18th AIAA Computational Fluid Dynamics Conference
25 - 28 June 2007, Miami, FL

AIAA 2007-4578

A Sharp Interface Method for Two-Phase Flows Interacting with Moving Bodies

Jianming Yang* and Frederick Stern†

IIHR - Hydroscience & Engineering, University of Iowa, Iowa City, IA 52242-1585, USA

In this paper, we present a combined immersed-boundary/ghost-fluid sharp interface method for the simulations of three-dimensional two-phase flows interacting with moving bodies on fixed Cartesian grids. This work is based on a sharp interface immersed boundary method for fluid flows interacting with moving bodies[32] and a level set based ghost fluid method (GFM) for two-phase interface treatment.[21,15] Several cases with different scales are used to validate our numerical method. In the first case, the parasitic current in still fluid induced by numerical errors is chosen to show the accuracy of our ghost fluid implementation for very small scale problems. Then the small scale cases of water exit and entry of a circular cylinder are used to demonstrate how the immersed boundary approach works together with ghost fluid method. Finally, the case of waves generated by sliding mass is presented to illustrate the applicability of our method in large scale problems.

I. Introduction

Incompressible two-phase flows interacting with moving bodies are encountered in many scientific and engineering applications, such as naval hydrodynamics, ocean and coastal engineering, and process engineering like ink jet printers, etc. The computer simulations of these flows are extremely complicated as the intricate physical phenomena involved present significant challenges to the numerical methods. Breaking waves, turbulence, wave/jet impact, wave/jet structure interactions, just to name a few, are very common in these flows.

In general, numerical methods for two-phase flows with moving bodies can be categorized into three groups: moving reference frame methods, moving grid methods, and fixed grid methods. The moving reference frame methods, which solve the transformed Navier-Stokes equations with non-inertia force terms due to a reference frame following the body motion, are usually only applicable to problems with single solid body undergoing rigid body motion. In the moving grid methods, body-fitted grids are attached to the solid surfaces or even the interfaces between two different fluids, while the overall grids can be unstructured, block-structured, or overlapping grids. Although significant progress has been made in moving grid methods, the grid deformation and re-generation are still very time-consuming and prone to errors, especially, when multiple bodies and large-amplitude motions have to be considered. On the other hand, in the fixed grid methods, solid boundaries and phase interfaces can have unrestricted motions across the underlying grid lines, usually, which are not aligned with the solid-fluid and/or the fluid-fluid interfaces. And, in most case, Cartesian grids, which further simplify the gridding requirement, are used to cover the whole computational domain, although some techniques developed on fixed Cartesian grids have been applied to fixed curvilinear and unstructured grids.

Broadly speaking, there are two types of fixed Cartesian grid methods for moving boundary problems: the immersed boundary methods (IBM) and the cut-cell methods. The former, originated by Peskin,[25] have been applied to solid/fluid interactions and then two-phase flow problems by many authors. The basic idea of IBM is to model the effects of solid/elastic boundaries and fluid/fluid interfaces on the fluid flow by a set of body forces distributed over the nearby flow field of the immersed boundaries/interfaces. In this force distribution process, the immersed boundaries/interfaces are smeared over several grid cells. In the past

*Assistant Research Scientist, AIAA Member.
†Professor of Mechanical Engineering.

few years, several sharp interface methods using forcing or correction terms have been developed for solid-fluid and fluid-fluid interface problems, such as the direct forcing immersed boundary methods,[7] and the immersed interface methods,[19] among others. On the other hand, in the cut cell methods, the irregular cells produced by solid boundaries cutting through grid lines are treated in the same way as in a unstructured grid. In some versions of this type of methods the difficulties brought in by small cells were treated using cell merging technique,[30] in which, essentially, a body fitted grid is generated at each time step for moving boundary problems. Although the cut-cell methods are sharp interface methods, they are very difficult to apply to three dimensional problems due to the large number of special treatments required by the numerous different interface cells generated during the cutting and merging process.

In this paper, we present a combined immersed-boundary/ghost-fluid sharp interface method for the simulations of three-dimensional two-phase flows interacting with moving bodies on fixed Cartesian grids. This work is based on the direct forcing immersed boundary method for fluid flows interacting with moving bodies[2,32] and the level set based ghost fluid method (GFM) for two-phase interface treatment.[21,15] Our target applications are the complicated flow problems in many engineering fields, especially, naval hydrodynamics, with a wide spectrum of scales, such as wave-body interactions, wave-wave interactions, breaking waves, bubbles and droplets. In the following parts, we shall describe briefly our numerical method and demonstrate its capabilities by several test cases with different scales.

II. Mathematical Model

The level set formulation for the incompressible flows of two immiscible fluids separated by an interface given by Chang et al.[5] is used here. In this approach, one set of equations including the Navier-Stokes equations and the level set evolution equation for both fluids are solved. The discontinuous density and viscosity across the interface linked by the Heaviside step function, which equals zero with positive level set and one with negative level set, or verse visa.

II.A. Navier-Stokes Equations

The incompressible viscous flows of two immiscible fluids, e.g., air and water, are governed by the Navier-Stokes equations:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{1}{\rho} \nabla \cdot (-p\mathbf{I} + \mathbf{T}) + \mathbf{g}, \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2}$$

where t is the time, $\mathbf{u}$ is the velocity vector, p is the pressure, $\mathbf{I}$ is the unit tensor, ρ is the density, $\mathbf{g}$ represents the gravity, and $\mathbf{T}$ is the viscous stress tensor defined as

$$\mathbf{T} = 2\mu\mathbf{S} = 2\mu \left[\frac{1}{2} \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \right] = \mu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right), \tag{3}$$

with μ the dynamic viscosity, $\mathbf{S}$ the strain rate tensor and T the transpose operator.

Since the fluid properties are discontinuous across the interface, which is a function of time and space, density and viscosity are also functions of time and space and only known with given interface position. Their definitions will be deferred to Sec. II.D after the introduction of interface representation using level set.

II.B. Interface Jump Conditions

The velocity across the interface Γ is continuous, as the fluids are viscous and no phase change is considered here:

$$[\mathbf{u}] = 0, \tag{4}$$

and the jump condition for stress is

$$-p + \left[\mathbf{n} \cdot \left(\mu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \right) \cdot \mathbf{n} \right] = \sigma\kappa, \tag{5}$$

where $[\ \cdot\ ]$ indicates the jump at the interface, i.e., $f_L^I - f_G^I$ for a variable f with superscript I denotes interface, $\mathbf{n}$ is the unit vector normal to the interface, σ is the coefficient of surface tension, and κ is the local curvature of the interface.

The gravity term can be removed from Eq. (1) by incorporating the gravity into the jump condition as

$$-p_d + \left[\mathbf{n} \cdot \left(\mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\right) \cdot \mathbf{n}\right] = \sigma\kappa + [\rho]\mathbf{X} \cdot \mathbf{g}, \tag{6}$$

where p_d represents dynamic pressure (for simplicity, p is used hereafter), $[\rho]$ is the density jump at the interface, and $\mathbf{X}$ is the position vector normal to the reference plane of zero hydrostatic pressure.

