducing the dimensionless piezometric pressure, Equation (10), at the fluid/fluid interface the pressure is

$$p = \frac{z}{Fr^2} \tag{18}$$

Enforcement of the pressure condition at the free surface

The pressure at the free surface is given by Equation (18) and was derived from the jump conditions. In a surface-capturing approach, however, the free surface is not located at the grid points, and therefore an interpolation scheme must be devised to enforce the interfacial pressure condition at the interface location. The free surface itself can be easily identified by locating the change in sign of ϕ between two contiguous grid points along any coordinate line.

There are in this method three types of grid points. Grid points in water with all the first neighbours in water will be computed without any special treatment and need no additional consideration. Grid points in air can have any pressure value, and then Equation (18) is enforced, with z, now, the vertical coordinate at the point. For points in water in which at least one of the neighbours is in air the following discussion applies.

For any grid point p in water that has a neighbour in air $\mathbf{n}_a$, the interfacial pressure condition of Equation (18) is enforced locally. Referring to Figure 1, the relative distance between the grid point in water and the interface is

$$\eta = \frac{\phi_p}{\phi_p - \phi_{n\mathrm{a}}} \tag{19}$$

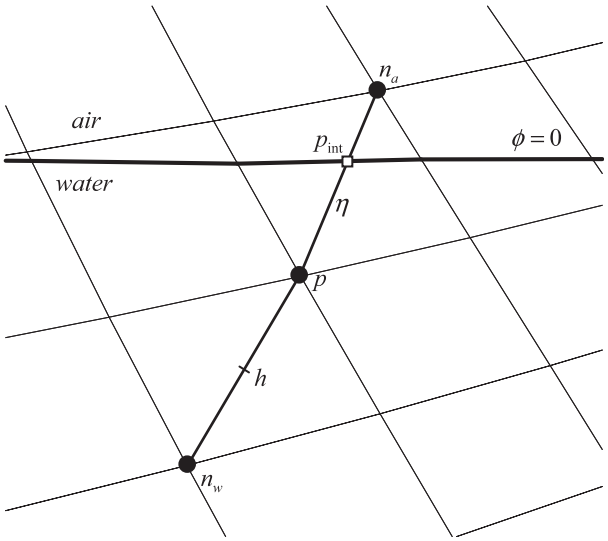

Figure 1. Computation of the free surface location and calculation of the neighbour pressure to enforce the pressure boundary condition.

DOI: 10.1002/fld

thus, interpolating along the line joining the points p and n_a and using Equation (18) the interfacial pressure is obtained:

$$p_{\text{int}} = \frac{(1-\eta)z_p + \eta z_{n\text{a}}}{Fr^2} \tag{20}$$

The pressure at the neighbour in air can then be found by extrapolation from the pressure values at the points h and at the interface, where h is located halfway between the local point p and the opposite neighbour in water to n_a, shown as n_w in Figure 1:

$$p_{n\text{a}} = (p_{\text{int}} - p_h)\frac{\text{dist}(\mathbf{r}_{n\text{a}}, \text{int})}{\text{dist}(\mathbf{r}_p, \text{int}) + \text{dist}(\mathbf{r}_p, \mathbf{r}_h)} + p_{\text{int}} \tag{21}$$

where $p_h = (p_p + p_{n\text{w}})/2$ and $\mathbf{r}_h = (\mathbf{r}_p + \mathbf{r}_{n\text{w}})/2$. This scheme can be easily implemented in a subroutine that computes the matrix coefficients for the pressure Poisson equation, by plugging Equation (21) in every neighbour in air. Since all the necessary neighbours to enforce the interfacial pressure condition are the same necessary to build the pressure matrix (for typical 19-point stencils), no additional connectivity has been added, and therefore can be implemented in multi-block codes with no modifications on the existing inter-block information transfer scheme. This is especially attractive in parallel implementations.

Reinitialization

In two-phase level set methods, the distance function is reinitialized periodically to keep it a smooth distance function and have a transition region uniform in thickness. The reinitialization step is extremely important in single-phase level set methods but for different reasons: the normal must be accurately evaluated at the interface because it is used in the boundary conditions and it must also be reasonable everywhere in air since it is used to extend the velocities into the air to transport the level set function.

In this work, the reinitialization procedure is split in two steps. The first step is a close point reinitialization for those grid points that are neighbours to the interface as in the *fast marching method* [29]. In the second step, a transport equation for the rest of the grid points is solved, using the near-boundary points as Dirichlet boundary conditions. This is easy to implement in parallel environments and reasonably inexpensive.

The method of Adalsteinsson and Sethian [30] was extended to three-dimensional curvilinear grids to obtain a good signed distance for the first neighbours to an interface (the beginning set of their *close* points). In curvilinear grids with very large aspect ratio (in boundary layer grids can be as large as 10^5), the distance to the first neighbour cannot be used to define the geometrical distance to the interface. In Figure 2 such a grid is shown, marking with circles all points to be geometrically reinitialized because they have at least a first neighbour in a different fluid. Consider the point that is filled with white. For that point the closest interface lies in the $+\eta$ direction before crossing the third neighbour. If the reinitialization was done using only first neighbours the interface in the $-\xi$ direction would be the only one found and a poor signed distance would result. This problem becomes unacceptable for very large aspect ratios, where the closest interface might lay several grid points away in some direction. Moreover, the closest interface might be in a different block, and thus that information belongs to a different processor in a typical domain decomposition parallel implementation.

DOI: 10.1002/fld

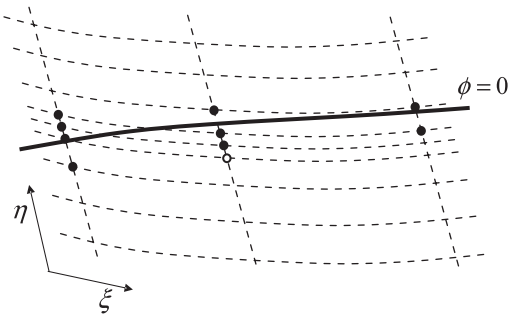

Figure 2. Reinitialization of the close points.

The close point reinitialization algorithm used in this work searches along the grid lines, including neighbour blocks, in each direction, and finds the location of the intersection of the grid lines with the interface. Let these locations be $\mathbf{r}_{\xi+}, \mathbf{r}_{\xi+}, \mathbf{r}_{\eta+}, \mathbf{r}_{\eta-}, \mathbf{r}_{\zeta+}, \mathbf{r}_{\zeta-}$, where it must be noticed that in some directions there may not be any interfaces. The distance from the point to be reinitialized to the interface is given by the distance to the plane formed by the three points:

$$\mathbf{r}_\xi = \min(\mathbf{r}_{\xi+}, \mathbf{r}_{\xi-}), \quad \mathbf{r}_\eta = \min(\mathbf{r}_{\eta+}, \mathbf{r}_{\eta-}), \quad \mathbf{r}_\zeta = \min(\mathbf{r}_{\zeta+}, \mathbf{r}_{\zeta-}) \tag{22}$$

$$d = \frac{[(\mathbf{r}_\xi - \mathbf{r}_p) \times (\mathbf{r}_\eta - \mathbf{r}_p)] \cdot (r_\zeta - \mathbf{r}_p)}{|(\mathbf{r}_\xi - \mathbf{r}_\eta) \times (\mathbf{r}_\xi - \mathbf{r}_\zeta)|} \tag{23}$$

If along any of the gridlines no interface is found then the distance to a line is computed if two interfaces were found, or to a point if only one interface was found.

This algorithm, though not prohibitive, is not very fast because it necessarily involves a much more careful search than that presented in Reference [30]. Thus, for the second step of the reinitialization and for the extension, Equation (15), a PDE is solved. For the reinitialization:

$$\mathbf{n} \cdot \nabla\phi = \text{sign}(\phi_0) \tag{24}$$

where $\mathbf{n}$ is in this case the normal pointing to the fluid being reinitialized. Since $\mathbf{n}$ is either $\nabla\phi$ or $-\nabla\phi$, Equation (24) is an eikonal equation and propagates information from the interface outwards. The corresponding Dirichlet boundary conditions for (24) are given by the value of ϕ at the close points. Notice also that, as written, Equation (24) is nonlinear and requires some iterations to converge, though its solution is very fast in the overall scheme.

