

A HEAVISIDE-FUNCTION APPROACH FOR THE INTERACTION OF TWO-PHASE FLUID AND NON-DEFORMABLE SOLID

MYUNGJOO KANG^a AND CHOHOONG MIN^{b,*}

ABSTRACT. We introduce a Heaviside-function formulation of the interaction between incompressible two-phase fluid and a non-deformable solid. Fluid and solid interact in two ways : fluid satisfies the Dirichlet boundary condition imposed by the velocity field of solid, and solid is accelerated by the surface traction exerted by fluid. The two-way couplings are formulated by the Heaviside function to the interface between solid and fluid. The cumbersome treatment of interface is taken care of by the Heaviside function, and the interaction is discretized in a simple manner. The discretization results in a stable and accurate projection method.

1. INTRODUCTION

In this paper, we introduce a Heaviside-function formulation of incompressible two-phase fluid interacting with a non-deformable solid. For the two-phase fluid, we take the level-set approach by Sussman et al [20, 21]. The interface between two phases, which we call air and water, is tracked by the level-set method [15]. The level-set function, which we denote by ϕ^{air} , splits the regions of air and water by its sign: it is taken positive in the region of water and negative in that of air.

$$(1) \quad \begin{cases} U_t + (U \cdot \nabla) U + \frac{1}{\rho} \nabla p = g + \frac{1}{\rho} \nabla \cdot (2\mu D) \\ \phi_t^{air} + (U \cdot \nabla) \phi^{air} = 0 \\ \nabla \cdot U = 0 \end{cases}$$

Here U is the fluid velocity field, ρ is the fluid density, μ is the fluid viscosity, $D(U) = \frac{1}{2} (\nabla U + (\nabla U)^T)$ is the strain-rate tensor, and g is the gravity acceleration. We do not include the surface-tension force acting on the interface between water and air. The magnitude of surface tension is linearly proportional to the mean

Received by the editors April 2, 2012. Accepted May 10, 2012.

2000 *Mathematics Subject Classification.* 76D05.

Key words and phrases. incompressible fluid, Navier-Stokes equations, fluid-solid interaction, Heaviside function.

*Corresponding author.

curvature $\nabla \cdot \left(\frac{\nabla \phi^{air}}{\|\nabla \phi^{air}\|} \right)$. The second order derivatives in the curvature calculation requires the level function to keep its smoothness during the simulation. Due to the lack of diffusion in the hyperbolic equation for ϕ^{air} , it is not simple to achieve the regularity requirement. It was pointed out in [5] that curvature calculation may become unreliable very soon, unless a high order method is used in approximating the hyperbolic equation. The temporal and spatial derivatives of the fluid equations will be discretized with some usual second order methods whose second order accuracy is not sufficient for the curvature calculation. Our main interest is not on the interfacial phenomenon between air and water, but on the interfacial phenomenon between fluid and water. From these reasons, we omit the surface tension term in fluid equations, and seek its inclusion in future work.

Within a narrow transition region around the interface, the two materials are assumed to be mixed, so that the density and viscosity are set to smoothly change across the interface by the following formula. In the below, α denotes the thickness of the transition region. In our computations, we set $\alpha = 2\Delta x$.

$$\begin{aligned} \rho(\phi^{air}) &= \rho^{air} + (\rho^{water} - \rho^{air}) H^\alpha(\phi^{air}) \\ \mu(\phi^{air}) &= \mu^{air} + (\mu^{water} - \mu^{air}) H^\alpha(\phi^{air}) \\ H^\alpha(\phi^{air}) &= \begin{cases} 1 & \text{if } \phi^{air} > \alpha \\ 0 & \text{if } \phi^{air} < -\alpha \\ \frac{1}{2} \left(\frac{\phi^{air}}{\alpha} + \frac{1}{\pi} \sin \left(\frac{\pi \phi^{air}}{\alpha} \right) \right) & \text{else} \end{cases} \end{aligned}$$

Non-deformable solid, or often called as rigid body, is represented by four state vectors : center of mass $C(t) \in \mathbb{R}^3$, linear momentum $P(t) \in \mathbb{R}^3$, angular momentum $L(t) \in \mathbb{R}^3$, and orientation matrix $R(t) \in \mathbb{R}^{3 \times 3}$. The state vectors are advanced in time by the Euler's equations [9, 7].

$$(2) \quad \begin{cases} C' = \frac{1}{m} P \\ P' = mg + f \\ L' = \tau \\ R' = \omega^\times R \end{cases},$$

where f is the given force, τ is the given torque, $\omega = Inertia^{-1}L$ is the angular velocity, $Inertia = \int \rho(x^T x - x x^T)$ is the inertia matrix, and ω^\times is the skew-symmetric matrix corresponding to the cross product $\omega \times x = \omega^\times x$ for all x . Fluid and solid interact in two ways: fluid satisfies the Dirichlet boundary condition imposed by the velocity field of solid, and solid is accelerated by the surface traction

exerted by fluid.

$$(3) \quad \begin{cases} U &= U^{solid} \text{ on } \Gamma \\ f &= \int_{\Gamma} (-p + 2\mu D) N \\ \tau &= \int_{\Gamma} (x - C) \times ((-p + 2\mu D) N) \end{cases},$$

where Γ is the interface between fluid and solid, $U^{solid}(x, t) = \frac{P(t)}{m} + \omega(t) \times (x - C(t))$ is the velocity field of solid, and N is the unit normal vector pointing outward to fluid.

Many efficient finite difference methods have been developed for fluid simulations. Chorin's projection method [4] introduced a way to decouple the pressure term from the equation of momentum conservation, so that the saddle-point system can be avoided. The projection method is first order accurate, and gets improved by many: Bell and Colella [2], Kim and Moin [8], and E and Liu [6]. Finite difference methods are by nature suited for rectangular domains. It is rather recent to see the development of finite difference methods for irregular domains. Purvis and Burkhalter [16] introduced an efficient projection method for fluid in irregular domains, and their work was followed by Min and Gibou [14]. In [1], Batty et al formulated the interaction between fluid and non-deformable solid as a Kinetic-energy minimization, and discretized the interaction by the Euler-Lagrange equation of the minimization. In [17], Robinson-Mosher et al discretized the interaction by conserving the momentum transfer between fluid and solid, so that the stability of simulation is enhanced.

In this paper, we formulate the interaction by a Heaviside function $H(x, t)$ taking value one in fluid region and zero in solid region. Note that $\nabla H = \delta N$, where δ is the Dirac delta function with support on the interface Γ , and N is the unit normal vector field on Γ . The Dirichlet boundary condition and the forcing terms are then formulated as

$$(4) \quad \begin{cases} U \cdot \nabla H &= U^{solid} \cdot \nabla H \\ f &= \int_{\mathbb{R}^d} (-p + 2\mu D) \nabla H \\ \tau &= \int_{\mathbb{R}^d} (x - C) \times ((-p + 2\mu D) \nabla H) \end{cases}.$$

The use of Heaviside function absorbs the information of irregular domain and enables us to discretize the interaction in a simple manner. More importantly, the discretization satisfies a stability condition stated in theorem 3.1.

2. TEMPORAL DISCRETIZATION

We take the projection approach : the fluid equations (1) and the solid equations (2) are advanced without the pressure terms, and then the advanced variables are

projected to satisfy the incompressible condition and the interaction equations (4). Because of the diffusion term, an explicit discretization of fluid momentum equation severely restrict the time step as $\Delta t = O(\Delta x^2)$. Thus, we implicitly discretize the fluid momentum equation and explicitly discretize the other equations that are free of diffusion.

The state of fluid is represented by velocity field $U(x, t)$ and level function $\phi^{air}(x, t)$, and the state of solid is represented by $C(t)$, $P(t)$, $L(t)$, and $R(t)$. A level function $\phi(x, t)$ describes the interface between fluid and solid, so that $\Omega^{fluid} = \{x | \phi(x) > 0\}$ and $\Omega^{solid} = \{x | \phi(x) < 0\}$, and the heaviside function is then derived as $H(x, t) = H(\phi(x, t))$, where $H(\cdot)$ is the one dimensional heavside function.