II.C. Interface Representation

Defining the interface Γ as the zero level set of a signed distance function, ϕ, or the level set function, the position of the interface can be tracked by solving the level set evolution equation

$$\frac{\partial\phi}{\partial t} + \mathbf{u} \cdot \nabla\phi = 0. \tag{7}$$

To keep ϕ as a signed distance function in the course of evolution, we iterate the reinitialization equation for the level set function:[28]

$$\frac{\partial\phi}{\partial\tau} + S(\phi_o)\left(|\nabla\phi| - 1\right) = 0, \tag{8}$$

where τ is the pseudo time and $S(\phi_o)$ is the numerically smeared-out sign function

$$S(\phi_o) = \frac{\phi_o}{\sqrt{\phi_o^2 + h^2}}, \tag{9}$$

with ϕ_o the initial values of ϕ and h a small amount, usually the grid cell size, to smear out the sign function.

Since the level set function is a signed distance function, the interface normal and mean curvature can be readily calculated by applying standard finite difference to the level set function:

$$\mathbf{N} = \frac{\nabla\phi}{|\nabla\phi|}, \tag{10}$$

and

$$\kappa = \nabla \cdot \mathbf{N} = \nabla \cdot \frac{\nabla\phi}{|\nabla\phi|}, \tag{11}$$

respectively.

II.D. Fluid Properties

With the level set function defined, the fluid properties, such as density and viscosity, are given by the following equations:

$$\begin{aligned} \rho &= \rho_G + (\rho_L - \rho_G)H(\phi), \\ \mu &= \mu_G + (\mu_L - \mu_G)H(\phi), \end{aligned} \tag{12}$$

where the subscripts G and L represent gas and liquid phase, respectively, and the Heaviside function is defined as

$$H(\phi) = \begin{cases} 1 & \text{if} \quad \phi \geq 0 \\ 0 & \text{if} \quad \phi < 0 \end{cases} \tag{13}$$

In this paper, the viscosity is smoothed over a transition region across the interface as

$$\mu = \mu_G + (\mu_L - \mu_G)H_\varepsilon(\phi), \tag{14}$$

using the smoothed Heaviside function[28]

$$H_\varepsilon(\phi) = \begin{cases} 1 & \text{if} \quad \phi > \varepsilon \\ \frac{1}{2}\left[1 + \frac{\phi}{\varepsilon} + \frac{1}{\pi}\sin\left(\frac{\pi\phi}{\varepsilon}\right)\right] & \text{if} \quad |\phi| \leq \varepsilon \\ 0 & \text{if} \quad \phi < -\varepsilon \end{cases} \tag{15}$$

Notice with a continuous viscosity and velocity field, the stress jump conditions Eq. (5) reduce to

$$[p] = p_L^I - p_G^I = -\sigma\kappa - [\rho]\mathbf{X} \cdot \mathbf{g}. \tag{16}$$

III. Numerical Method

III.A. Navier-Stokes Solver

The finite difference method is used to discretize the Navier-Stokes equations on a non-uniform staggered Cartesian grid, in which the velocity components u, v, and w are defined at centers of cell faces in the x, y, and z directions, respectively, and all other variables, i.e., p, ϕ, ρ, μ, and ν_t are defined at cell centers. A semi-implicit time-advancement scheme is adopted to integrate the momentum equations with the second-order Crank-Nicolson scheme for the diagonal viscous terms and the second-order Adams-Bashforth scheme for the convective terms and other viscous terms. A four-step fractional-step method[6] is employed for velocity-pressure coupling, in which a pressure Poisson equation is solved to enforce the continuity equation:

- Predictor:
$$\frac{\hat{u}_i - u_i^n}{\Delta t} = \frac{1}{2}\left[3A_i^n - A_i^{n-1}\right] + \frac{1}{2}\left[C_i^{n+1} + C_i^n\right] - \mathrm{Grad}_i(p^n), \tag{17}$$

- First Corrector:
$$\frac{u_i^* - \hat{u}_i}{\Delta t} = \mathrm{Grad}_i(p^n) \tag{18}$$

- Pressure Poisson Equation:
$$\frac{\partial}{\partial x_i}\mathrm{Grad}_i(p^{n+1}) = \frac{1}{\Delta t}\frac{\partial u_i^*}{\partial x_i} \tag{19}$$

- Second Corrector:
$$\frac{u_i^{n+1} - u_i^*}{\Delta t} = -\mathrm{Grad}_i(p^{n+1}) \tag{20}$$

where superscript n denotes time step, subscript $i = 1, 2, 3$ represents i-coordinate, A and C denote terms treated by the Adams-Bashforth and Crank-Nicolson schemes, $\hat{u}$ and u^* are the first and second intermediate velocities, respectively. $\mathrm{Grad}_i(p)$ is a pressure gradient term defined at the center of cell face (collocated with velocity components) with the jump conditions incorporated in (for similar definition, see Ref. 29).

The ghost fluid method for incompressible two-phase flows[15] is adopted here to handle the jump conditions at the interface. Unfortunately, the viscous terms were treated explicitly in the original paper,[15] which imposed a sever time step limitation to the simulation. And due to the coupled manner of the interface jump conditions, an implicit treatment of viscous terms requires the solution of the entire coupled system.[12] Several authors chose to regularize the viscosity with a smoothed Heaviside function across the interface.[31,9] In this paper, to avoid the explicit treatment of viscous terms, we also choose to smear the discontinuous viscosity at the interface with a smoothed Heaviside function as discussed in Sec. II.D. However, the discontinuous or singular properties of other quantities, i.e., density and surface tension (if considered in a test case), at the interface are preserved.

To handle the jump conditions, for example, in the x direction, we can have

$$\mathrm{Grad}(p)_{i+1/2} = \frac{1}{\rho_{i+1/2}}\frac{(p_{i+1} - p_i) + (\sigma\kappa^I + [\rho](x^I g_x + y^I g_y + z^I g_z))\,(H_{i+1} - H_i)}{\Delta x} \tag{21}$$

where H is the Heaviside function, the superscript I denotes interface. The cell face density is defined as

$$\rho_{i+1/2} = \rho_L\theta_{i+1/2} + \rho_G\left(1 - \theta_{i+1/2}\right), \tag{22}$$

with

$$\theta_{i+1/2} = \begin{cases} 1 & \phi_i \geq 0 \quad \text{and} \quad \phi_{i+1} \geq 0 \\ 0 & \phi_i < 0 \quad \text{and} \quad \phi_{i+1} < 0 \\ \frac{\phi_i}{|\phi_i| + |\phi_{i+1}|} & \phi_i \geq 0 \quad \text{and} \quad \phi_{i+1} < 0 \\ \frac{\phi_{i+1}}{|\phi_i| + |\phi_{i+1}|} & \phi_i < 0 \quad \text{and} \quad \phi_{i+1} \geq 0 \end{cases} \tag{23}$$

Note that the surface tension as a singular force and gravity don't appear in the momentum equations explicitly since it enters the system through the pressure jump condition. Also, Eq. (19) is rearranged to move the pressure jump condition into the right hand side.[21]

The above four-step time advancement scheme doesn't require special treatment of boundary conditions for the intermediate velocity to obtain an overall second order splitting. The pressure correction methods are not applicable here because the density and pressure are treated in a sharp interface manner and the following equality cannot be applied directly

$$\frac{1}{\rho}\nabla p^{n+1} = \frac{1}{\rho}\nabla p^{n} + \frac{1}{\rho}\nabla \psi, \tag{24}$$

with ψ for the pressure correction or increment.