Computing the total time derivatives

The proper computation of the total time derivative terms near the interface is a problem in single-phase level set techniques. To understand this, consider an interface moving vertically, as shown in Figure 3, where the total derivative at point p needs to be computed. At the current time shown, the point p is in water and in the previous time step was in air. Since an Eulerian approach is being used, the local time derivative at point p is expressed as a first-order approximation as

$$\frac{\partial\varphi}{\partial t} \cong \frac{\varphi_t - \varphi_{t-\Delta t}}{\Delta t} \tag{25}$$

DOI: 10.1002/fld

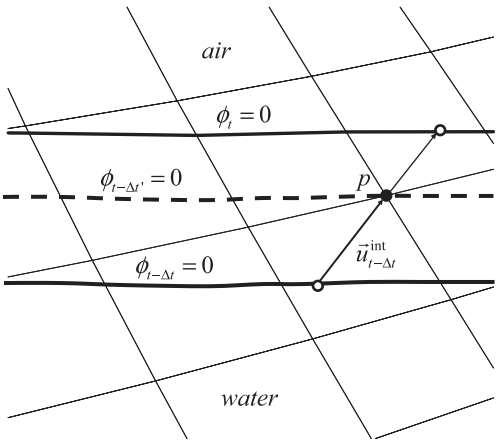

Figure 3. Computation of the time derivative at a grid point that is changing fluid.

where φ is any velocity or turbulence variable. The problem appears because $\varphi_{t-\Delta t}$ was in air in the previous time step, and therefore its value does not satisfy the field equations since it was computed using the extension of Equation (15). In References [17, 21] the authors recognize this limitation and deal with steady-state problems only. Notice that in the standard two-phase level set method similar problems could arise.

Assume, for instance, that the level set method allows completely sharp interfaces and that in Figure 3 the air is moving from left to right and the liquid is moving vertically, which is allowed because the stress and velocities must be continuous at the interface but not between contiguous grid points crossing the interface. Then when liquid reaches a point that before was in air it will have the information that in the previous time steps was moving from left to right. This results in a spurious inertia and consequently the liquid velocity will gain some left to right component. This effect is highly limited in level set methods by the thickness of the transition region and the limitations on the time step in explicit methods that prevent large changes on the interface location between time steps.

In this analysis, the velocities are used, though all the variables should be computed across the interface in the same way.

In single-phase level set approaches the interface is sharp, and thus suitable velocities for the previous time steps need to be defined at grid points which were previously in air. One possibility is to use the extended velocity as the previous time step velocity (or time steps in a higher-order scheme). However, this extended velocity will not result in the right total derivative. An extension that will yield a good approximation of the total derivative is presented below.

Consider a particle belonging to the free surface that at time $t - \Delta t'$ during the time step advancing from $t - \Delta t$ to t crosses a grid point, marked as p in Figure 3. From $t - \Delta t'$ to t, the total derivative of the velocity is

$$\left.\frac{D\mathbf{u}}{Dt}\right|_1 = \frac{\mathbf{u}(\mathbf{r}_p, t - \Delta t') - \mathbf{u}^{\mathrm{int}}_{t-\Delta t}}{\Delta t - \Delta t'} \tag{26}$$

DOI: 10.1002/fld

where $\mathbf{u}(\mathbf{r}_p, t-\Delta t')$ is the velocity of the particle on the interface that at $t-\Delta t'$ is exactly on grid point p, and $\mathbf{u}^{\text{int}}_{t-\Delta t}$ is the velocity of the same particle at $t-\Delta t$. This is a Lagrangian evaluation of the acceleration.

From $t-\Delta t'$ to t the Eulerian acceleration can be used, since from that time on the grid point p is in water:

$$\left.\frac{D\mathbf{u}}{Dt}\right|_2 = \frac{\mathbf{u}(\mathbf{r}_p, t) - \mathbf{u}(\mathbf{r}_p, t-\Delta t')}{\Delta t'} + \mathbf{u}(\mathbf{r}_p, t)\cdot\nabla\mathbf{u}(\mathbf{r}_p, t) \tag{27}$$

Since both total derivatives are computed within the same time step advancing from $t-\Delta t$ to t, $D\mathbf{u}/Dt|_1 = D\mathbf{u}/Dt|_2$. Solving for the unknown velocity $\mathbf{u}(\mathbf{r}_p, t-\Delta t')$:

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}(\mathbf{r}_p, t) - \mathbf{u}^{\text{int}}_{t-\Delta t}}{\Delta t} + \frac{\Delta t'}{\Delta t}\mathbf{u}(\mathbf{r}_p, t)\cdot\nabla\mathbf{u}(\mathbf{r}_p, t) \tag{28}$$

Notice that the ratio $\Delta t'/\Delta t$ can be computed from the level set function as

$$\frac{\Delta t'}{\Delta t} = \frac{\phi(\mathbf{r}_p, t)}{\phi(\mathbf{r}_p, t) - \phi(\mathbf{r}_p, t-\Delta t)} \tag{29}$$

Thus, according to Equations (28) and (29), the total time derivative in grid points in which the level set function changes from air to water is replaced by

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}(\mathbf{r}_p, t) - \mathbf{u}^{\text{int}}_{t-\Delta t}}{\Delta t} + \frac{\phi(\mathbf{r}_p, t)}{\phi(\mathbf{r}_p, t) - \phi(\mathbf{r}_p, t-\Delta t)}\mathbf{u}(\mathbf{r}_p, t)\cdot\nabla\mathbf{u}(\mathbf{r}_p, t) \tag{30}$$

where the interfacial particle velocity at $t-\Delta t$, $\mathbf{u}^{\text{int}}_{t-\Delta t}$, remains to be determined. Notice that replacing this velocity in Equation (30) by the velocity at point p in the previous time step the following expression is obtained:

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}(\mathbf{r}_p, t) - \mathbf{u}(\mathbf{r}_p, t-\Delta t)}{\Delta t} + \frac{\phi(\mathbf{r}_p, t)}{\phi(\mathbf{r}_p, t) - \phi(\mathbf{r}_p, t-\Delta t)}\mathbf{u}(\mathbf{r}_p, t)\cdot\nabla\mathbf{u}(\mathbf{r}_p, t) \tag{31}$$

The implementation of Equation (31) requires an easy modification to the convective term and to load in the points in air the interfacial velocity of the particle that will pass through grid point p, and assign that to the previous time step velocity. This second step is achieved by simply extending the variables not with the normal but with the velocity itself:

$$\mathbf{u}(\mathbf{r}_p, t-\Delta t)\cdot\nabla\mathbf{u}(\mathbf{r}_p, t-\Delta t) = 0 \tag{32}$$

which performs convective extension, thus assigning the interfacial velocity properly. It should be noticed that Equation (32) does not satisfy the normal zero gradient boundary conditions, and thus all the extensions in air during a given time step must be carried out using Equation (15). Once the time step is converged, the extension of Equation (32) is performed and loaded, only in air, as the previous time step velocity. Another view of Equation (32) is as an extension along the particle path or characteristic.

DOI: 10.1002/fld

IMPLEMENTATION

The single-phase level set model was implemented in the code CFDShip-Iowa [31], a parallel unsteady RANS code. The previous version was based on a surface-tracking algorithm that, though successful for many applications, failed to resolve problems with complex free surface topologies. The CFDShip-Iowa code uses body-fitted structured multi-block grids with ghost cells and chimera interpolations to accommodate complex geometries. The equations are first transformed from the physical (x, y, z, t) domain to the non-orthogonal computational domain (ξ, η, ζ, τ). The resulting equations in water are

$$\frac{\partial U_i}{\partial \tau} + \frac{1}{J} b_j^k \left(U_j - \frac{\partial x_j}{\partial \tau} \right) \frac{\partial U_i}{\partial \xi^k} = -\frac{1}{J} b_i^k \frac{\partial p}{\partial \xi^k} + \frac{1}{J} \frac{\partial}{\partial \xi^j} \left(\frac{b_\mathrm{I}^j b_\mathrm{I}^k}{J \mathrm{Re}_\mathrm{eff}} \frac{\partial U_i}{\partial \xi^k} \right) + S_i \tag{33}$$

$$\frac{1}{J} \frac{\partial}{\partial \xi^j} (b_i^j U_i) = 0 \tag{34}$$

$$\frac{\partial \phi_i}{\partial \tau} + \frac{1}{J} b_j^k \left(U_j - \frac{\partial x_j}{\partial \tau} \right) \frac{\partial \phi_i}{\partial \xi^k} = 0 \tag{35}$$

The convective terms are discretized using a second-order upwind scheme and the time derivatives are discretized using a second-order backward differencing formula. This discretization scheme also applies to the grid velocity terms in Equations (33) and (35). The viscous terms in Equation (33) are discretized using second-order central differences. Similar differences schemes are used to discretize the turbulence equations.

The incompressibility constraint is enforced using the PISO algorithm [32]. Upon discretization, the i-velocity component at node p can be written as

$$U_i = -\frac{a_{nb} U_{i,nb} - S_i}{a_p} - \frac{b_i^k}{J a_p} \frac{\partial p}{\partial \xi^k} \tag{36}$$

with a_p and a_{nb} the pivot and neighbour coefficients of the discretized momentum equations. Enforcing the continuity equation, Equation (34), results in a Poisson equation for the pressure:

$$\frac{\partial}{\partial \xi^j} \left(\frac{b_l^j b_l^k}{J a_p} \frac{\partial p}{\partial \xi^k} \right) = \frac{\partial}{\partial \xi^j} \frac{b_i^j}{a_p} (a_{nb} U_{i,nb} - S_i) \tag{37}$$

Notice that in curvilinear, non-orthogonal grid systems, Equation (37) leads to a 19-point stencil. In order to avoid pressure-velocity decoupling, the contravariant pressure gradients in Equation (37) must be evaluated at the faces of the cells in the computational domain. This forces the metric coefficients to be available at half-cell locations and these are computed directly to prevent the appearance of artificial mass sources if averages are performed [33].