2.1. Advancing solid variables without pressure. The solid variables are advanced with the second order Adams-Bashford method.

$$\left\{ \begin{array}{l} \frac{\frac{3}{2}C^{n+1} - 2C^n + \frac{1}{2}C^{n-1}}{\Delta t} = 2P^n - P^{n-1} \\ \frac{\frac{3}{2}P^* - 2P^n + \frac{1}{2}P^{n-1}}{\Delta t} = 2f^{*,n} - f^{*,n-1} + mg \\ \frac{\frac{3}{2}L^* - 2L^n + \frac{1}{2}L^{n-1}}{\Delta t} = 2\tau^{*,n} - \tau^{*,n-1} \\ \frac{\frac{3}{2}R^{n+1} - 2R^n + \frac{1}{2}R^{n-1}}{\Delta t} = 2(\omega^n)^\times R^n - (\omega^{n-1})^\times R^{n-1} \end{array} \right.$$

Here $f^* = \int_{\mathbb{R}^d} (2\mu D) \nabla H$ and $\tau^* = \int_{\mathbb{R}^d} (x - C) \times (2\mu D) \nabla H$. Note that f^* and τ^* miss the pressure terms in their integrations. P^* and L^* require a further update with the pressure term, while the update of C^{n+1} and R^{n+1} is complete. The level function ϕ^{n+1} representing the interface between fluid and solid is updated as $\phi^{n+1}(x) = \phi^0 \left((R^{n+1})^{-1} (x - C^{n+1}) + C^0 \right)$, for each x .

2.2. Advancing fluid variables without pressure. Writing the fluid momentum equation in Lagrangian coordinates, we have

$$\frac{DU}{Dt} = \frac{1}{\rho} \nabla \cdot (-p + 2\mu D(U)) + g.$$

We omit the pressure term and discretize the time derivative with the second order backward-difference-formula to have

$$\begin{aligned} \frac{\frac{3}{2}U^* - 2U_d^n + \frac{1}{2}U_d^{n-1}}{\Delta t} &= \frac{1}{\rho^{n+1}} \nabla \cdot (\mu^{n+1} D(U^*)) + g \text{ in } \Omega^{fluid, n+1} \\ U^* &= U^{solid, n+1} + \frac{3}{2} \Delta t \cdot \frac{\nabla p^n}{\rho^{n+1}} \text{ on } \Gamma^{n+1} \end{aligned}$$

2.3. Projection. Given a velocity field U^* of fluid and momentum vectors P^* and L^* of solid, the incompressible condition and the interaction identities (4) can be enforced by the following projection method. From the fluid equations (1) and the solid equations (2), we make the following ansatz with pressure variable p .

$$\begin{aligned} U^{n+1} &= U^* - \frac{2\Delta t}{3} \frac{\nabla p^{n+1}}{\rho^{n+1}} \text{ in } \Omega^{fluid,n+1} \\ P^{n+1} &= P^* - \frac{2\Delta t}{3} \int_{\mathbb{R}^d} p^{n+1} \nabla H^{n+1} \\ L^{n+1} &= L^* - \frac{2\Delta t}{3} \int_{\mathbb{R}^d} p^{n+1} (x - C^{n+1}) \times \nabla H^{n+1} \end{aligned}$$

For simplicity, let $q^{n+1} = \frac{2\Delta t}{3} p^{n+1}$, then

$$\begin{aligned} U^{n+1} H^{n+1} &= U^* H^{n+1} - \frac{H^{n+1}}{\rho^{n+1}} \nabla q^{n+1} \\ P^{n+1} &= P^* - \int_{\mathbb{R}^d} q^{n+1} \nabla H^{n+1} dV(x) \\ L^{n+1} &= L^* - \int_{\mathbb{R}^d} q^{n+1} (x - C^{n+1}) \times \nabla H^{n+1} dV(x) \end{aligned}$$

From the fluid incompressible condition $H(\nabla \cdot U) = 0$ and the interaction identity $U \cdot \nabla H = U^{solid} \cdot \nabla H$,

$$\begin{aligned} -\nabla \cdot \left(\frac{H^{n+1}}{\rho^{n+1}} \nabla q^{n+1} \right) &= -\nabla \cdot (H^{n+1} U^*) + \nabla \cdot (H^{n+1} U^{n+1}) \\ &= -\nabla \cdot (H^{n+1} U^*) + (\nabla H^{n+1}) \cdot U^{solid,n+1} \end{aligned}$$

Here $U^{solid} = \frac{1}{m} P + (Inertia)^{-1} L \times (x - C)$. There are pressure terms on the right-hand-side, since

$$\begin{aligned} &(\nabla H^{n+1}) \cdot U^{solid,n+1} \\ &= (\nabla H^{n+1}) \cdot U^{solid,*} - (\nabla H^{n+1}) \cdot \left(\int_{\mathbb{R}^d} q^{n+1} \frac{1}{m} \nabla H^{n+1} dV(x) \right) \\ &\quad - (\nabla H^{n+1}) \cdot \left((Inertia^{n+1})^{-1} \left(\int_{\mathbb{R}^d} q^{n+1} (x - C^{n+1}) \times \nabla H^{n+1} dV(x) \right) \right) \\ &\quad \times (x - C^{n+1}). \end{aligned}$$

Collecting all the pressure terms on the left-hand-side, we obtain the following linear system.

$$(5) \quad -\nabla \cdot \left(\frac{H^{n+1}}{\rho^{n+1}} \nabla q^{n+1} \right) + \nabla H^{n+1} \cdot \left(\int_{\mathbb{R}^d} q^{n+1}(x) \frac{1}{m} \nabla H^{n+1}(x) dx \right) \\ (\text{Inertia}^{n+1})^{-1} \left((x - C^{n+1}) \times \nabla H^{n+1} \right) \cdot \left(\int_{\mathbb{R}^d} q^{n+1}(x) ((x - C^{n+1}) \right. \\ \left. \times \nabla H^{n+1}(x)) dx \right) = -\nabla \cdot \left(H^{n+1} (U^* - U^{\text{solid},*}) \right)$$

Theorem 2.1. *The linear system of equation (5) is symmetric positive definite.*

Proof. Let $\mathfrak{L}(q)$ denote the left-hand-side operation of q . For notational conveniences, we omit the super-script $n + 1$. For the symmetry of the operator \mathfrak{L} , we show that $\int_{\mathbb{R}^d} \mathfrak{L}(q_1) q_2 = \int_{\mathbb{R}^d} \mathfrak{L}(q_2) q_1$ for any L^2 integrible functions q_1 and q_2 .

$$\int_{\mathbb{R}^d} \mathfrak{L}(q_1) q_2 = - \int_{\mathbb{R}^d} \left(\nabla \cdot \left(\frac{H}{\rho} \nabla q_1 \right) \right) q_2 + \int_{\mathbb{R}^d} \nabla H \cdot \left(\int_{\mathbb{R}^d} q_1 \frac{1}{m} \nabla H \right) q_2 \\ + \int_{\mathbb{R}^d} (\text{Inertia})^{-1} \left((x - C) \times \nabla H \right) \cdot \left(\int_{\mathbb{R}^d} q_1 \left((x - C) \times \nabla H \right) \right) q_2 \\ = \int_{\mathbb{R}^d} \frac{H}{\rho} \nabla q_1 \cdot \nabla q_2 + \frac{1}{m} \left(\int_{\mathbb{R}^d} q_2 \nabla H \right) \cdot \left(\int_{\mathbb{R}^d} q_1 \nabla H \right) \\ + \left((\text{Inertia})^{-1} \left(\int_{\mathbb{R}^d} q_2 \left((x - C) \times \nabla H \right) \right) \right) \cdot \left(\int_{\mathbb{R}^d} q_1 \left((x - C) \times \nabla H \right) \right)$$

Note that we arrived a symmetric form for q_1 and q_2 . For positive-definiteness,

$$\int_{\mathbb{R}^d} \mathfrak{L}(q) q = \int_{\mathbb{R}^d} \frac{H}{\rho} \|\nabla q\|^2 + \frac{1}{m} \left\| \int_{\mathbb{R}^d} q \nabla H \right\|^2 + \left((\text{Inertia})^{-1} \left(\int_{\mathbb{R}^d} q \left((x - C) \times \nabla H \right) \right) \right) \\ \cdot \left(\int_{\mathbb{R}^d} q \left((x - C) \times \nabla H \right) \right) \geq 0$$

□

After the solution q^{n+1} is obtained, the variables are updated as

$$\begin{aligned}
 U^{n+1} &= U^* - \frac{\nabla q^{n+1}}{\rho^{n+1}} \text{ if } H^{n+1} \neq 0 \\
 P^{n+1} &= P^* - \int_{\mathbb{R}^d} q^{n+1} \nabla H^{n+1} dV(x) \\
 L^{n+1} &= L^* - \int_{\mathbb{R}^d} q^{n+1} (x - C^{n+1}) \times \nabla H^{n+1} dV(x).
 \end{aligned}$$

2.4. Extrapolation. The fluid variables U^{n+1} and p^{n+1} are valid when $H^{n+1} \neq 0$, or in the fluid region. Since the fluid regions are changing with time, We extrapolate them to the whole region, or at least a small band across the interface.