In Eq. (17) the convective terms are discretized using a third-order QUICK scheme[18] and higher-order WENO schemes[14] are available. All other terms are discretized with the standard second-order central difference scheme. And Eq. (17) is approximated with the approximate factorization method[4] and the resulting tridiagonal linear equations are solved with the parallel tridiagonal system solver given in Ref. 23. The parallelization is done via a domain decomposition technique using the MPI library. The pressure Poisson equation is solved using a multigrid-preconditioned Krylov subspace solver from the PETSc library.[3] In general, this is the most expensive part of the whole algorithm.

III.B. Level Set Solver

The level set function is evolved using an Eulerian convection equation and reinitialized using a similar equation given by Sussman et al.[28] to keep the level set as a distance function. Here we adopt the local (narrow band) level set method by Peng et al.[24] for the level set and the reinitialization equations. The level set equation and the reinitialization equation are solved using third-order TVD Runge-Kutta scheme[27] for time advancement and fifth-order HJ WENO scheme[13] for spatial discretization. The additional solve of these equations does not pose a significant overhead as they are solved in a narrow band about several grid-cell wide.

III.C. Immersed Boundary Treatment

We use the sharp interface immersed boundary formulation for moving boundaries by Yang and Balaras[32] to treat the immersed boundaries/bodies in a non-uniform Cartesian grid. In this approach, the grid generation for complex geometries is trivial as the requirement that the grid points coincide with the boundary, which is imperative for body-fitted methods, is relaxed; while the solution near the immersed boundary is reconstructed using momentum forcing in a sharp interface manner. The detailed procedure is given in Refs. 1, 32 and summarized here.

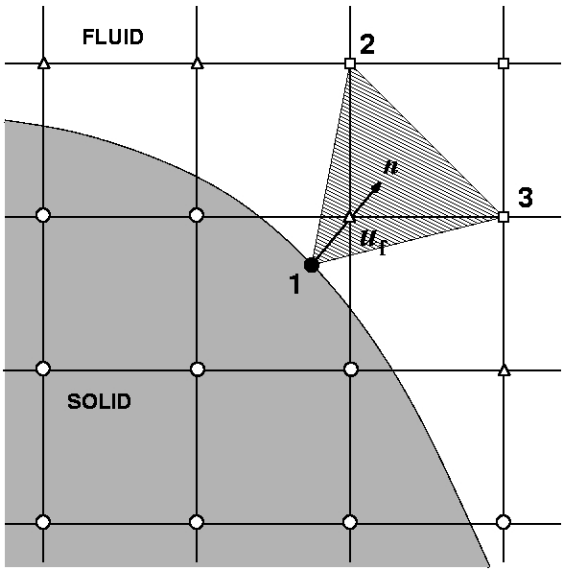


Figure 1: Grid/interface relation and Interpolation stencil for u_f (point 1, 2, and 3): ◯ *solid points*; □ *fluid points*; △ *forcing points*.

The first step is to establish the grid-interface relation with a given immersed boundary description, such as parametrized curve/surface or triangulation. In this step all Cartesian grid nodes are split into the three categories shown in Fig. 1: (1): *fluid-points*, which are points in the fluid phase; (2) *forcing points*, which are grid points in the fluid phase with one or more neighboring points in the solid phase; (3) *solid-points*, which are points in the solid phase. The Navier-Stokes solver described in the previous section is applied on all points of the Eulerian grid as if the fluid/solid interface is not present. The effect of immersed boundary on the flow is introduced through the discrete forcing function, f_i. It is computed only at the forcing points by substituting $\hat{u}_i^n$ with u_f in Eq. (17) and solving for f_i:

$$f_i^n = \frac{u_f - u_i^{n-1}}{\Delta t} - \mathrm{RHS}_i^n, \tag{25}$$

where RHS_i^n denotes the right-hand-side of Eq. (17). In the special case where the interface and the forcing point coincide, u_f is simply the local velocity on the rigid body, and therefore the boundary condition one wishes to enforce. In the general case, however, the points on the Eulerian grid and the interface almost never coincide and u_f has to be computed using some interpolation strategy. An example is shown in Fig. 1, where the velocity at the forcing point is computed by means of linear interpolation that involves the projection of the forcing point on the interface (point 1 in Fig 1) and two points in the fluid phase (points 2 and 3 in Fig 1). For the semi-implicit time advancement scheme used here, a provisional step is applied with all terms by the Crank-Nicolson scheme treated using the forward Euler scheme. Then the forcing function in Eq. (25) can be evaluated straightforward.[16,2] The above procedure has been extensively tested for a variety of laminar and turbulent flow problems involving stationary and moving immersed boundaries with results in excellent agreement with reference computations and experiments.[1,2,32]

III.D. Contact Angle Boundary Condition

The contact angle boundary condition on the immersed boundary is implemented by following an interpolation strategy similar to the idea discussed above. As shown in Fig. 2, the contact angle boundary condition can be written as

$$\mathbf{n}_{ls} \cdot \mathbf{n}_{ib} = -cos(\alpha), \tag{26}$$

where $\mathbf{n}_{ls}$ is the normal of level set function at the conact point, which is calculated using Eq. (10), $\mathbf{n}_{ib}$ is the outward normal of the immersed boundary at the conact point, which is well-defined as a basic element of our immersed boundary method, and α is the contact angle between the interface and the immersed wall.

Substituting $\mathbf{n}_{ls}$ by Eq. (10) in Eq. (26), we can obtain

$$\frac{\nabla\phi}{|\nabla\phi|} \cdot \mathbf{n}_{ib} = \frac{1}{|\nabla\phi|}\left(\frac{\partial\phi}{\partial n}\right)_{ib} = -cos(\alpha), \tag{27}$$

or,

$$\left(\frac{\partial\phi}{\partial n}\right)_{ib} = -cos(\alpha)|\nabla\phi| = -cos(\alpha), \tag{28}$$

as ϕ is a signed distance function and $|\nabla\phi| = 1$.

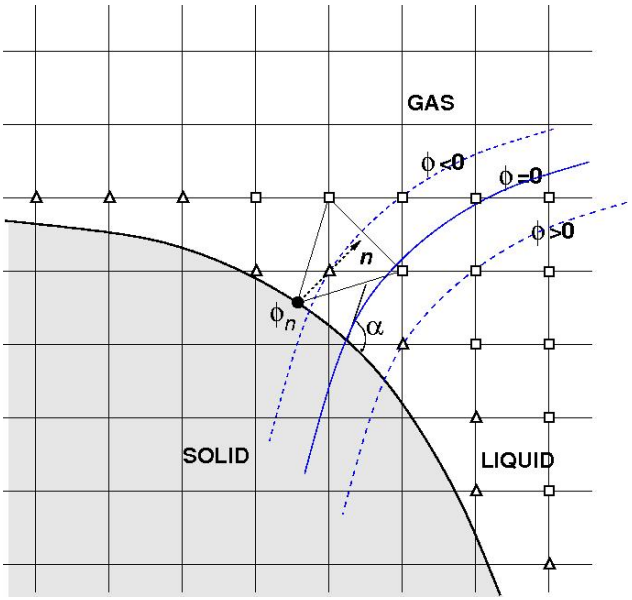


Figure 2: Contact angle boundary condition.