In air, the velocity is extended from the air/water interface using Equation (15) in discretized form:

$$b_j^k \mathbf{n}_j \frac{\partial U_i}{\partial \xi^k} = 0 \tag{38}$$

which enforces the velocity boundary condition at the interface and also provides a velocity field to transport the level set function. In addition, the same extension procedure is performed for the

DOI: 10.1002/fld

turbulence quantities k and ω. The reinitialization of the level set function as a distance, Equation (24), is discretized similar to Equation (38) and requires a few iterations to converge since the transport velocity depends on the solution. In Equation (38), n_j is the normal to the free surface, defined after Equation (24) as the signed level set function gradient.

The convective extension, Equation (32), is discretized as the convective terms in Equation (33), dropping all the other terms. This is solved only at the end of each time step and the resulting velocity loaded on the points in air as the previous time step velocity. It was noticed that performing the convective extension on the turbulence quantities has little effect on the results, but is very important for the velocities.

The resulting algebraic systems for the variables (u, v, w, p, ϕ, k, w) are solved in sequential form and iterated within each time step until convergence. For the pressure Poisson equation, a matrix system is built and solved using the PETSc toolkit [34], while all the other systems are solved using the ADI method.

EXAMPLES

Since the aim of this paper is an unsteady method, the focus is on unsteady example problems. The method has been tested on steady-state three-dimensional problems, including the flow around a surface ship model DTMB 5415 for different Froude numbers [35].

The numerical method described in the previous sections has been applied in this paper to three unsteady cases: a two-dimensional linear progressive wave, a viscous wave in a two-dimensional tank and a three-dimensional wave diffraction problem around a surface combatant.

Linear progressive wave

For analysis purposes, an ideal linear plane progressive wave is attractive because it has an exact analytical solution, and therefore allows for direct comparison with the numerical method. A small amplitude wave of the form:

$$\zeta(x, t) = A \sin(kx - \omega t) \tag{39}$$

with k the wavenumber and ω the encounter frequency, was imposed on a two-dimensional domain as initial condition ($t = 0$). Equation (39) is also the exact solution of the elevation as a function of time for comparison purposes; see, for instance, [36]. Initial velocities and pressure are also imposed according to the exact solution:

$$U(x, z, t) = U_0 + \frac{A}{Fr} \sqrt{k} e^{kz} \cos(kx - \omega t) \tag{40}$$

$$W(x, z, t) = \frac{a}{Fr} \sqrt{k} e^{kz} \sin(kx - \omega t) \tag{41}$$

$$p(x, z, t) = \frac{A}{Fr^2} e^{kz} \left[\cos(kx - \omega t) - \frac{1}{2} \frac{A}{k} e^{kz} \right] \tag{42}$$

The domain extends vertically from $z = -1.5$ to 0.1 and horizontally from $x = 0$ to $x = x_{\max}$. At $x = 0$ Equation (39) is used to impose the level set inlet boundary condition as:

$$\phi(0, t) = A \sin(-\omega t) - z \tag{43}$$

DOI: 10.1002/fld

Other inlet boundary conditions follow the exact solution. In this case, the wave amplitude is $A=0.004$, small enough to ensure negligible finite depth effects. The wavelength λ is set to 1, resulting in a wavenumber $k=2\pi/\lambda=2\pi$ or a small steepness $Ak=0.025$, which guarantees linear wave behaviour. The Froude number is $Fr=0.5$, and the Reynolds is set to $Re=10^9$ so that viscous effects are negligible. This results in a phase velocity $V_p=1+\sqrt{\lambda/2\pi}/Fr=1.798$. An absorption condition (numerical beach) is imposed at the exit to avoid spurious amplitude oscillations. This is implemented damping the level set function of Equation (6) as

$$\frac{\partial\phi}{\partial t}+\mathbf{v}\cdot\nabla\phi=-\beta(\phi+z) \tag{44}$$

where β is a function that is zero everywhere except on the damping region where it grows quadratically from zero at $x=x_d$ to a maximum of 10 at the exit.

Figure 4 shows a comparison of the wave predicted by the convective extension explained in the previous section with the normal extension for the previous time step velocity. In this case, the domain extends to $x_{\max}=9$ and the numerical beach is activated starting at $x_d=5$. The grid has 4 blocks, each block 136 grid points in the x-direction, covering each wavelength by 36 grid points, by 70 in z, conveniently clustered around the wave amplitude. Eight non-dimensional time units were run with constant time step of 0.01 to cover about 14 wave periods.

It is clear that the normal extension tends to grow the wave as it travels in the positive x-direction, while the proposed extension maintains the wave shape appropriately. To make a quantitative comparison with the exact solution, a similar case was run but with a longer domain, extending to $x_{\max}=15$ and damping the free surface starting at $x_d=10$. This covers 10 wavelengths.

Figure 5 shows a plot of the numerical and exact solutions, this latter up to $x_d=10$ since the exact solution does not apply in the damped region. The wave phase is followed correctly, with an error of only $\Delta x=0.00655$ (0.655%) at the end of the computation on the crossing by zero at the

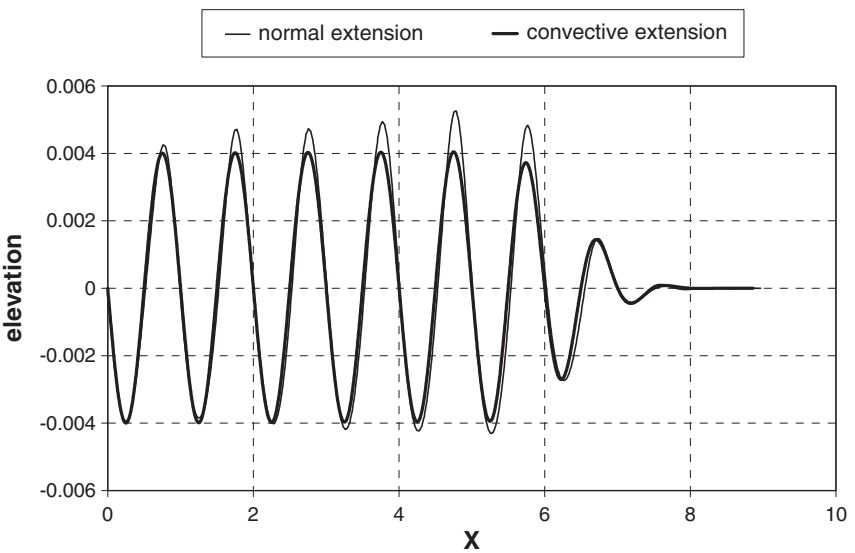

Figure 4. Comparison of normal extension with convective extension for a progressive wave.

DOI: 10.1002/fld

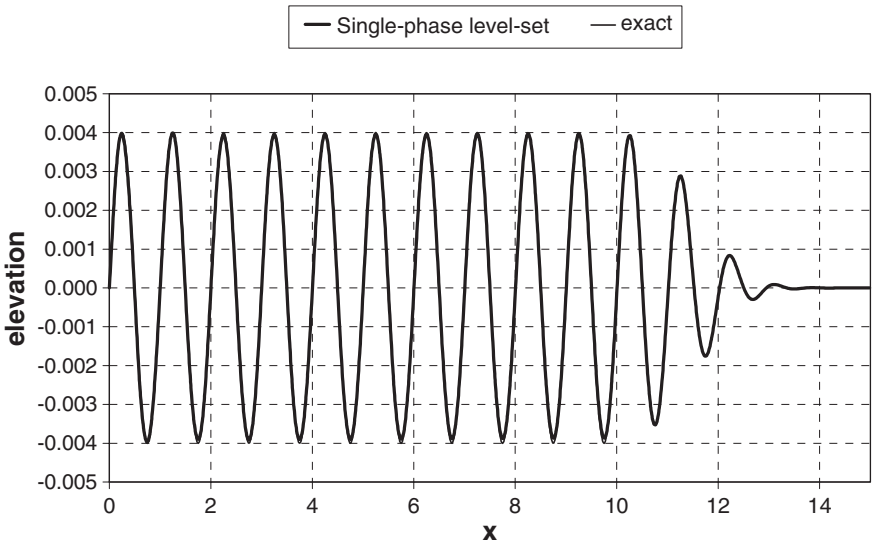

Figure 5. Exact and numerical solution of a progressive linear wave.

tenth wavelength. The wave deforms and the amplitude decreases slowly as the wave progresses. At the last wave before the numerical beach is activated, the maximum of the wave has decreased 1% and the minimum has 2.8% error. The RMS difference between the numerical and the exact solutions gives an idea of the deformation of the wave, in this case 1.9%. Following an initial transient, to allow the first wave entering the domain to reach the numerical beach, there is no significant change with time on these errors. This shows that some level of numerical diffusion is affecting the computation, but to acceptable levels for most applications.