3. SPATIAL DISCRETIZATION

We take the Marker and Cell (MAC) grid: pressure p is sampled on grid centers, velocity field U on grid facets. The level function ϕ is sampled on grid centers. Let D_x denote the central finite differences in the x direction such that

$$\begin{aligned}
 (D_x p)_{i+\frac{1}{2},j} &= \frac{p_{i+1,j} - p_{i,j}}{\Delta x} \\
 (D_x u)_{ij} &= \frac{u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}}{\Delta x}.
 \end{aligned}$$

Similarly D_y denotes the central finite difference in the y direction. The gradient and divergence operators are approximated by the central finite differences, and denoted by G and Div , respectively.

$$\begin{aligned}
 G[p_{ij}] &= \left(\left[(D_x p)_{i+\frac{1}{2},j} \right], \left[(D_y p)_{i+\frac{1}{2},j} \right] \right) \\
 Div \left(\left[u_{i+\frac{1}{2},j} \right], \left[v_{i+\frac{1}{2},j} \right] \right) &= \left[(D_x u + D_y v)_{ij} \right]
 \end{aligned}$$

The spatial variables are discretized in a dimension-by-dimension approach, or can be extended to three dimensions in a straight-forward manner, so that we state only the two dimensional cases.

3.1. Strain-rate and diffusion terms. The strain-rate tensor

$$D(U) = \frac{1}{2} \left(\nabla U + (\nabla U)^T \right)$$

is discretized by the central finite differences. The elements of the matrix are sampled either at grid nodes or at grid cell centers.

$$\begin{cases} D_{i,j}^{11} &= (D_x u)_{ij} \\ D_{ij}^{22} &= (D_y v)_{ij} \\ D_{i+\frac{1}{2},j+\frac{1}{2}}^{12} = D_{i+\frac{1}{2},j+\frac{1}{2}}^{21} &= \frac{1}{2} (D_y u + D_x v)_{i+\frac{1}{2},j+\frac{1}{2}} \end{cases}$$

The diffusion term $\nabla \cdot (2\mu D)$ is also discretized by the central finite differences, and sampled at grid facets.

$$\begin{cases} (D_x(2\mu D))_{i+\frac{1}{2},j} &= (D_x(2\mu D^{11}) + D_y(2\mu D^{12}))_{i+\frac{1}{2},j} \\ (D_x(2\mu D))_{i,j+\frac{1}{2}} &= (D_x(2\mu D^{12}) + D_y(2\mu D^{22}))_{i,j+\frac{1}{2}} \end{cases}$$

3.2. Quadratic interpolation. We employ the stabilized quadratic interpolation on uniform grid [12]. Given function values at the corners of $[0, 1]^2$, the bilinear interpolation has the formula

$$f^{bi}(x, y) = (f_{00}(1-x) + f_{10}x)(1-y) + (f_{01}(1-x) + f_{11}x)y.$$

The bilinear interpolation preserves the maximum of the values, but is only second order accurate. Correcting its second order error terms with the minmod finite differences, we obtain the stabilized quadratic interpolation.

$$\begin{aligned} f^{quad}(x, y) &= f^{bi}(x, y) - \frac{1}{2}x(1-x)(\min\text{mod}(D_x^2 f_{00}, D_x^2 f_{10})(1-y) \\ &\quad + \min\text{mod}(D_x^2 f_{01}, D_x^2 f_{11})y) - \frac{1}{2}y(1-y)(\min\text{mod}(D_y^2 f_{00}, D_y^2 f_{01}) \\ &\quad \cdot (1-x) + \min\text{mod}(D_y^2 f_{10}, D_y^2 f_{11})x) \end{aligned}$$

Here the minmod operation takes the value of the argument with the smaller absolute magnitude. Though defined piecewisely on each grid cell, the interpolation is globally continuous.

3.3. Heaviside function. The Heaviside function H is discretized as $H = \frac{\nabla\phi^+ \cdot \nabla\phi}{\nabla\phi \cdot \nabla\phi}$ by Towers [22].

$$H_{i+\frac{1}{2},j} = \begin{cases} \left(\frac{D_x\phi \cdot D_x\phi^+ + D_y\phi \cdot D_y\phi^+}{(D_x\phi)^2 + (D_y\phi)^2} \right)_{i+\frac{1}{2},j} & \text{if } \left\| (D_x\phi)^2 + (D_y\phi)^2 \right\|_{i+\frac{1}{2},j} > 0 \\ 1 & \text{if } \left\| (D_x\phi)^2 + (D_y\phi)^2 \right\|_{i+\frac{1}{2},j} = 0, \phi_{i+\frac{1}{2},j} > 0 \\ 0 & \text{if } \left\| (D_x\phi)^2 + (D_y\phi)^2 \right\|_{i+\frac{1}{2},j} = 0, \phi_{i+\frac{1}{2},j} \leq 0 \end{cases}$$

Here ϕ^+ takes the value of ϕ if $\phi \geq 0$ and zero otherwise. Similarly $H_{i,j+\frac{1}{2}}$ is discretized.

The Heaviside discretization is very simple and second order accurate with smooth interfaces. With the presence of kink points near the interface, we note that it may not work properly and prefer a stabler discretization of [13]. ϕ is sampled on grid nodes, but the above discretization needs its values on grid cell centers. The values at the centers are calculated by the above quadratic interpolation algorithm.

3.4. Integration. The calculation of the surface traction terms requires integration on the interface Γ . Using the fact that $\nabla H = \delta N$, the integrations are approximated by the following summation formula.

$$\begin{aligned} \int_{\Gamma} pN &= \sum_{i,j} \left(p \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \Delta x \Delta y \\ \int_{\Gamma} 2\mu DN &= \sum_{i,j} \left(2\mu \begin{bmatrix} D^{11} & 0 \\ 0 & D^{22} \end{bmatrix} \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \Delta x \Delta y \\ &\quad + \sum_{i,j} \left(2\mu \begin{bmatrix} 0 & D^{12} \\ D^{21} & 0 \end{bmatrix} \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{i+\frac{1}{2},j+\frac{1}{2}} \Delta x \Delta y \end{aligned}$$

The torque term $\int_{\Gamma} (x - C) \times (-p + 2\mu D) N$ is discretized in the same manner as above.

3.5. Extrapolation. Consider extrapolating a function f defined in region Ω to a larger region $\Omega^\epsilon = \{x \mid \|x - y\| < \epsilon \text{ for some } y \in \Omega\}$. For each grid point $x \in \Omega^\epsilon - \Omega$, collect all of its neighboring grid points $\{x_1, \dots, x_N\}$ such that $\|x - x_i\| < \epsilon$ and $x_i \in \Omega$ for each i . Now we face a problem to evaluate $f(x)$ out of the grid points $\{x_1, \dots, x_N\}$ and the function values on them $\{f_1, \dots, f_N\}$. We take the Shepard algorithm [19] that is simple and stable in the maximum norm.

$$f(x) = \frac{\sum_{i=1}^N w_i f(x_i)}{\sum_{i=1}^N w_i}$$

The weight is chosen as $w_i = 1/\|x - x_i\|^2$ for each i , so that closer points influence the evaluation more. In our computations, we take $\epsilon = 4 \cdot \Delta x$.