Therefore, with the normal gradient of level set available, the contact angle boundary condition can be readily specified. In this paper, a fixed contact angle $\alpha = 90^o$ is used for all cases.

III.E. Time Step Restriction

The time step Δt is restricted by the CFL condition, gravity, and surface tension. Follow Ref. 15 and choose a CFL restriction of 0.5, we have

$$\Delta t \leq 0.5 \left(\frac{C_{cfl} + \sqrt{(C_{cfl})^2 + 4\,(G_{cfl})^2 + 4\,(S_{cfl})^2}}{2} \right)^{-1}, \tag{29}$$

with the convective time step restriction

$$C_{cfl} = \max\left(\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} + \frac{|w|}{\Delta z} \right), \tag{30}$$

the time step restriction due to gravity

$$G_{cfl} = \sqrt{\frac{|g_x|}{\Delta x} + \frac{|g_y|}{\Delta y} + \frac{|g_z|}{\Delta z}}, \tag{31}$$

and the time step restriction due to surface tension

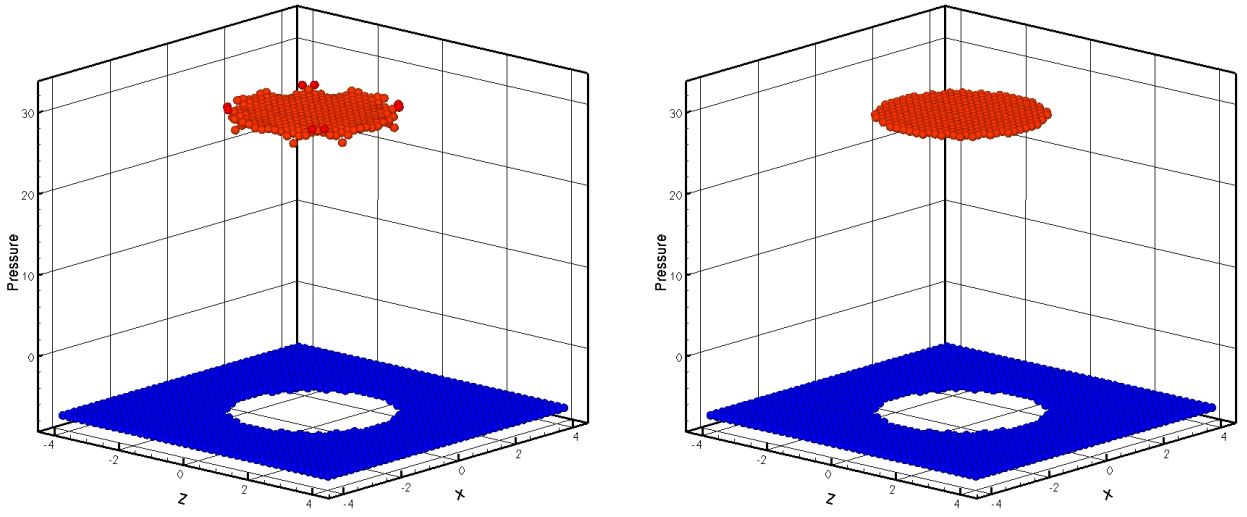
$$S_{cfl} = \sqrt{\frac{\sigma|\kappa|}{\rho_G \left(\min(\Delta x, \Delta y, \Delta z)\right)^2}}. \tag{32}$$

IV. Results

In this part, several cases with different scales are used to validate our numerical method. In the first case, a standard test case for surface tension model, parasitic current in still fluid induced by numerical errors, is chosen to show the accuracy of our ghost fluid implementation for very small scale problems. Then the small scale cases of water exit and entry of a circular cylinder are used to demonstrate how the immersed boundary approach works together with ghost fluid method. Finally, the case of waves generated by sliding mass is presented to illustrate the applicability of our method in large scale problems.

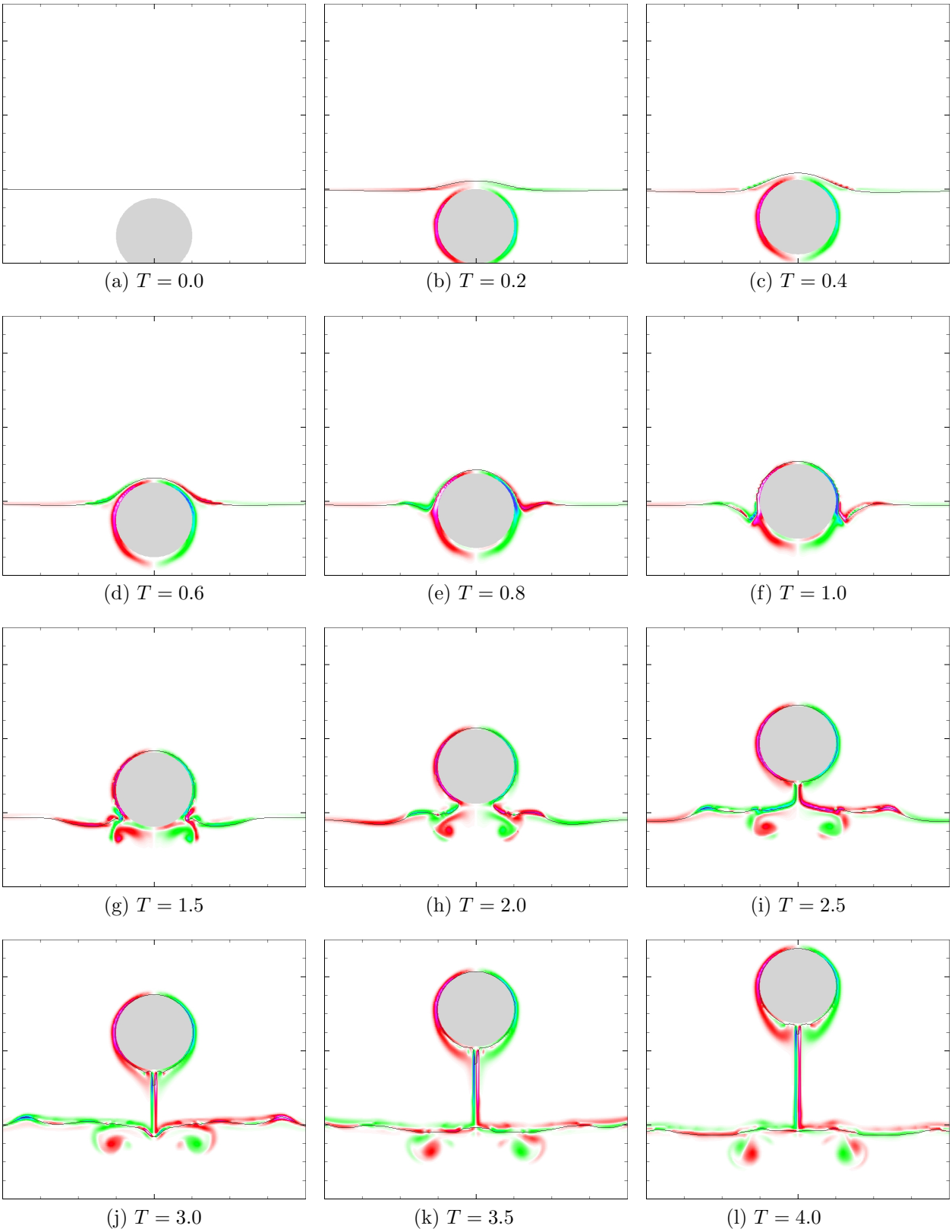
Table 1: Maximum velocity $|u|_{max}$ after first time step for a 2D viscous static drop