Sloshing in a fixed rectangular tank

Consider a tank with a length that is twice the depth of the still water level, in which a viscous fluid is allowed to oscillate freely. The grid is comprised of 4 blocks each with 51×46 grid points in the x- and z-directions, respectively, and the extents are $x \in (-1, 1)$ and $z \in (-1, 0.1)$ (see Figure 6). At $t = 0$ the free surface has a sinusoidal profile of small amplitude ζ_0 and wavelength $2d$, which in this test case is represented by

$$\zeta_0(x) = 1 - 0.01 \sin\left(\frac{\pi x}{d}\right) \tag{45}$$

with d the tank depth. The free surface is then released and the wave elevation shows an amplitude decay in time $\zeta(x, t)$. In order to simulate an infinite wave, slip conditions at the lateral walls and at the bottom of the computational domain are imposed. Since the velocities are very small at the bottom, the boundary condition there has very little effect.

This problem was studied analytically by Wu *et al.* [37], who solved the linearized Navier–Stokes equations. The solutions are expressed for different (small) Reynolds numbers ($Re = d\sqrt{gd}/v$) as a function of a dimensionless time expressed as $\tau = t\sqrt{g/d}$. In this work, the

DOI: 10.1002/fld

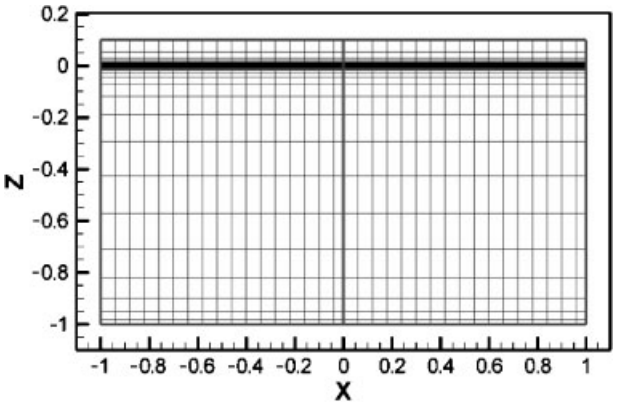

Figure 6. Two-dimensional grid used for the tank case. The bold grey lines indicate the limits of the blocks. One every third grid point is shown for clarity.

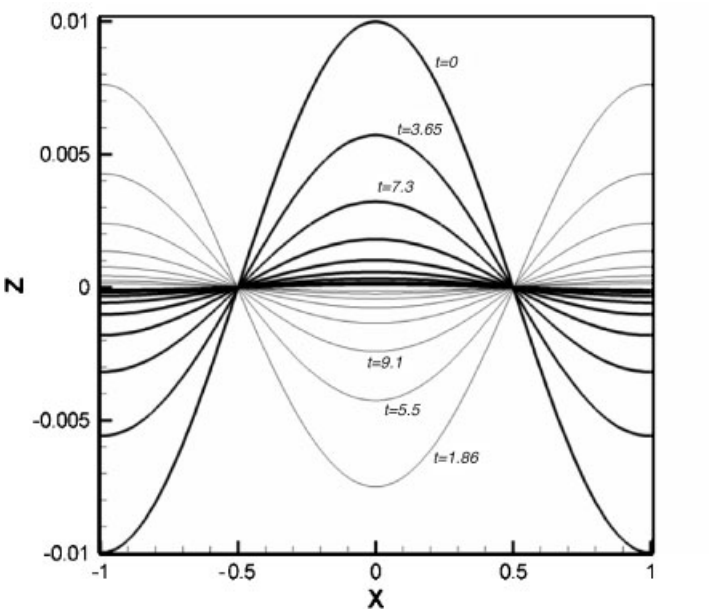

Figure 7. Free surface elevation evolution in a tank for different times at $Re = 100$.

same non-dimensionalization is obtained by setting $Fr = 1$ and using the same initial amplitude as in the analytical problem. In addition, Eatock Taylor *et al.* [38] provide numerical solutions to this problem using a pseudo-spectral matrix element method, which is deemed to be very accurate, though the authors use linearized free surface conditions.

Figure 7 shows the free surface evolution for different non-dimensional times for a problem with $Re = 100$. To evaluate quantitatively the performance of the single-phase level set method,

DOI: 10.1002/fld

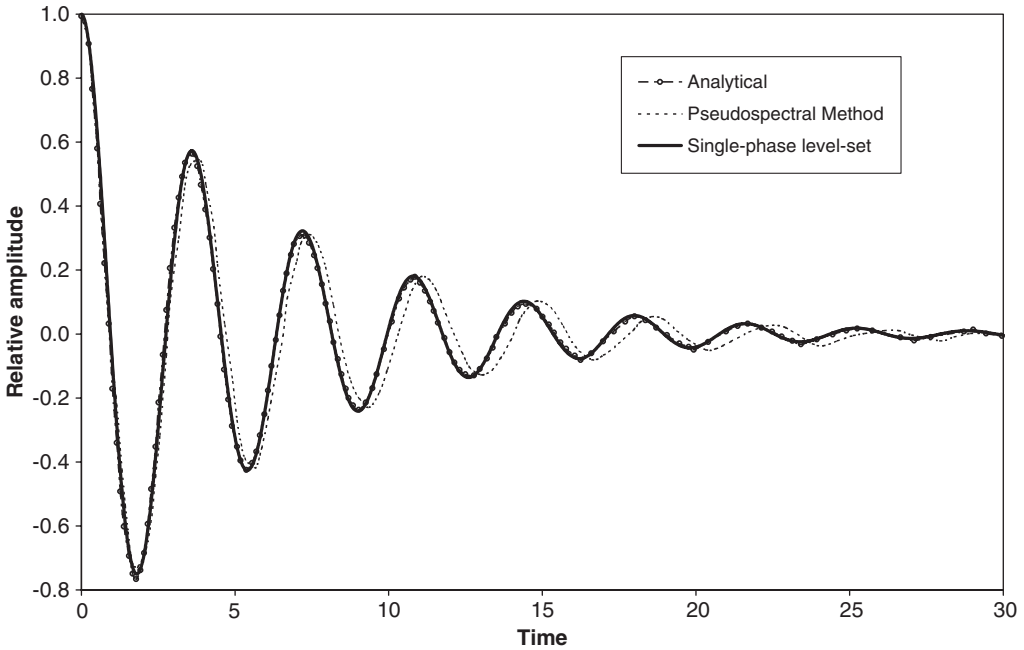

Figure 8. Wave amplitude evolution at the centre in a two-dimensional tank ($Re = 100$).

comparisons are made against the analytical and the pseudo-spectral method solutions [38] in Figure 8. It can be seen that the single-phase method does an excellent job in predicting both the amplitude and phase of the analytical solution outperforming the pseudo-spectral method predictions, probably due to the fact that the pseudo-spectral method uses linearized free surface boundary conditions.

The solution for a higher Reynolds number ($Re = 2000$) is shown in Figure 9, compared against the analytical solution reported in Reference [37]. At this Reynolds number, the single-phase level set method shows slight phase and amplitude differences. Notice, again, that the analytical solution neglects the nonlinear terms in the Navier–Stokes equations and uses a linearized free surface boundary condition, which might lead to error at this Reynolds number. This error, however, has not been quantified.

Forward speed diffraction in a surface ship

This problem is attractive as a benchmark because it involves considerable complications with respect to the previous two cases. In this problem, a ship is moving with constant speed in the presence of regular head waves, that is, the ship and the waves move in opposite directions. Wilson and Stern [7] and Rhee and Stern [8] have performed numerical simulations of this problem using surface-tracking methods and Cura Hochbaum and Vogt [20] have used the two-phase level set method.

DOI: 10.1002/fld

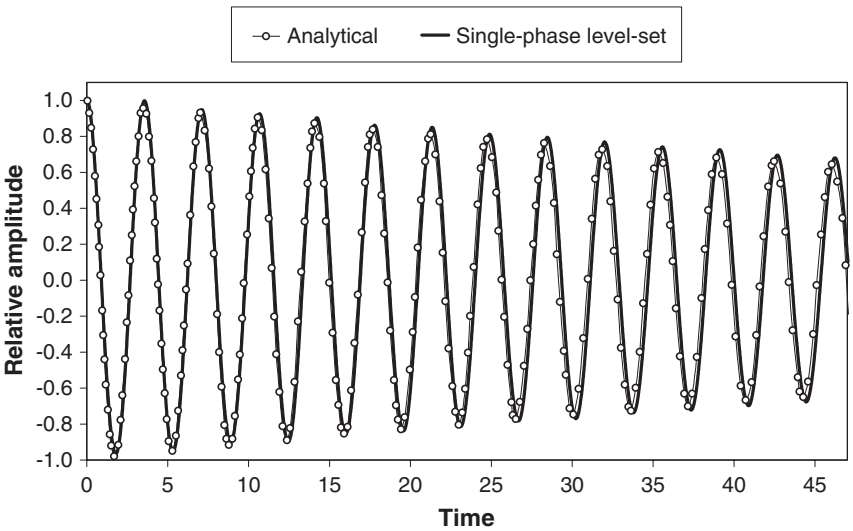

Figure 9. Wave amplitude evolution at the centre in a two-dimensional tank ($Re = 2000$).