3.6. Projection. Given a velocity field U^* of fluid and momentum vectors P^* and L^* of solid, the incompressible condition and the interaction identities (4) can be enforced by the following projection method. From the fluid equations (1) and the solid equations (2), we make the following ansatz with pressure variable p .

$$\begin{aligned} U &= U^* - \Delta t \cdot \frac{Gp}{\rho} \\ P &= P^* - \Delta t \cdot \sum_{i,j} \left(p \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \Delta x \Delta y \\ L &= L^* - \Delta t \cdot \sum_{i,j} \left(p(x - C) \times \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \Delta x \Delta y \end{aligned}$$

From the fluid incompressible condition $\nabla \cdot U = 0$ and the interaction identity $U \cdot \nabla H = U^{solid} \cdot \nabla H$, we set

$$(Div(HU))_{ij} = U_{ij}^{solid} \cdot \begin{bmatrix} D_x H \\ D_y H \end{bmatrix}_{ij}.$$

Here $U_{ij}^{solid} = m^{-1}P + M^{-1}L \times (X_{ij} - C)$ and X_{ij} denotes the grid node with index (i, j) . Combining the equations, a Poisson equation is derived for the pressure.

$$\begin{aligned} - \left(Div \left(\frac{H}{\rho} Gp \right) \right)_{ij} &= - (Div(H(U^* - U)))_{ij} \\ &= - (Div(HU^*))_{ij} + U_{ij}^{solid} \cdot \begin{bmatrix} D_x H \\ D_y H \end{bmatrix}_{ij} \end{aligned}$$

There are pressure terms on the right-hand-side, since

$$U_{ij}^{solid} = U_{ij}^{solid,*} + \sum_{ij} \left(p \left(\frac{1}{m} + M^{-1}(X - C) \right) \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \Delta x \Delta y.$$

Collecting all the pressure terms on the left-hand-side, we obtain the following linear system.

$$\begin{aligned} - \left(Div \left(\frac{H}{\rho} Gp \right) \right)_{ij} + \begin{bmatrix} D_x H \\ D_y H \end{bmatrix}_{ij} \cdot v + \left((X - C) \times \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \cdot \omega \\ = - (Div(HU^*))_{ij} + U_{ij}^{solid,*} \cdot \begin{bmatrix} D_x H \\ D_y H \end{bmatrix}_{ij} \\ \sum_{ij} \left(p \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} - \frac{m}{\Delta x \Delta y} v = 0 \\ \sum_{ij} \left(p (X - C) \times \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} - \frac{K}{\Delta x \Delta y} \omega = 0 \end{aligned}$$

For notational conveniences, we write the system in a block form.

$$\begin{bmatrix} A & J_1 & J_2 \\ J_1^T & -\frac{m}{a} & 0 \\ J_2^T & 0 & -\frac{K}{a} \end{bmatrix} \begin{bmatrix} p \\ v \\ \omega \end{bmatrix} = \begin{bmatrix} rhs \\ 0 \\ 0 \end{bmatrix}$$

The matrix is symmetric, but is not positive-definite, which puts a hesitation in using the conjugate gradient(CG) method. Actually it is positive-definite as long as the last two equations are satisfied. Moreover, being a Krylov space method, CG method can solve any symmetric linear system $Ax = b$ as long as the direction vector has non-zero conjugate length, $d \cdot Ad \neq 0$. From these reasons, we solved the linear

system by CG with the block LU preconditioner. In the below, M is the MILU preconditioner for A .

$$\begin{bmatrix} M & J_1 & J_2 \\ J_1^T & -\frac{m}{a} & 0 \\ J_2^T & 0 & -\frac{M}{a} \end{bmatrix}^{-1} = \left(\begin{bmatrix} I & 0 & 0 \\ J_1^T M^{-1} & I & 0 \\ J_2^T M^{-1} & U^T D_1^{-1} & I \end{bmatrix} \begin{bmatrix} M & J_1 & J_2 \\ 0 & D_1 & U \\ 0 & 0 & D_2 \end{bmatrix} \right)^{-1}$$

Here $D_1 = -\frac{m}{a} - J_1^T M^{-1} J_1$, $U = -J_1^T M^{-1} J_2$, and $D_2 = -\frac{M}{a} - J_2^T M^{-1} J_2 - U^T D_1^{-1} U$. Since ∇H has a support only near the interface, the matrix blocks J_1 and J_2 are stored sparse for faster matrix multiplication.

Theorem 3.1. *The projection by the pressure does not increases the kinetic energy of fluid and solid.*

Proof. The kinetic energy of fluid and solid is $K = \int_{\Omega} \frac{\rho}{2} U^2 + \frac{1}{2m} P \cdot P + \frac{1}{2} M^{-1} L \cdot L$, and discretized as

$$K(U, P, L) = \frac{1}{2} \sum_{ij} \left((\rho H u^2)_{i+\frac{1}{2},j} + (\rho H v^2)_{i,j+\frac{1}{2}} \right) + \frac{1}{2m} P \cdot P + \frac{1}{2} M^{-1} L \cdot L.$$

Comparing $K(U^*, P^*, L^*)$ with $K(U, P, L)$,

$$\begin{aligned} K(U^*, P^*, L^*) &= K(U, P, L) + \frac{\Delta t^2}{2} \left(\sum_{ij} (D_x p)_{i+\frac{1}{2},j}^2 + (D_y p)_{i,j+\frac{1}{2}}^2 + m v^2 + M \omega^2 \right) \\ &\quad + \Delta t \cdot \left(\sum_{ij} \left((H u D_x p)_{i+\frac{1}{2},j} + (H v D_y p)_{i+\frac{1}{2},j} \right) + P \cdot v + L \cdot \omega \right). \end{aligned}$$

Using the discrete integration-by-parts formula in MAC grids,

$$\begin{aligned} & - \sum_{ij} \left((H u D_x p)_{i+\frac{1}{2},j} + (H v D_y p)_{i+\frac{1}{2},j} \right) \\ &= \sum_{ij} (p \text{Div}(HU))_{ij} \\ &= \sum_{ij} \left(p U^{solid} \cdot \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \\ &= \sum_{ij} \left(p (P/m + M^{-1} L \times (X - C)) \cdot \begin{bmatrix} D_x H \\ D_y H \end{bmatrix} \right)_{ij} \\ &= P \cdot v + L \cdot \omega. \end{aligned}$$

Thus $K(U^*, P^*, L^*) \geq K(U, P, L)$. □

4. TEMPORAL DISCRETIZATION

We take the projection approach : the Navier-Stokes equations (1) and the Euler equations (2) are advanced with a pressure guess, and then the advanced variables are projected. While explicit temporal discretization for ordinary differential equations have stability condition $\Delta t = O(1)$, their stability condition for convection-diffusion partial differential equations is much restrictive, $\Delta t = O(\Delta x^2)$. Thus we explicitly discretize the Euler equations for conveniences and implicitly discretize the Navier-Stokes equations for faster simulation.

4.1. Adams-Bashford method. The Euler equations are advanced with the second order Adams-Bashford method.

$$\begin{cases} (C^{n+1} - C^n) / \Delta t &= \frac{3}{2}P^n - \frac{1}{2}P^{n-1} \\ (P^* - P^n) / \Delta t &= \frac{3}{2}F^n - \frac{1}{2}F^{n-1} + mg \\ (L^* - L^n) / \Delta t &= \frac{3}{2}T^n - \frac{1}{2}T^{n-1} \\ (R^{n+1} - R^n) / \Delta t &= \frac{3}{2}(\omega^n)^\times R^n - \frac{1}{2}(\omega^{n-1})^\times R^{n-1} \end{cases}$$

Here $F = \int_{\mathbb{R}^d} (2\mu D) \nabla H$ and $T = \int_{\mathbb{R}^d} (x - C) \times (2\mu D) \nabla H$ are discretized by the integration algorithm in the previous section. Let $\phi^0(x)$ be the initial level function for the solid, taking positive value outside the solid and negative value inside. The level function ϕ^{n+1} is calculated as

$$\phi_{ij}^{n+1} = \phi^0 \left((R^{n+1})^{-1} (X_{ij} - C^{n+1}) + C^0 \right),$$

for each grid index i and j . The function value of ϕ^0 is calculated by the quadratic interpolation in the previous section.