ρ_{in}/ρ_{out}	curvature from level set function	exact curvature
10	2.14×10^{-7}	5.73×10^{-16}
10^3	2.32×10^{-7}	7.24×10^{-16}
10^5	2.33×10^{-7}	1.78×10^{-15}



(a) Curvature from level set function (b) Exact curvature

Figure 3: Pressure after 1000 time steps with $\Delta t = 10^{-5}$ for the viscous static drop using curvatures: (a) calculated from level set function; (b) exact.



(a) $T = 0.0$ (b) $T = 0.2$ (c) $T = 0.4$

(d) $T = 0.6$ (e) $T = 0.8$ (f) $T = 1.0$

(g) $T = 1.5$ (h) $T = 2.0$ (i) $T = 2.5$

(j) $T = 3.0$ (k) $T = 3.5$ (l) $T = 4.0$

Figure 4: Water exit problem: The free surface position (solid black line) and vorticity contours ($-10 \sim 10$ with intervals of 0.1).

IV.A. Parasitic Current

In this case, we test our ghost fluid implementation by performing a test of the parasitic current produced by surface tension model. The setup is the same as the one listed in Table 4 of Ref. 8. A 2D computational domain of 8×8 is filled with a light fluid and a static drop of radius $r = 2$ is centered in it. A uniform 40×40 grid and a constant time step $\Delta t = 10^{-6}$ is used. The density of the drop $\rho_{in} = 1$ and the density of the ambient fluid $\rho_{out} = 10^{-1}, 10^{-3}, 10^{-5}$ in three different cases, respectively. The fluid viscosity $\mu_{in} = 10^{-2}$ and $\mu_{out} = 10^{-3}$. The surface tension coefficient $\sigma = 73$.

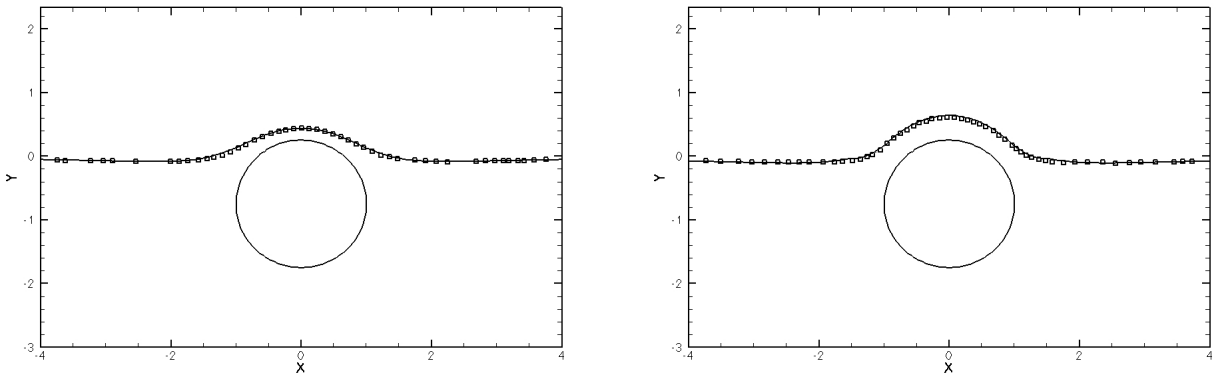
Tab. 1 shows the maximum velocity after first time step for several different density ratio and using the exact curvature and curvature evaluated with level set function. With exact curvature, our method produces a maximum velocity within machine accuracy. And those using level set function to calculate curvature are in the same magnitude of the results in Ref. 8.

Fig. 3 shows the computed pressure after 1000 time steps with $\Delta t = 10^{-5}$. As expected, the pressure has a sharp jump across the interface due to the surface tension. Comparing with the one using exact curvature Fig. 3b, level set calculated curvatures (Fig. 3a) contains errors and produce non-uniform pressure distribution, especially, inside the drop.

IV.B. Water Exit

Here we use the same setup conditions as those in Refs. 11, 20: a circular cylinder of radius $r = 1$ is placed in calm water and the distance of its center to free surface is $H = -1.25$. The gravity is set to be $g = 1$ and the cylinder is given a constant upward velocity $V = 0.39$ at $T = Vt/H = 0$ with t the time in the calculation. A uniform grid size of $\Delta h = 0.05r$, which is the same as that in Ref. 20, is used to cover the path of the cylinder.

The snapshots of the interactions between cylinder and the two-phase interface are shown in Fig. 4. As the cylinder moves upward, two shear layers are developing along the left and right sides of the cylinder. Two vortices shed from these two shear layers interact with the free surface and form two dipoles beneath the free surface. Waves are generated in the exiting process and propagate toward two sides of the domain.