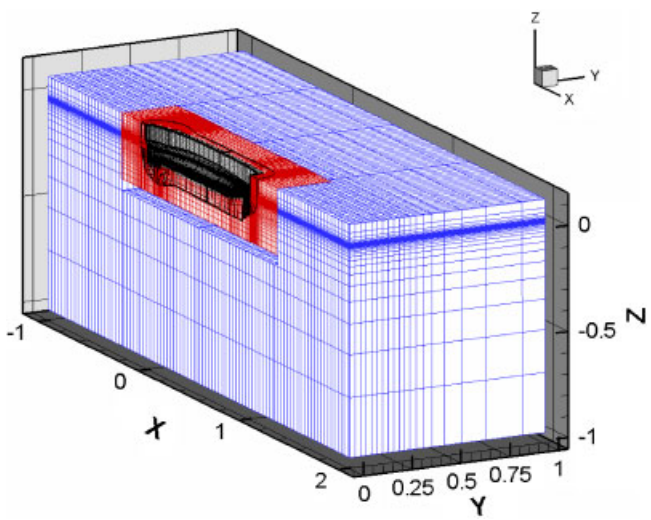

Figure 10. Multi-block overset grid for the wave diffraction problem. One every other point shown for clarity.

The test case chosen has been experimentally studied extensively for a DTMB 5512 model and unsteady free surface elevation, resistance, heave force and pitch moments [39, 40] and velocities [41] were measured. Though the authors measured forces and moments for a wide range of test conditions, free surface elevations and velocities were measured at medium Froude number

DOI: 10.1002/fld

($Fr = 0.28$), long wavelength ($\lambda = 1.5\,L$) and low wave steepness ($Ak = 0.025$). Thus, these conditions were selected for comparison with the numerical method. Since the model ship has a length $L = 3.048$ m, a Froude number $Fr = 0.28$ corresponds to $Re = 4.65 \times 10^6$.

In order to simulate the boundary layer turbulence using the k–ω model, a fine near wall discretization is necessary, with spacing around $10^{-6}L$. This makes the design of the computational grid difficult, since an orthogonal grid is convenient in the far-field to avoid deformation of the incoming wave. This problem can be avoided using a body-fitted grid for the boundary layer and an orthogonal grid for the far-field using overset grids with Chimera interpolation, though this introduces interpolations that could reduce the accuracy of the solution. The grid system is comprised by an eight block boundary layer body-fitted grid, a 16 block close-field orthogonal grid and an eight block far-field orthogonal grid, for a total of 32 blocks and approximately 2 000 000 grid points. The interpolation coefficients for the overset grids were generated using Pegasus [42]. The overall grid, shown in Figure 10, extends from $x = -1$ to $x = 2$, $y \in (0, 1)$, taking advantage of the symmetry of the problem about the centreplane $y = 0$, and $z \in (-1, 0.1)$, with the ship located between $x = 0$ to $x = 1$. Ghost cells were used for inter-block coupling inside each of the three main grid systems.

The initial conditions are set to those for a progressive wave, similar to the first example presented in this paper but in this case the wave amplitude is $A = 0.006$, in accordance to the experimental conditions. No numerical damping was used at the exit. The boundary conditions are summarized in the table below.

	ϕ	p	k	ω	U	V	W
Inlet ($x = -1$)	Equation (43)	Equation (42)	$k_{fs} = 10^{-7}$	$\omega_{fs} = 9$	Equation (40)	$V = 0$	Equation (41)
Exit ($x = 2$)	$\frac{\partial \phi}{\partial n} = 0$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial^2 U}{\partial n^2} = 0$	$\frac{\partial^2 V}{\partial n^2} = 0$	$\frac{\partial^2 W}{\partial n^2} = 0$
Far-field ($y = 1$)	$\frac{\partial \phi}{\partial n} = 0$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial U}{\partial n} = 0$	$\frac{\partial V}{\partial n} = 0$	$\frac{\partial W}{\partial n} = 0$
Far-field ($z = -1$)	$\frac{\partial \phi}{\partial n} = 1$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$U = 1$	$V = 0$	$W = 0$
Symmetry ($y = 0$)	$\frac{\partial \phi}{\partial n} = 0$	$\frac{\partial p}{\partial n} = 0$	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial U}{\partial n} = 0$	$V = 0$	$\frac{\partial W}{\partial n} = 0$
Far-field ($z = 0.1$)	$\frac{\partial \phi}{\partial n} = -1$	Not needed	$\frac{\partial k}{\partial n} = 0$	$\frac{\partial \omega}{\partial n} = 0$	$\frac{\partial U}{\partial n} = 0$	$\frac{\partial V}{\partial n} = 0$	$\frac{\partial W}{\partial n} = 0$
No slip (ship wall)	$\frac{\partial \phi}{\partial n} = 0$	Equation (37)	$k = 0$	$\omega = \frac{60}{Re \beta y^{+2}}$	$U = 0$	$V = 0$	$W = 0$

DOI: 10.1002/fld

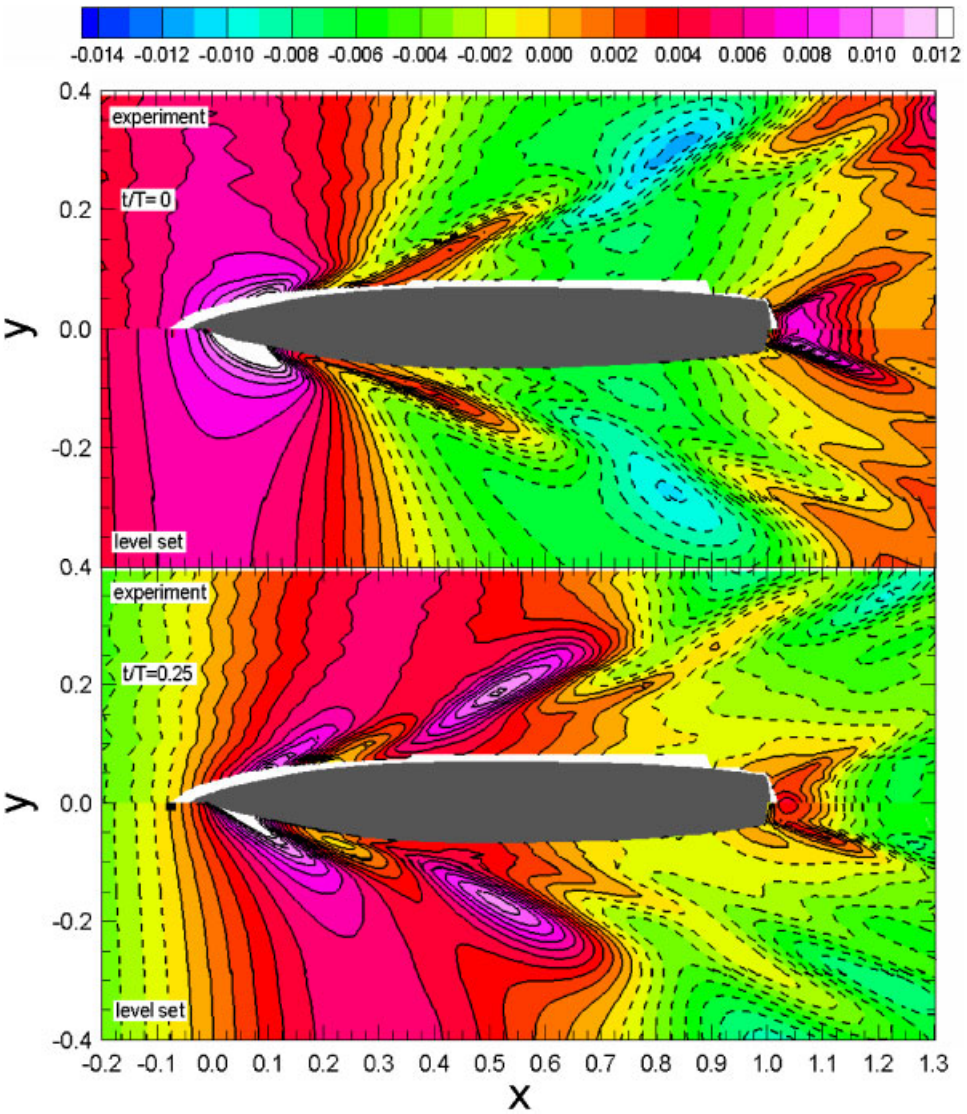

Figure 11. Free surface contours for $t/T = 0$ and $1/4$. Port: single-phase level set, starboard: experimental data.