4.2. Collision resolution. When the rigid body collides with the rectangular boundary, the collision is resolved by the momentum impulse J calculated as follows. Let the collision occur at position X with normal vector N on the boundary.

$$\begin{aligned} J_I &= - \left(\frac{N \cdot V^-}{N \cdot KN} \right) N \\ J_{II} &= -\tilde{K}^{-1} V^- \\ J &= (1 + e_n) J_I + (1 + e_t) (J_{II} - J_I) \end{aligned}$$

Here $V^- = \frac{1}{m}P + (K^{-1}L) \times (X - C)$ is the velocity at the collision point X and $\tilde{K} = \frac{1}{m} - (X - C)^\times K^{-1} (X - C)^\times$ is the collision matrix. J_I is the sliding impulse and J_{II} is the sticking impulse. $e_n \in [0, 1]$ and $e_t \in [-1, 1]$ are the bouncing parameter in the normal and tangential direction, respectively. A moderate elastic

bouncing is taken by choosing $e_n = 0.7$ and $e_t = -0.7$. See the details in [3, 10]. Then the linear and angular momentum are updated as

$$\begin{aligned} P^+ &= P^- + J \\ L^+ &= L^- + (X - C) \times J. \end{aligned}$$

Here P^- and L^- are momentum before the collision resolution, and P^+ and L^+ are after collision resolution.

4.3. Semi-Lagrangian method. The convection term is discretized by the second order semi-Lagrangian method. Let x^{n+1} be a grid node at time t^{n+1} . The node point is moved backward along the characteristic curves by the second order Runge-Kutta method.

$$\begin{aligned} x^{n+\frac{1}{2}} &= x^{n+1} - \frac{\Delta t^{n+\frac{1}{2}}}{2} U^n(x^{n+1}) \\ x^n &= x^{n+1} - \Delta t^{n+\frac{1}{2}} U^{n+\frac{1}{2}}(x^{n+\frac{1}{2}}) \end{aligned}$$

The intermediate velocity is extrapolated as

$$U^{n+\frac{1}{2}}(x^{n+\frac{1}{2}}) = U^n(x^{n+\frac{1}{2}}) + \frac{1}{2} \frac{\Delta t^{n+\frac{1}{2}}}{\Delta t^{n-\frac{1}{2}}} (U^n(x^{n+\frac{1}{2}}) - U^{n-1}(x^{n+\frac{1}{2}})).$$

The Lagrangian coordinates do not fall on grid nodes in general, and the function values on them are evaluated using the quadratic interpolation introduced in the previous section.

$$\begin{aligned} U_d^n(x^{n+1}) &= U^n(x^n) \\ \phi^{air,n+1}(x^{n+1}) &= \phi^{air,n}(x^n) \end{aligned}$$

4.4. Reinitializing Level-set function. The updated level function ϕ^{air} needs to be reinitialized to prevent cases that its graph is too flat or too steep around the interface. In such cases, a small perturbation of the level function may result in a big change of interface location, and the level function may lose enough regularity near the interface. It is therefore desired to replace the level function with a better behaved one, the signed distance function to the interface. We update the level function through the reinitialization equation [21].

$$\phi_t + \text{sgn}(\phi^0) (\|\nabla\phi\| - 1) = 0$$

Here $\phi^0(x) = \phi(x, 0) = \phi^{air}(x)$. The spatial derivatives are discretized by the second order ENO finite differences with subcell resolution near the interface [18, 11]. The temporal derivative is discretized by the efficient Gauss-Seidel update [11].

$$\phi_{ij} := \phi_{ij} - \Delta t_{ij} \cdot \text{sgn}(\phi_{ij}^0) (H_G(D_x^+ \phi_{ij}, D_x^- \phi_{ij}, D_y^+ \phi_{ij}, D_y^- \phi_{ij}) - 1)$$

Here H_G is the Gouunov numerical Hamiltonian [15] whose formula is

$$H_G(a, b, c, d) = \begin{cases} \sqrt{\max((a^-)^2, (b^+)^2) + \max((c^-)^2, (d^+)^2)} & \text{when } \text{sgn}(\phi^0) \geq 0 \\ \sqrt{\max((a^+)^2, (b^-)^2) + \max((c^+)^2, (d^-)^2)} & \text{when } \text{sgn}(\phi^0) < 0 \end{cases}$$

Here D_x^\pm and D_y^\pm denote the one-sided ENO finite differences at x and y directions. Since it is a dimension by dimension approach, we only consider D_x^+ . The details of the other cases can be found in [11].

$$D_x^+ \phi_{ij} = \frac{\phi_{i+1,j} - \phi_{ij}}{\Delta x} - \frac{\Delta x}{2} \text{minmod}(D_x^2 \phi_{ij}, D_x^2 \phi_{i+1,j})$$

Near the interface or $\phi_{i+1,j} \phi_{ij} < 0$, the finite difference is modified to impose the condition that $\phi = 0$ whenever $\phi^0 = 0$.

$$D_x^+ \phi_{ij} = \frac{0 - \phi_{ij}}{\Delta x^+} - \frac{\Delta x^+}{2} \text{minmod}(D_{xx} \phi_{ij}, D_{xx} \phi_{i+1,j})$$

Here Δx^+ is the distance from the grid node to the interface point. The time step is taken as

$$\Delta t_{ij} = .45 \cdot \min(\Delta x^+, \Delta x^-, \Delta y^+, \Delta y^-).$$

4.5. Crank-Nicolson method. The intermediate variable U^* is obtained by applying the Crank-Nicolson method to the equation of momentum conservation, which results in the following linear system.

$$\rho^{n+1} U^* - \Delta t \cdot \text{Div}(2\mu^{n+1} D(U^*)) = \rho^{n+1} U_d^n + \Delta t \cdot \text{Div}(2\mu^{n+1} D(U_d^n)) + \Delta t \cdot \rho^{n+1} g$$

The system is discretized at grid facets inside the fluid region, $\{\phi^{n+1} > 0\}$. Outside the fluid region, the Dirichlet boundary condition is imposed as follows.

$$U^* = U^{\text{solid},n+1} + \Delta t \left(\frac{Gp^n}{\rho^n} \right)_d$$

Note that the linear system couples all the components of the velocity field U^* , unless μ is constant everywhere. The linear system is symmetric positive definite, and can be efficiently solved in all our tested examples by the conjugate gradient method with the Jacobi preconditioner.

4.6. CFL condition. The SL-CN is unconditionally stable. For simplicity, we take the CFL condition

$$\Delta t_{CFL} = \min \left(\frac{\Delta x}{\|u^n\|_\infty}, \frac{\Delta y}{\|v^n\|_\infty}, \frac{\Delta z}{\|w^n\|_\infty} \right).$$

If the time step is similar to the CFL condition such that $\frac{1}{2}\Delta t_{CFL} \leq \Delta t \leq \Delta t_{CFL}$, the variables are updated as $U^{n-1} := U^n$ and $U^n := U^{n+1}$. When $\Delta t > \Delta t_{CFL}$, the time step Δt keeps being reduced by half with averaging $U^{n-1} := \frac{1}{2}(U^{n-1} + U^n)$ until $\Delta t \leq \Delta t_{CFL}$. When $\Delta t < \frac{1}{2}\Delta t_{CFL}$, the time step is increased twice with the update $U^n := U^{n+1}$.

5. EXAMPLE

In all the following examples, the meter-kilogram-second(MKS) units are assumed. The physically parameters are chosen : $\rho^{air} = 1.226$, $\rho^{water} = 1000$, $\mu^{air} = 1.780 \times 10^{-5}$, $\mu^{water} = 1.137 \times 10^{-3}$, and $g = (0, -9.8, 0)$. The fluid and solid are bounded by a rectangular domain. On the rectangular boundary, the non-penetration(homogeneous Dirichlet) boundary condition is imposed for the normal velocity component, and the slip(homogeneous Neumann) boundary condition for the tangent velocity components.

5.1. Single vortex in a square domain. We test the convergence order of the fluid solver in a simple setting. Consider a domain $[-\frac{\pi}{2}, \frac{\pi}{2}]^2$ and a single vortex flow whose exact solutions are given as $u = -\cos(t) \cos(x) \sin(y)$, $v = \cos(t) \cos(y) \sin(x)$, and $p = -\frac{1}{4} \cos^2(t) (\cos(2x) + \cos(2y))$. Table 1 shows the second order accuracy of the fluid solver in the L^∞ and L^1 norms at time $t = \pi$. The momentum equation in equation (1) is discretized by the SL-BDF method. A better known choice for discretizing diffusion terms is the Crank-Nicolson method. Though it has the same order of accuracy as BDF, its combination with semi-Lagrangian and projection method results in a drop of overall accuracy to the first order [23]. The table also shows the accuracy drop of SL-CN method, and we take the SL-BDF combination to all the following examples.