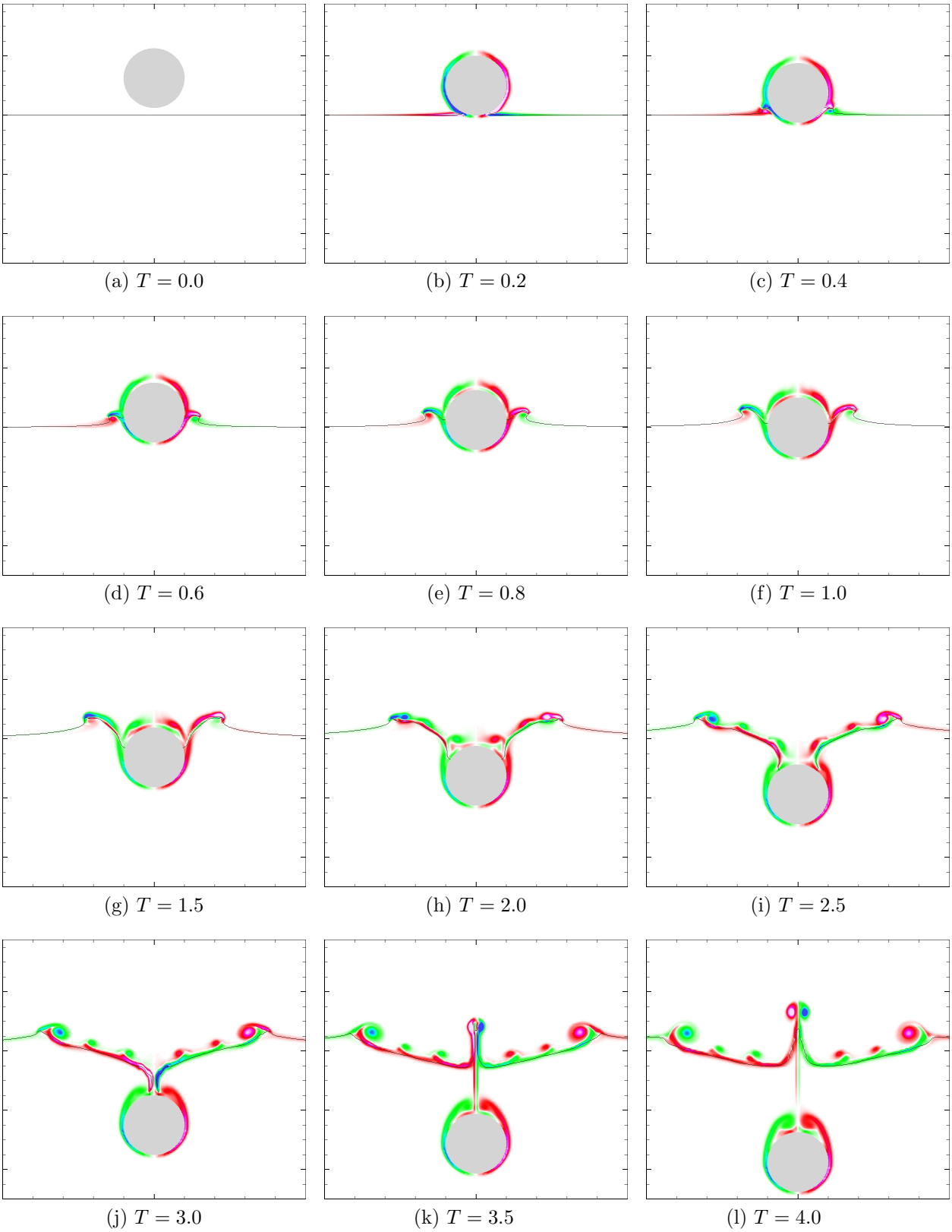
(a) $T = 0.4$ (b) $T = 0.6$

Figure 5: Comparison of free surface profiles with boundary element simulation in Ref. 11(□).

To further validate our numerical method, the comparison between our results with the boundary element simulation of Ref. 11 is shown in Fig. 5. Very good agreement is observed.

IV.C. Water Entry

The water impact/entry problems have been studied in Refs. 10, 17, 20, 26, among others. We again use the same parameters as those in Ref. 20. This time, the cylinder starts its downward motion from a height of $H = 1.25$ to the calm water surface with a constant velocity $V = -1$.



(a) $T = 0.0$ (b) $T = 0.2$ (c) $T = 0.4$

(d) $T = 0.6$ (e) $T = 0.8$ (f) $T = 1.0$

(g) $T = 1.5$ (h) $T = 2.0$ (i) $T = 2.5$

(j) $T = 3.0$ (k) $T = 3.5$ (l) $T = 4.0$

Figure 6: Water impact and entry problem: The free surface position (solid black line) and vorticity contours ($-20 \sim 20$ with intervals of 0.2).

A series of snapshots are shown in Fig. 6. As the cylinder impacts the free surface, two jets are generated along two sides of the cylinder. Just like the water exit case, the vortices shed from the shear layers interact with the free surface. It is very interesting to observe a series of these vortices attached to the free surface and following the free surface motion.

The results are very close to those given in Ref. 20 up to $T = 2$. However, we are able to capture the entrainment of air as the two-phase model is used here and give a higher water jet at a later time than that in Ref. 20, in which single phase free surface flow is solved.

IV.D. Waves Generated by 3D Sliding Mass

In Ref. 22 the waves generated by 3D sliding masses were studied using both large-scale wave tank experiments and numerical methods. Here we simulate the case of a subaerial wedge slide as detailed in Ref. 22. The setup is the same as in the reference. However, one coordinate direction is chosen to be parallel to the slope in order to obtain better grid resolution close to the wall. The computational domain in our simulation is $6.5m \times 3.7m \times 3m$ in the direction along the slope, the spanwise direction, and the direction normal to the slope, respectively. A simulation with a uniform grid size of $0.025m$ near the initial position of the landslide has been carried out. The motion of the sliding wedge is given by laboratory measurement.

The snapshots of free surface profile are shown in Fig. 7. At the beginning, the sliding wedge enters the water slowly and pulls away the water in front of it. As the wedge reaches its full speed, the water around it follows the downward motion of the wedge and big cave appears on top of the wedge. The wedge keeps sinking and water rushes into the cave. A big jet is generated near the shoreline when the left and right flushing waves merge at the centerline. This jet then collapses and strong waves propagate outward.