The computation was started at $t = 0$ with sudden imposition of the boundary conditions, which causes an acceleration transient. The time step was chosen to be 0.00683, so that each wave period was discretized in 80 time steps. The initial transient is due to the time needed for the ship boundary layer to grow and for the Kelvin waves to develop, but an essentially periodic solution was achieved after about three non-dimensional time units, equivalent to the time to advance three ship lengths L. After the periodic solution was achieved, five more periods were run.

DOI: 10.1002/fld

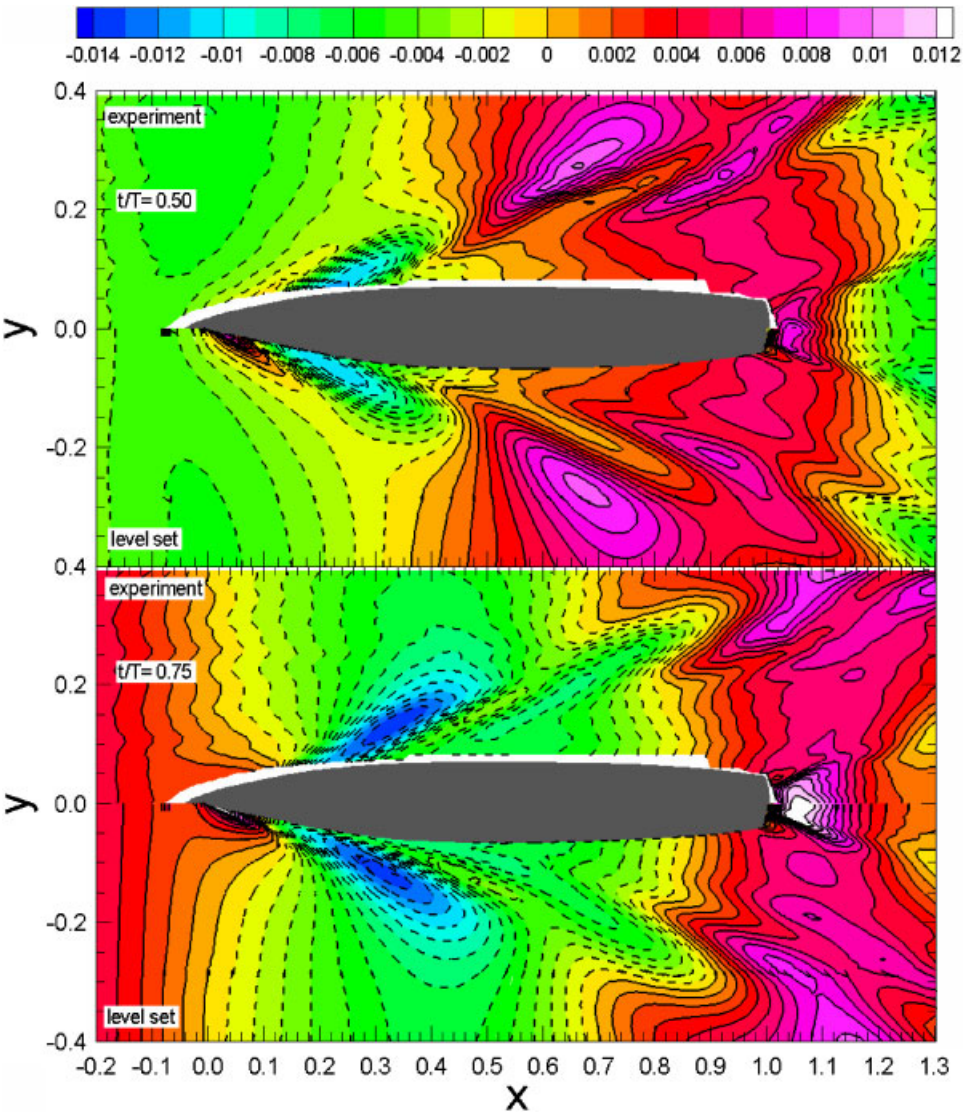

Figure 12. Free surface contours for $t/T = 1/2$ and $3/4$. Port: single-phase level set, starboard: experimental data.

In this paper, comparisons with free surface elevations and wake velocities are shown. Since the behaviour at the experimental conditions is linear, the free surface elevations were reconstructed and reported in terms of the zero and first Fourier harmonic amplitudes and first Fourier phase [39, 40]. Comparisons are made against quarter periods at $t/T = 0, 1/4, 1/2$ and $3/4$. The phase is set in such a way that at the beginning of the period T the crest of the wave is coincident with the bow of the ship, $x = 0$. Since the wavelength is $\lambda = 1.5\,L$, the far-field crest will be located at $x = 0.375, 0.75$ and 1.125 for $t/T = 1/4, 1/2$ and $3/4$, respectively.

DOI: 10.1002/fld

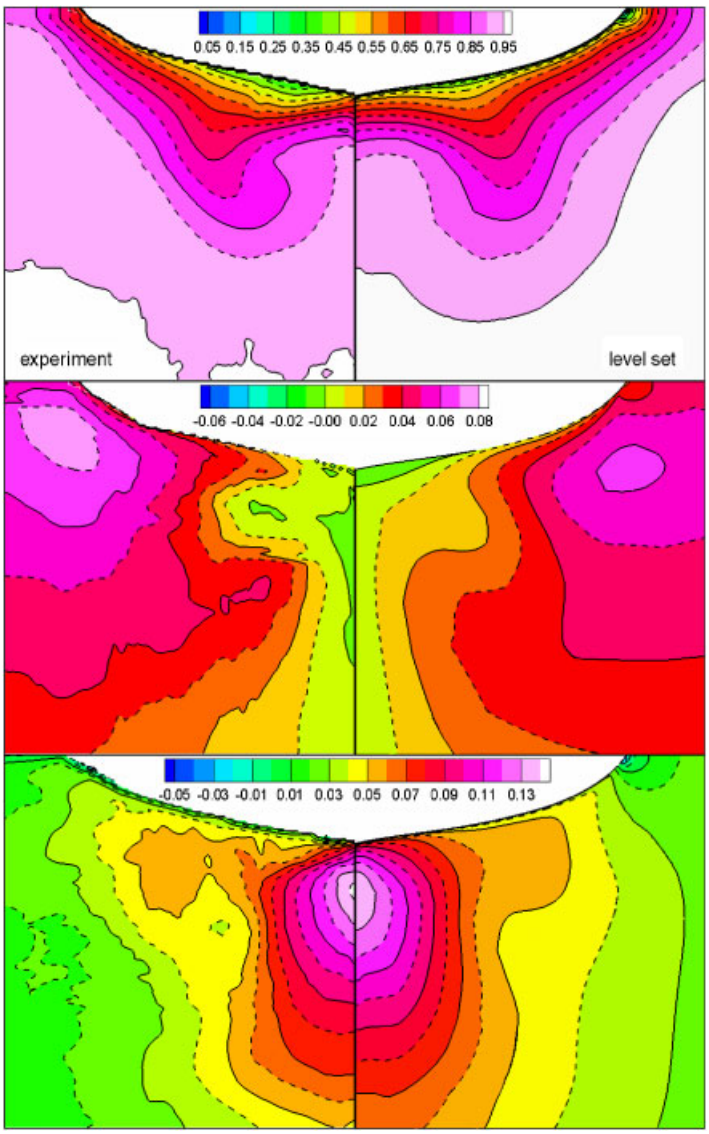

Figure 13. Velocity contours at the nominal wake plane for $t/T = 0$. U, V and W are shown on the upper, centre and lower figures, respectively. Experimental data (port) vs single-phase level set (starboard).

The free surface elevation predicted by the single-phase level set method is compared against the experimental data for the periodic state at $t/T = 0, 1/4, 1/2$ and $3/4$ in Figures 11 and 12. The single-phase level set results show an excellent agreement with the experimental data, capturing appropriately the Kelvin waves and the near-hull features of the free surface. The elevation gradients appear to be slightly smoothed, resulting in some under prediction of the crests and troughs on the Kelvin wave. Contrarily, the wave elevation on the stern is slightly overpredicted. One possible