5.2. Single vortex in an irregular domain. Consider an irregular domain

$$\left\{ (x, y) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]^2 \mid \cos x \cdot \cos y \geq .5 \right\}$$

and a single vortex flow whose exact solutions are given as $u = -\cos(t) \cos(x) \sin(y)$, $v = \cos(t) \cos(y) \sin(x)$, and $p = -\frac{1}{4} \cos^2(t) (\cos(2x) + \cos(2y))$. There are many

SL-BDF				
Grid resolution	L^∞ error	order	L^1 error	order
50^2	5.10×10^{-4}		1.08×10^{-4}	
100^2	1.29×10^{-4}	1.98	2.69×10^{-5}	2.01
200^2	3.16×10^{-5}	2.03	6.74×10^{-6}	2.00
400^2	7.81×10^{-6}	2.02	1.69×10^{-6}	2.00

SL-CN				
Grid resolution	L^∞ error	order	L^1 error	order
50^2	1.19×10^{-2}		4.18×10^{-3}	
100^2	5.98×10^{-3}	0.99	2.07×10^{-3}	1.01
200^2	2.99×10^{-3}	1.00	1.03×10^{-3}	1.01
400^2	1.49×10^{-3}	1.00	5.15×10^{-4}	1.00

Table 1. Accuracy of the velocity field for the single vortex example

Heaviside function : length fraction				
grid	L^∞ error	order	L^1 error	order
50^2	1.12×10^{-3}		1.72×10^{-4}	
100^2	2.24×10^{-4}	2.32	3.83×10^{-5}	2.17
200^2	5.28×10^{-5}	2.08	8.57×10^{-6}	2.16
400^2	1.33×10^{-5}	1.99	2.07×10^{-6}	2.05

Heaviside function : Tower's method				
grid	L^∞ error	order	L^1 error	order
50^2	3.21×10^{-1}		4.87×10^{-3}	
100^2	3.46×10^{-1}	-0.11	3.03×10^{-3}	0.68
200^2	3.78×10^{-1}	0.09	1.86×10^{-3}	0.70
400^2	3.83×10^{-1}	-0.02	1.49×10^{-3}	0.32

Table 2. Accuracy of the velocity field for the single vortex in an irregular domain

ways for discretizing heaviside function: One way is $H(x) = \frac{\nabla\phi^+ \cdot \nabla\phi}{\nabla\phi \cdot \nabla\phi}$, and another is a control-volume approach. Table 2 shows that the choice of length fraction leads to the second order accuracy in the L^∞ and L^1 norms at time $t = \pi$, but the Tower's approach leads to the first order convergence in L^1 and non-convergence in L^∞ . There are many discretizations of Heaviside functions that are second order accurate in approximate the integrals in irregular domains, but their accuracy applications to partial differential equations seems quite different, and may require an extensive

grid	arc-length computed	error	order
20^2	6.225879	5.73×10^{-2}	
40^2	6.270235	1.29×10^{-2}	2.15
80^2	6.280100	3.08×10^{-3}	2.06
160^2	6.282432	7.54×10^{-4}	2.03

Table 3. Accuracy of Heaviside function in integration

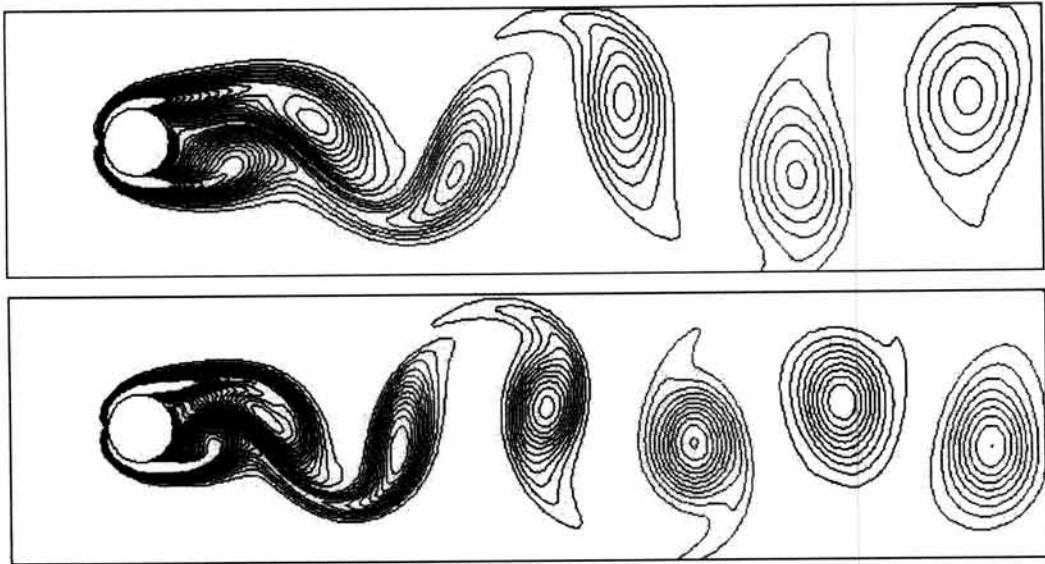


Figure 1. Vorticity contours for $Re = 100$ (top) and $Re = 200$ (bottom) cases

trial and investigation. We discretize the Heaviside function as the length fraction for all the following examples.

5.3. Accuracy of Heaviside function in integration. $\phi = \sqrt{x^2 + y^2} - 1$ on a domain $[-2, 2]^2$. The arc-length of the circle is measured as $\int_{\Gamma} 1 = \int_{[-2, 2]^2} \nabla H \cdot \nabla \phi \simeq \sum_{i,j} (D_x H \cdot D_x \phi + D_y H \cdot D_y \phi)_{i,j}$.

5.4. Flow past circular cylinder.

5.5. Floating rectangle. We consider inviscid two-phase flows with physical densities for water and air. domain : $[0, 1]^2$.

5.6. Emerging disk. A rectangular domain $[-0.2, 0.2] \times [-0.2, 0.4]$ is filled with water such that $\phi^{air} = -y + 0.2$. A solid of circular shape is in the middle of the water, $\phi = \sqrt{x^2 + y^2} - 0.07$. The fluid and solid are initially motionless. The density

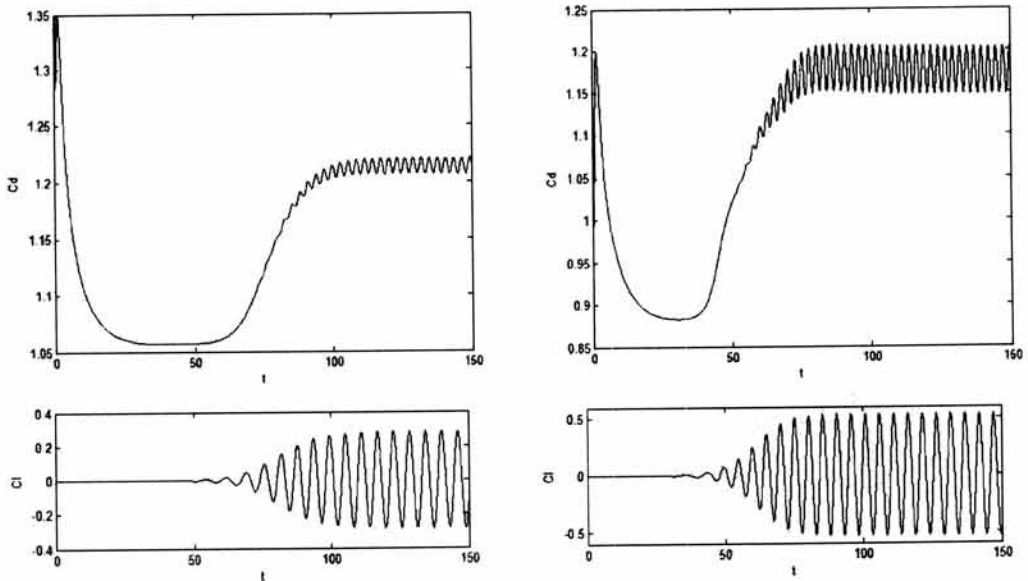


Figure 2. Temporal variations of drag and lift coefficients for $Re = 100$ (left) and $Re = 200$ (right) cases