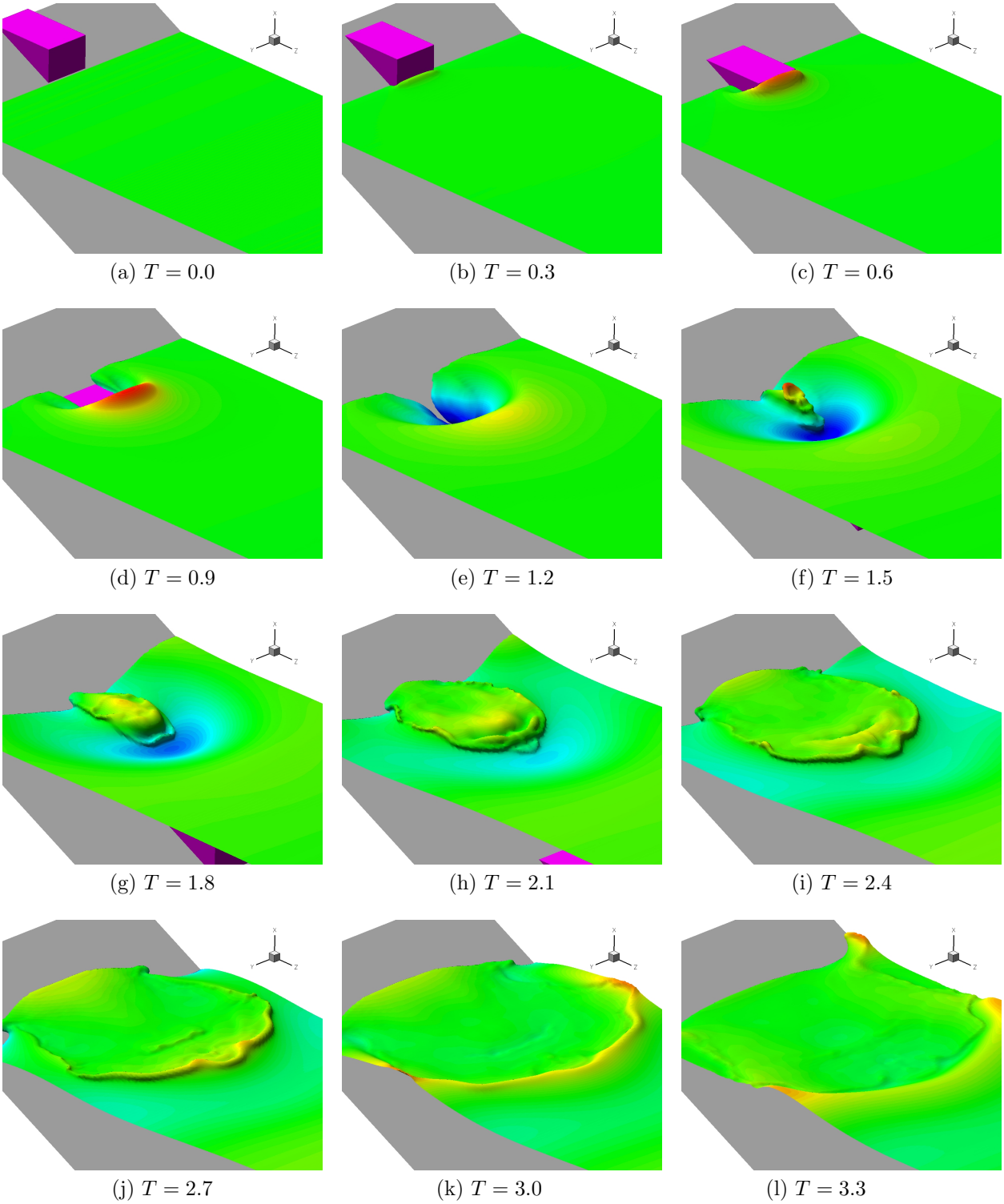
These snapshots qualitatively agree with the simulation in Ref. 22 very well. To further verify the solution, the free surface fluctuations are compared with the data recorded from a series of wave gauges in the experiment and shown in Fig. 8. The overall agreement is very good, especially, in the beginning stage of the landslide process. Simulations on finer grids are necessary to obtain a possible better agreement.

V. Conclusions

In this paper, a combined immersed-boundary/ghost-fluid sharp interface method has been developed for the simulations of three-dimensional two-phase flows interacting with moving bodies on fixed Cartesian grids. This work is based on a sharp interface immersed boundary method for fluid flows interacting with moving bodies[32] and a level set based ghost fluid method (GFM) for two-phase interface treatment.[21,15] The mathematical formulation for two-phase flows with jump conditions has been presented. Detailed numerical methods for the solution of complicated solid/liquid/gas systems have been discussed. The emphases have been on the four-step fractional-step method for velocity-pressure coupling, jump conditions treatment, and contact angle boundary conditions. Several cases with different scales are used to validate our numerical method. In the first case, the parasitic current in still fluid induced by numerical errors is chosen to show the accuracy of our ghost fluid implementation for very small scale problems. Then the small scale cases of water exit and entry of a circular cylinder are used to demonstrate how the immersed boundary approach works together with ghost fluid method. Finally, the case of waves generated by sliding mass is presented to illustrate the applicability of our method in large scale problems. The results from all cases are satisfying and agree with reference data very well.

Acknowledgments

This research is sponsored by the Office of Naval Research under Grant N00014-01-1-0073 and N00014-06-1-0420, under the administration of Dr. Patrick Purtell. The authors are grateful to Prof. T.-R. Wu for providing the landslide experimental data.



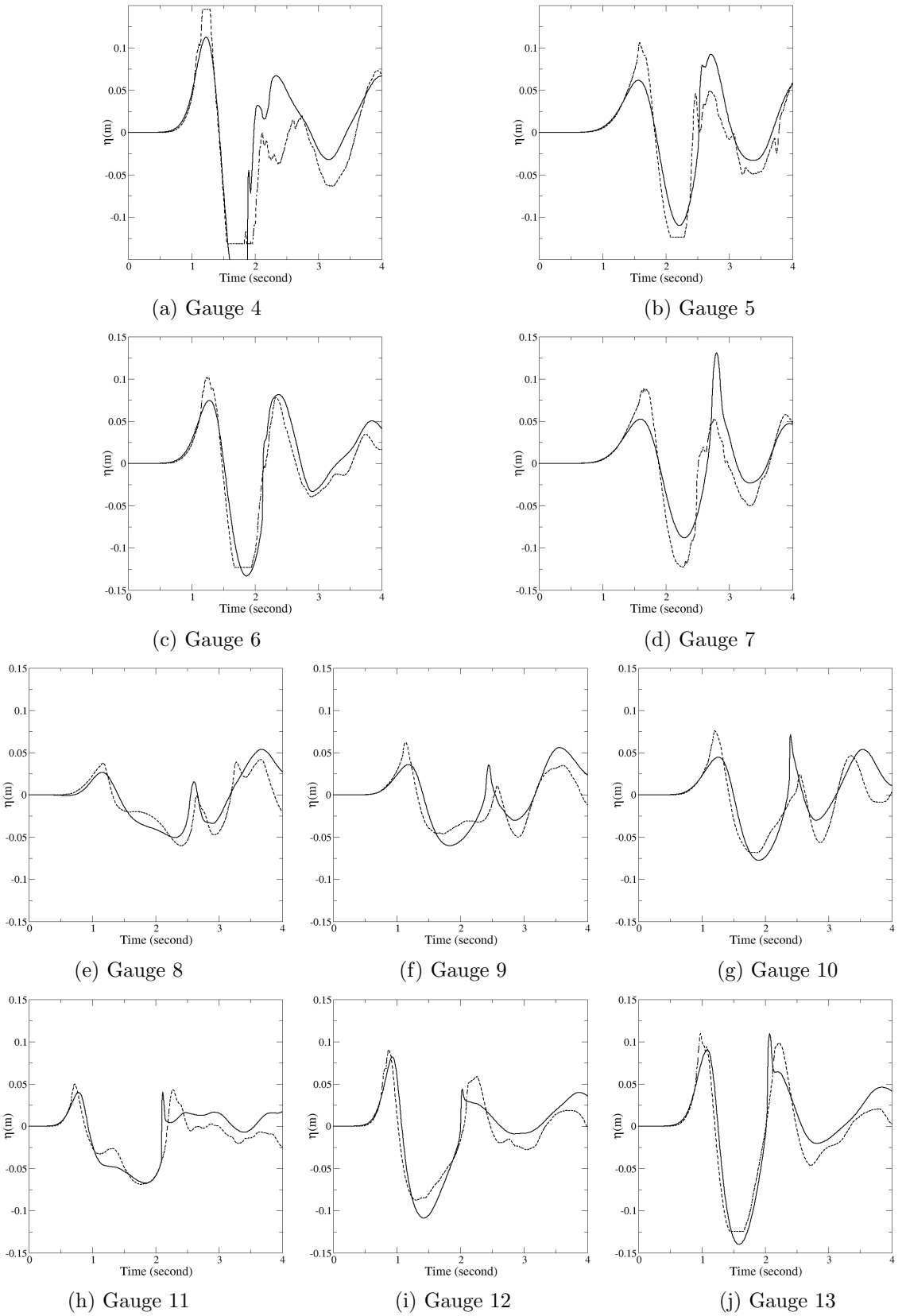
(a) $T = 0.0$ (b) $T = 0.3$ (c) $T = 0.6$

(d) $T = 0.9$ (e) $T = 1.2$ (f) $T = 1.5$

(g) $T = 1.8$ (h) $T = 2.1$ (i) $T = 2.4$

(j) $T = 2.7$ (k) $T = 3.0$ (l) $T = 3.3$

Figure 7: Landslide generated waves. Free surface colored by elevation.