DOI: 10.1002/fld

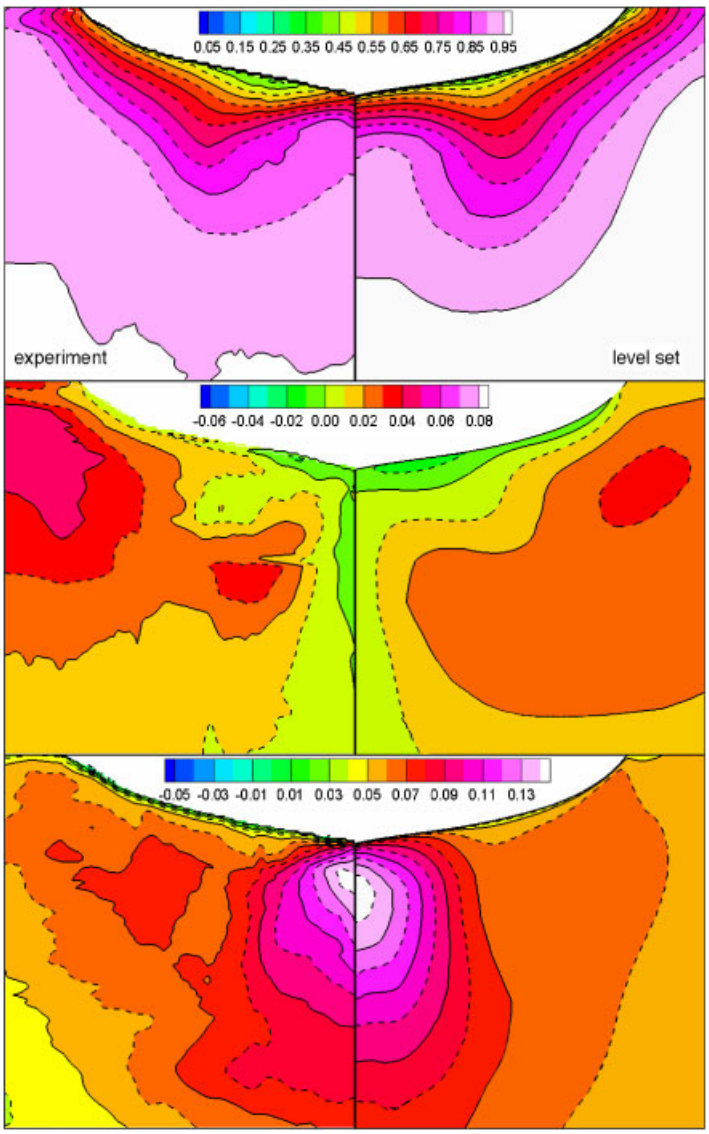

Figure 14. Velocity contours at the nominal wake plane for $t/T = 1/4$. U, V and W are shown on the upper, centre and lower figures, respectively. Experimental data (port) vs single-phase level set (starboard).

explanation for this trend is that the towing tank walls, located at $y = 0.5$, have not been included in the numerical simulation, and may have a small blocking effect that could modify the peaks and troughs.

PIV data of the velocity field at the nominal wake plane ($x/L = 0.935$) has been obtained by Longo *et al.* [41] using a phase-averaging technique. In Figures 13–16 the single-phase level set method results are compared against the experimental data. Though the results are reasonably good,

DOI: 10.1002/fld

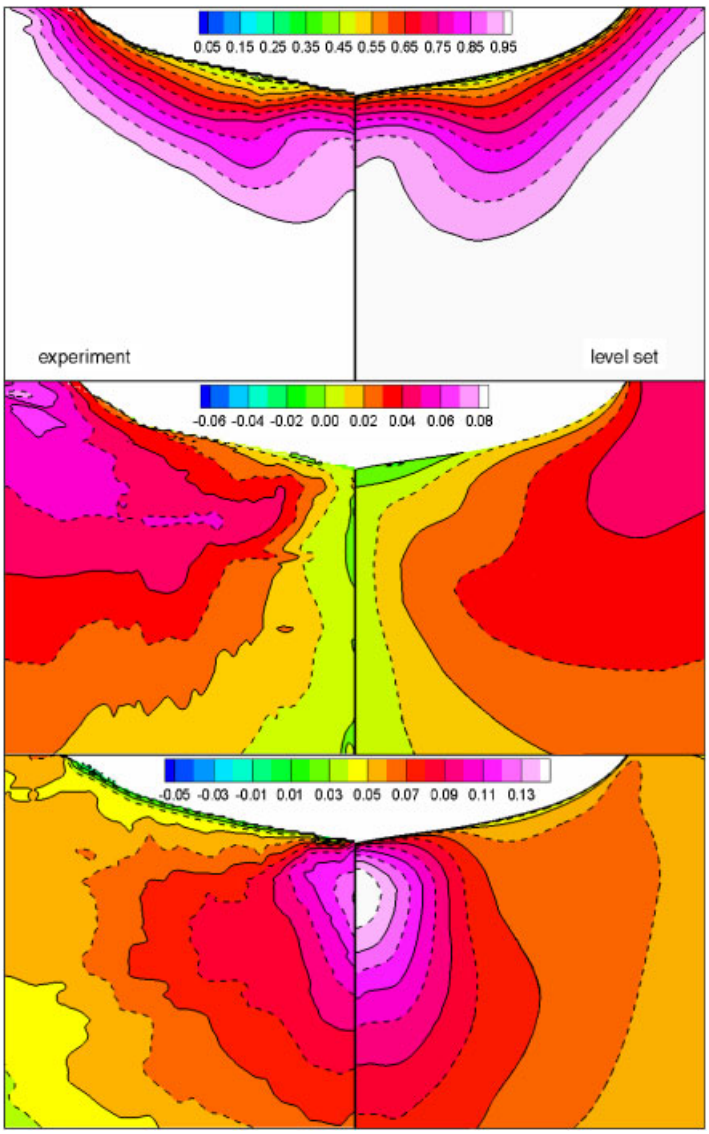

Figure 15. Velocity contours at the nominal wake plane for $t/T = 1/2$. U, V and W are shown on the upper, centre and lower figures, respectively. Experimental data (port) vs single-phase level set (starboard).

the boundary layer thickness is apparently to some extent overpredicted and the V and W velocities show that the vortex detached from the sonar dome has lost more strength in the computations than in the experiments. This trend could be due to the two-equation turbulence model used in the computations that cannot capture anisotropy for the Reynolds stresses.

DOI: 10.1002/fld

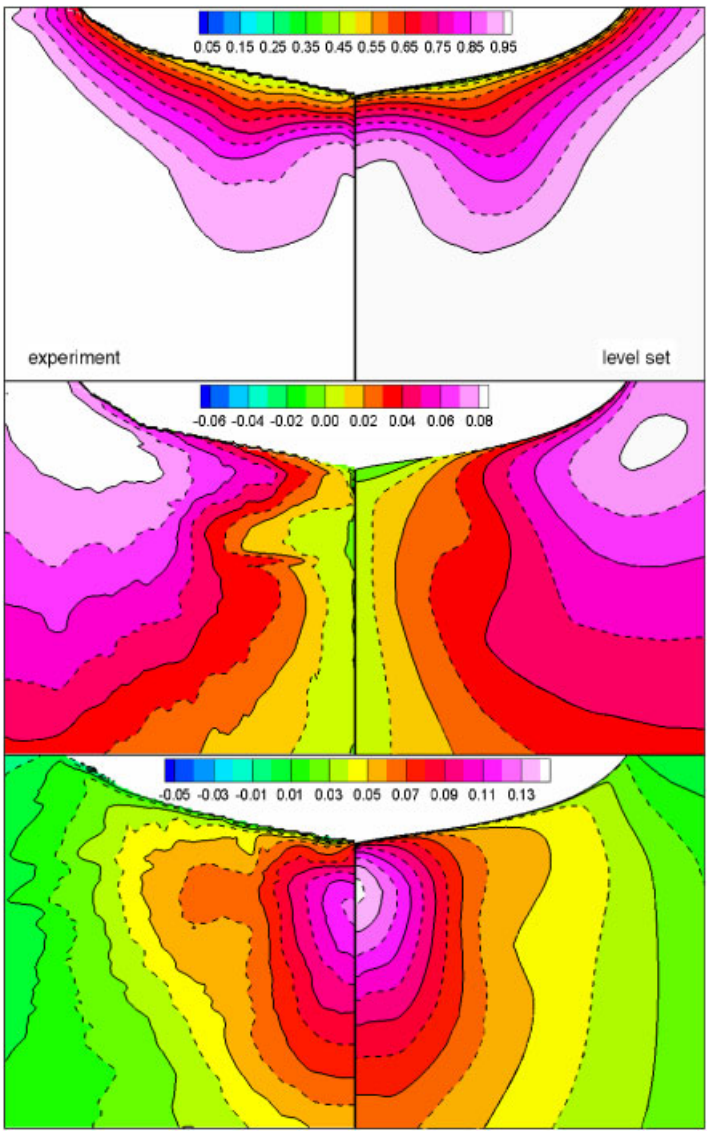

Figure 16. Velocity contours at the nominal wake plane for $t/T = 3/4$. U, V and W are shown on the upper, centre and lower figures, respectively. Experimental data (port) vs single-phase level set (starboard).

CONCLUSIONS

An unsteady single-phase level set method has been presented. The method relies on the level set function to detect the interface, and on velocity extensions and pressure interpolations to enforce the jump conditions at the interface. The computation of the time derivatives to properly evaluate the total derivatives is discussed in detail. The method is tested against three unsteady cases: an

DOI: 10.1002/fld

inviscid linear progressive wave, the viscous sloshing in a two-dimensional tank and the flow around a surface ship under regular head waves. In the three cases the method has performed very well compared to either analytical or experimental results.