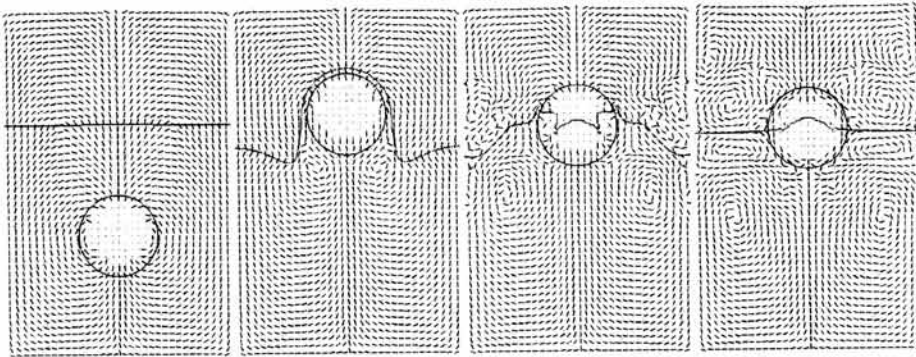


Figure 3. the result of the emerging disk: $t = 0, 0.4, 1.0$ and 5.0 from the left. The blue curve is the interface between solid and fluids, and the red one is between water and air.

of solid is 400 that is smaller than that of water, so that it emerges upward breaking the air-water interface. Figure 3 shows the simulation calculated in a 100×150 grid. As seen in figure 4, the disk fluctuates on the water surface and becomes stationary at around $t = 5.0$.

5.7. Emerging sphere. A rectangular domain $[-0.2, 0.2] \times [-0.2, 0.4] \times [-0.2, 0.2]$ is filled with water such that $\phi^{air} = -y + 0.2$. A solid of spherical shape is in the middle of the water, $\phi = \sqrt{x^2 + y^2 + z^2} - 0.07$. The fluid and solid are initially

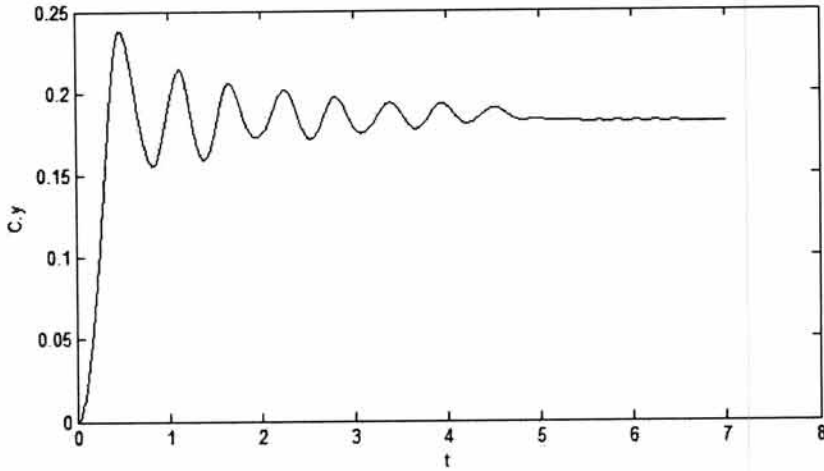


Figure 4. the oscillation of the emerging disk

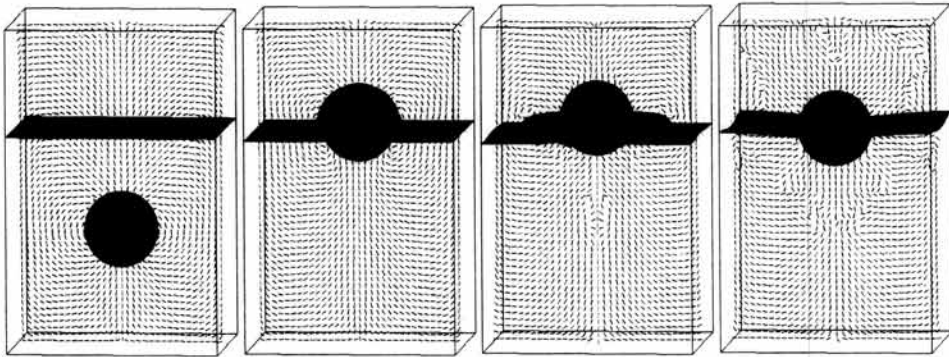


Figure 5. the result of the emerging sphere: $t = 0, 0.28, 0.6$ and 5.0 from the left. The blue surface is the interface between solid and fluids, and the red one is between water and air. The vector field is drawn on the center of the y -section.

motionless. The density of solid is 400 that is smaller than that of water, so that it emerges upward breaking the air-water interface. Figure 3 shows the simulation calculated in a $100 \times 150 \times 100$ grid. As seen in figure 6, the sphere fluctuates on the water surface with amplitude getting smaller in time.

5.8. Floating cylinder. A rectangular domain $[-0.2, 0.2] \times [-0.2, 0.4] \times [-0.2, 0.2]$ is filled with water up to level $y = 0$. A cylinder-shaped solid is floating in the water. Its density is 500, that is a half of that of water, and its center is at the water level, so that the gravity force balances out with the buoyancy force. A water ball of radius 0.07 and center $(0.1, 0.2, 0)$ is up in the air. The initial level functions are given as

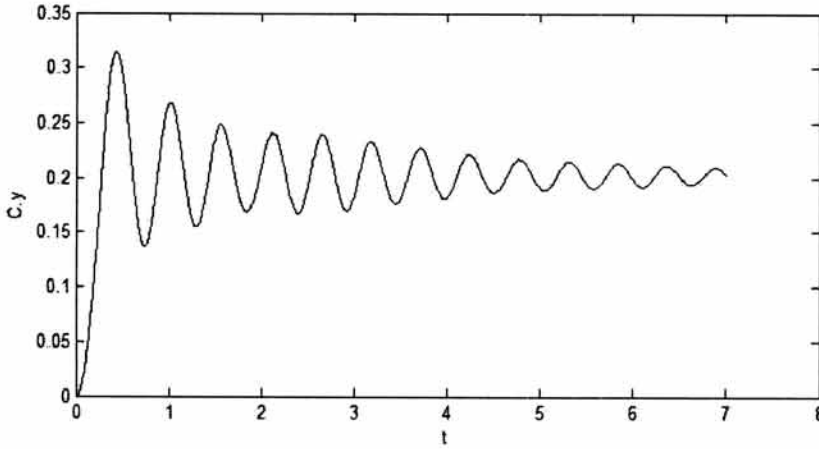


Figure 6. the oscillation of the emerging sphere

$$\begin{aligned}\phi^{air}(x, y, z) &= \max\left(-y, .07 - \sqrt{(x - 0.1)^2 + (y - 0.2)^2 + z^2}\right) \\ \phi(x, y, z) &= \min\left(\max(x - a, r - b), \sqrt{(x - a)^2 + (r - b)^2}\right) - 0.02,\end{aligned}$$

where $r = \sqrt{y^2 + z^2}$, $a = 0.08$ and $b = 0.03$. Figure 7 shows the simulation calculated in a $100 \times 150 \times 100$ grid. As the water ball collides with the cylinder, the water-air interface violently fluctuates, and then the fluid diffusion calms down the fluctuation as time goes on. Figure 8 shows the balance between the potential energy $U(t)$ and the kinetic energy $K(t)$. The decrease of potential energy leads to the increase of kinetic energy, and vice versa. Kinetic energy is measured as described in theorem 3.1, and potential energy is measured as follows.

$$U(t) = - \sum_{ijk} (\rho_{ijk} g H_{ijk} (X_{ijk} \cdot g)) - m(g \cdot C)$$

Here X_{ijk} is the grid node of index (i, j, k) , and H_{ijk} is averaged from $H_{i \pm \frac{1}{2}jk}$, $H_{ij \pm \frac{1}{2}k}$, and $H_{ijk \pm \frac{1}{2}}$. Analytically, the total energy dissipates proportionally to the magnitude of strain-rate tensor.