(a) Gauge 4 (b) Gauge 5

(c) Gauge 6 (d) Gauge 7

(e) Gauge 8 (f) Gauge 9 (g) Gauge 10

(h) Gauge 11 (i) Gauge 12 (j) Gauge 13

Figure 8: Time series of free surface fluctuations at different wave gauges. Dotted lines: experimental data; Solid lines: present simulation.

References

[1]E. Balaras, Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations, Comput. Fluids, 33 (2004) 375–404.

[2]E. Balaras and J. Yang, Non-boundary conforming methods for large-eddy simulations of biological flows, ASME J. Fluids Eng, 127 (2005) 851-857.

[3]S. Balay, W.D. Gropp, L.C. McInnes, and B.F. Smith, Efficient Management of Parallelism in Object Oriented Numerical Software Libraries, in Modern Software Tools in Scientific Computing, Edited by E. Arge, A.M. Bruaset, H.P. Langtangen, Birkhauser Press, 1997, 163–202.

[4]R.M. Beam and R.F. Warming, An implicit finite-difference algorithm for hyperbolic systems in conservation-law form, J. Comput. Phys., 22 (1976) 87–110.

[5]Y.C. Chang,T.Y. Hou,B. Merriman, and S. Osher, A Level Set Formulation of Eulerian Interface Capturing Methods for Incompressible Fluid Flows, J. Comp. Phys. 124 (1996) 449–464.

[6]H. Choi and P. Moin, Effects of the Computational Time Step on Numerical Solutions of Turbulent Flow, J. Comput. Phys., 113 (1994) 1–4.

[7]E.A. Fadlun, R. Verzicco, P. Orlandi, and J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys, 161 (2000) 35–60.

[8]M.M. Francois, S.J. Cummins, E.D. Dendy, D.B. Kothe, J.M. Sicilian, and M.W. Williams, A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework, J. Comput. Phys., 213 (2006) 141–173.

[9]F. Gibou, L. Chen, D. Nguyen and S. Banerjee, A level set based sharp interface method for the multiphase incompressible NavierStokes equations with phase change, J. Comput. Phys., 222 (2007) 536–555.

[10]Greenhow M, Lin W.-M., Non-linear free surface effects: experiments and theory, Report no. 83-19, Dept. Ocean Engng., Cambridge, MIT, 1983.

[11]M. Greenhow, S. Moyo, Water entry and exit of horizontal circular cylinders, Philos. Trans. Math. Phys. Eng. Sci., 355 (1997) 551–563.

[12]J.-M. Hong, T. Shinar, M. Kang, R. Fedkiw, On boundary condition capturing for multiphase interfaces, J. Sci. Comput., 31 (2007) 99–125.

[13]G.-S. Jiang, and D. Peng, Weighted ENO Schemes for HamiltonJacobi Equations, SIAM J. Sci. Comp., 21 (2000) 2126–2143.

[14]G.-S. Jiang, C.-W. Shu, Efficient Implementation of Weighted ENO Schemes, J. Comput. Phys., 126 (1996) 202–228.

[15]M. Kang, R.P. Fedkiw, X.-D. Liu, A Boundary Condition Capturing Method for Multiphase Incompressible Flow, Journal of Scientific Computing, 15 (2000) 323–360.

[16]J. Kim, D. Kim, H. Choi, An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries, J. Comput. Phys., 171 (2001) 132–150.

[17]K.M.T. Kleefsman, G. Fekken, A.E.P. Veldman, B. Iwanowski, and B. Buchner, A Volume-of-Fluid based simulation method for wave impact problems, J. Comput. Phys., 206 (2005) 363-393.

[18]B.P. Leonard, Stable and accurate convective modeling procedure based on quadratic upstream interpolation, Computer Methods in Applied Mechanics and Engineering, 19 (1979) 59–98.

[19]R. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal., 31 (1994) 1019–1044.

[20]Lin, P., A fixed-grid model for simulation of a moving body in free surface flows, Computers & Fluids, 36 (2007) 549–561.

[21]X.D. Liu, R. Fedkiw and M. Kang, A boundary condition capturing method for Poissons equation on irregular domains, J. Comput. Phys. 154 (2000) 151–178.

[22]P.L.F Liu, T.-R. Wu, F. Raichlen, C.E. Synolakis, and J.C. Borrero, Runup and rundown generated by three-dimensional sliding masses, J. Fluid Mech., 536 (2005) 107–144.

[23]N. Mattor, T.J. Williams, and D.W. Hewett, Algorithm for Solving Tridiagonal Matrix Problems in Parallel, Parallel Computing, 21 (1995) 1769-1782.

[24]D. Peng, B. Merriman,S. Osher, H. Zhao, and M. Kang, A PDE-Based Fast Local Level Set Method, J. Comp. Phys., 155 (1999) 410-438.

[25]C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys., 10 (1972) 252–271.

[26]L. Qia, D.M. Causon, C.G. Mingham, and D.M. Ingram, A free-surface capturing method for two fluid flows with moving bodies, Proc. R. Soc. A-Math. Phys. Eng. Sci. 462 (206): 21–42.

[27]C. W. Shu and S. Osher, Efficient Implementation of Essentially Non-oscillatory Shock-capturing Schemes, J. Comp. Phys., 77 (1988) 439-471.

[28]M. Sussman, P. Smereka, and S. Osher, A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow, J. Comp. Phys., 114 (1994) 146-159.

[29]M. Sussman, K.M. Smith, M.Y. Hussaini, M. Ohta, R. Zhi-Wei, A sharp interface method for incompressible two-phase flows, J. Comput. Phys., 221 (2007) 469–505.

[30]H.S. Udaykumar, H.-C. Kan, W. Shyy, and R. Tran-Son-Tay, Multiphase Dynamics in Arbitrary Geometries on Fixed Cartesian Grids, J. Comput. Phys., 137 (1997) 366–405.

[31]S.P. van der Pijl, A. Segal, C. Vuik, and P. Wesseling, A mass-conserving Level-Set method for modelling of multi-phase flows, Int. J. Numer. Meth. Fluids, 47 (2005) 339–361.

[32]J. Yang, and E. Balaras, An Embedded-boundary Formulation for Large-eddy Simulation of Turbulent Flows Interacting with Moving Boundaries, J. Comp. Phys., 215 (2006) 12-40.