The presented method has several advantages against the standard two-phase level set method, and of course some disadvantages. One of the advantages is that the computation takes place only in water, with potentially important savings by computing simpler equations in air. In addition, the computation is performed in a fluid with constant properties, avoiding the problem related with large density ratios in two-phase level set methods. Since the jump conditions are imposed explicitly, no transition zone appears. As for the shortcomings, the method does not solve the fluid equations in air, and therefore no problems in which air entrapment occurs can be solved. In addition, the stresses on the liquid caused by the air must be negligible or specified somehow.

In principle any problem that is not restricted by the limitations previously stated can be tackled with the unsteady single-phase level set method. In particular, this includes surface-piercing bodies with large-amplitude waves or motions, very steep waves, etc. Linear and nonlinear problems can be solved since the nonlinear terms are retained in the equations and jump conditions.

The capability of the single-phase level set method to solve complex transient free surface three-dimensional problems has been demonstrated. The future work includes an extensive analysis of the forward speed wave diffraction problem, including quantitative verification and validation and study of linear and nonlinear behaviour of forces and moments for larger Froude numbers and shorter wavelength. Extensions of interest that require 6DOF capability are the prediction of large-amplitude motions, including pitching and heaving, free and forced rolling, and manoeuvres.

ACKNOWLEDGEMENTS

This research was sponsored by the Office of Naval Research under Grant N00014-01-1-0073. Dr Patrick Purtell was the program manager.

REFERENCES

1. Sethian JA, Smereka P. Level set methods for fluid interfaces. *Annual Review of Fluid Mechanics* 2003; **35**:341–372.
2. Osher SJ, Sethian JA. Front propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics* 1988; **79**:12–49.
3. Osher SJ, Fedkiw RP. Level set methods: an overview and some recent results. *Journal of Computational Physics* 2001; **169**:463–502.
4. Kim MS, Lee WI. A new VOF-based numerical scheme for the simulation of fluid flow with free surface. Part I: New free surface-tracking algorithm and its verification. *International Journal for Numerical Methods in Fluids* 2003; **42**:765–790.
5. Wilson R, Stern F, Coleman H, Paterson E. Comprehensive approach to verification and validation of CFD simulations—Part 2: Application for RANS simulation of a cargo/container ship. *Journal of Fluids Engineering* (*ASME*) 2001; **123**:803–810.
6. Burg COE, Sreenivas K, Hyams DG. Unstructured nonlinear free surface simulations for the fully-appended DTMB model 5415 series hull including rotating propulsors. *24th ONR Symposium on Naval Hydrodynamics*, Fukuoka, Japan, 2002.
7. Wilson RV, Stern F. Unsteady CFD method for naval combatants in waves. *22nd ONR Symposium on Naval Hydrodynamics*, Washington, DC, U.S.A., 1998.
8. Rhee SH, Stern F. Unsteady RANS method for surface ship boundary layer and wake and wave field. *International Journal for Numerical Methods in Fluids* 2001; **37**:445–478.
9. Wilson RV, Carrica PM, Stern F. Unsteady RANS method for ship motions with application to roll for a surface combatant. *Computers and Fluids* 2006; **35**:501–524.

DOI: 10.1002/fld

10. Weymouth G, Wilson RV, Stern. RANS CFD predictions of pitch and heave ship motions in head seas. *Journal of Ship Research* 2005; **49**:80–97.
11. Larsson L, Stern F, Bertram V. Benchmarking of computational fluid dynamics for ship flows: the Gothenburg 2000 workshop. *Journal of Ship Research* 2003; **47**:63–81.
12. Hirt CW, Nichols BD. Volume of fluid (VOF) method for dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–221.
13. Unverdi SO, Tryggvason G. A front tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics* 1992; **100**:25–37.
14. Azcueta R, Muzaferija S, Peric M. Computation of breaking bow waves for a very full hull ship. *7th International Conference in Numerical Ship Hydrodynamics*, Nantes, France, 1999.
15. Sato Y, Miyata H, Sato T. CFD simulation of 3-dimensional motion of a ship in waves: application to an advancing ship in regular heading waves. *Journal of Marine in Science and Technology* 1999; **4**:108–116.
16. Nichols DS. Development of a free surface method utilizing an incompressible multi-phase algorithm to study the flow about surface ships and underwater vehicles. *Ph.D.* Thesis, Department of Aerospace Engineering, Mississippi State University, 2002.
17. Vogt M, Larsson L. Level set methods for predicting viscous free surface flows. *7th International Conference on Numerical Ship Hydrodynamics*, Nantes, France, 1999.
18. Cura Hochbaum A, Shumann C. Free surface viscous flow around ship models. *7th International Conference on Numerical Ship Hydrodynamics*, Nantes, France, 1999.
19. Cura Hochbaum A, Vogt M. Flow and resistance prediction for a container ship. *Proceeding of the Gothenburg 2000. A Workshop on CFD in Ship Hydrodynamics*, Gothenburg, Sweden, 2000.
20. Cura Hochbaum A, Vogt M. Towards the simulation of seakeeping and maneuvering based on the computation of the free surface viscous ship flow. *24th ONR Symposium on Naval Hydrodynamics*, Fukuoka, Japan, 2002.
21. Di Mascio A, Broglia R, Muscari R. Computation of free surface flows around ship hulls by a level set approach. *8th International Conference on Numerical Ship Hydrodynamics*, Busan, Korea, 2003.
22. Sussman M, Smereka P, Osher SJ. A level set approach to computing solutions to incompressible two-phase flow. *Journal of Computational Physics* 1994; **114**:146–159.
23. Drew DA, Passman SL. *Theory of Multicomponent Fluids*. Springer: New York, 1998; 89–91.
24. Marangoni C. Ueber die Ausbreitung der Tropfen einer Flüssigkeit auf der Oberfläche einer Anderen. *Annual Physical Chemistry* 1871; **143**:337–354.
25. Brackbill JU, Kothe DB, Zemach CA. A continuum method for modeling surface tension. *Journal of Computational Physics* 1992; **100**:335–354.
26. Fedkiw RP, Aslam T, Merriman B, Osher SJ. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *Journal of Computational Physics* 1999; **154**:393–427.
27. Kang M, Fedkiw RP, Liu LD. A boundary condition capturing method for multiphase incompressible flow. *Journal on Scientific Computing* 2000; **15**:323–360.
28. Menter FR. Two-equation eddy viscosity turbulence models for engineering applications. *AIAA Journal* 1994; **32**:1598–1605.
29. Sethian JA. *Level Set Methods*: *Evolving Interfaces in Geometry*, *Fluid Mechanics*, *Computer Vision and Material Science*. Cambridge University Press: Cambridge, 1996.
30. Adalsteinsson D, Sethian JA. The fast construction of extension velocities in level set methods. *Journal of Computational Physics* 1999; **148**:2–22.
31. Paterson EG, Wilson RV, Stern F. General-purpose parallel unsteady RANS ship hydrodynamics code: CFDShip-Iowa. *IIHR report 432*, Iowa Institute of Hydraulic Research, The University of Iowa, 2003.
32. Issa RI. Solution of the implicitly discretized fluid flow equations by operator splitting. *Journal of Computational Physics* 1985; **62**:40–65.
33. Thompson JF, Warsi ZUA, Mastin CW. *Numerical Grid Generation*. North-Holland: Amsterdam, 1992.
34. Balay S, Buschelman K, Gropp W, Kaushik D, Knepley M, Curfman L, Smith B, Zhang H. *PETSc User Manual*, ANL-95/11-Revision 2.1.5, Argonne National Laboratory, 2002.
35. Wilson R, Carrica PM, Stern F. Steady and unsteady single-phase level set method for large amplitude ship motions and maneuvering. *25th ONR Symposium on Naval Hydrodynamics*, New Foundland, Canada, 2004.
36. Newman JN. *Marine Hydrodynamics*. MIT Press: Cambridge, 1977; 240–241.
37. Wu GX, Taylor Eatock R, Greaves DM. The effect of viscosity on the transient free surface waves in a two-dimensional tank. *Journal of Engineering Mathematics* 2001; **40**:77–90.
38. Eatock Taylor R, Borthwick AGL, Chern MJ, Zhu G. Modelling unsteady viscous free surface flows. *International Maritime Research and Technology Conference*, Crete, Greece, October 2001.

DOI: 10.1002/fld

39. Gui L, Longo J, Metcalf B, Shao J, Stern F. Forces, moment and wave pattern for surface combatant in regular head waves. Part I: Measurement systems and uncertainty analysis. *Experiments in Fluids* 2001; **31**:674–680.
40. Gui L, Longo J, Metcalf B, Shao J, Stern F. Forces, Moment and wave pattern for surface combatant in regular head waves. Part I: Measurement results and discussion. *Experiments in Fluids* 2002; **32**:27–36.
41. Longo J, Shao J, Irvine M, Stern F. Phase-averaged PIV for surface combatant in regular head waves. *IIHR Report 447*, The University of Iowa, IA, November 2005.
42. Suhs NE, Rogers SE, Dietz WE. Pegasus 5: an automated pre-processor for overset-grid CFD. *AIAA Paper 2002-3186*, *32nd AIAA Fluid Dynamics Conference*, St. Louis, 2002.

DOI: 10.1002/fld