$$\begin{aligned}\frac{d}{dt}(U(t) + K(t)) &= \frac{d}{dt} \left(- \int \rho H g \cdot x - m g \cdot C \right) \\ &\quad + \frac{d}{dt} \left(\frac{1}{2} \int \rho H U^2 + \frac{1}{2} m^{-1} P^2 + \frac{1}{2} M^{-1} L^2 \right)\end{aligned}$$

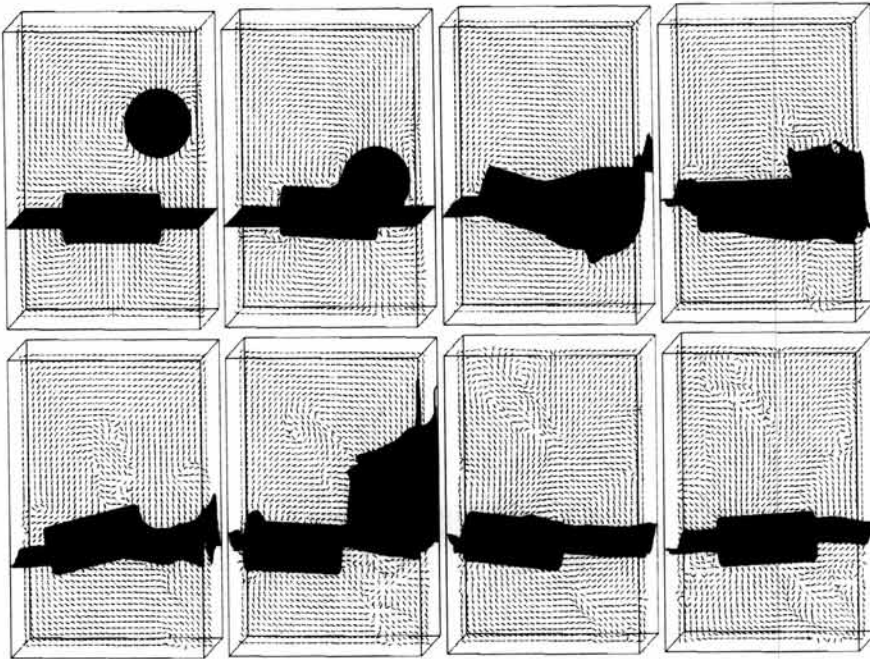


Figure 7. the result of the floating cylinder : $t = 0, 0.16, 0.33, 0.56, 0.72, 1.0, 2.0$ and 3.0 from the left and from the top. The blue surface is the interface between solid and fluids, and the red one is between water and air. The vector field is drawn on the middle of the y -section.

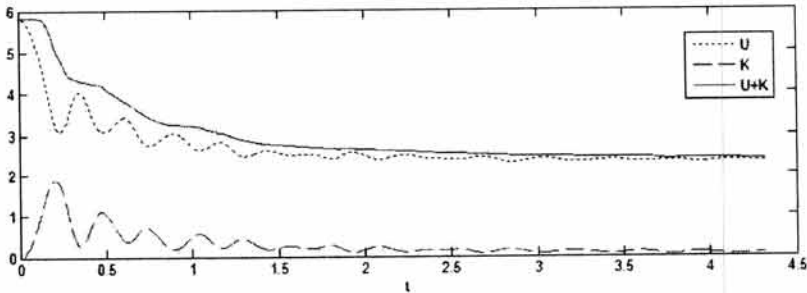


Figure 8. potential and kinetic energy in the floating cylinder

$$\begin{aligned}
 &= \int \rho H U \cdot (U_t + (U \cdot \nabla) U - g) + m^{-1} P \cdot \int (-p + 2\mu D) \nabla H \\
 &\quad + M^{-1} L \cdot \int (x - C) \times (-p + 2\mu D) \nabla H \\
 &= - \int \mu H \|D\|_{Fr}^2
 \end{aligned}$$

Here $\|D\|_{Fr}$ denotes the Frobenius norm. Coping with the analysis, the total energy in figure 8 keeps decreasing and becomes stationary as fluid gradually becomes

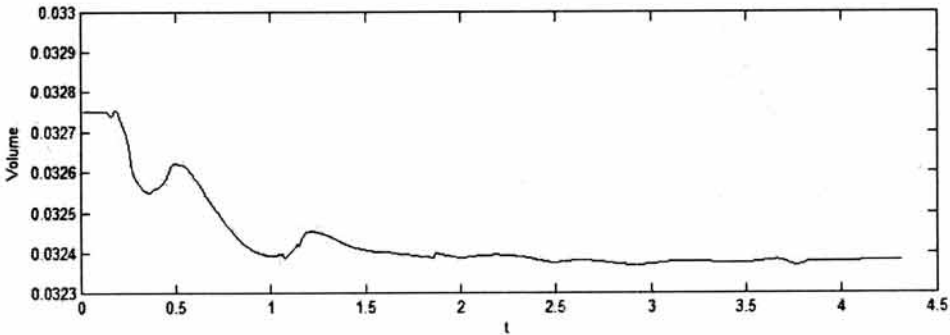


Figure 9. volume of water in the floating cylinder

at rest. In figure 9, the volume of water is lost or gained by numerical errors, but the error is less than 1.2% of the initial volume.

Acknowledgments. Myungjoo Kang was supported by Ministry of Culture, Sports and Tourism(MCST) and Korea Creative Content Agency(KOCCA) in the Culture Technology(CT) Research & Development Program. Chohong Min was supported by the Priority Research Centers Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0028298) and by the Korea Research Foundation Grant funded by the Korean Government (KRF-2011-0013649).

REFERENCES

1. C. Batty, F. Bertails & R. Bridson: A fast variational framework for accurate solid-fluid coupling. *ACM Trans. Graph. (SIGGRAPH Proc.)* **26** (2007), no. 3.
2. J. B. Bell, P. Colella & H. M. Glaz: A second order projection method for the incompressible Navier-Stokes equations. *J. Comput. Phys* **85** (1989), 257-283.
3. A. Chatterjee & A. Ruina: A new algebraic rigid body collision law based on impulse space considerations. *J. Appl. Mech.* **65** (1998), 939-950.
4. A. Chorin: A Numerical Method for Solving Incompressible Viscous Flow Problems. *J. Comput. Phys.* **2** (1967), 12-26.
5. A. du Ch en  & F. Gibou: Second-order accurate computation of curvatures in a level set framework using novel high-order reinitialization schemes. *J. Sci. Comput. (in press)* **35** (2008), 114-131.
6. W. E & J. G. Liu: Gauge method for viscous incompressible flows. *Comm. Math. Sci.* **1** (2003), 317-332.
7. K. Erleben, J. Sporring, K. Henriksen & H. Dohlmann: *Physics-based animation*. Charles river media, 2005.

8. J. Kim & P Moin: Application of a fractional-step method to incompressible Navier-Stokes equations. *J. Comput. Phys.* **59** (1985), 308-323.
9. L. D. Landau & E. M. Lifshitz: *Mechanics*. Butterworth-Heinemann, 1976.
10. C. Min: *Numerical analysis and scientific computing*. Chungmoongak, Korea, 2010.
11. ———: On reinitializing level set functions. *J. Comput. Phys.* **229** (2010), 2764-2772.
12. C. Min & F. Gibou: A second order accurate level set method on non-graded adaptive Cartesian grids. *J. Comput. Phys.* **225** (2007), 300-321.
13. ———: Robust second order accurate discretizations of the multi-dimensional heaviside and dirac delta functions. *J. Comput. Phys.* **227** (2008), 9686-9695.
14. ———: An efficient fluid-solid coupling algorithm for single-phase flows. *J. Comput. Phys.* **228** (2009), 8807-8829.
15. S. Osher & J. Sethian: Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Comput. Phys.* **79** (1998), 12-49.
16. J.W. Purvis & J.E. Burkhalter: Prediction of critical mach number for store configurations. *AIAA J.* **17** (1979), 1170-1177.
17. A. Robinson-Mosher, T. Shinar, J. Gretarsson, J. Su & R. Fedkiw: Two-way coupling of fluids to rigid and deformable solids and shells. *ACM Trans. Graph.* **27** (2008), no. 46.
18. G. Russo & P. Smereka: A remark on computing distance functions. *J. Comput. Phys.* **163** (2000), 51-67.
19. D. Shepard: A two-dimensional interpolation function for irregularly spaced data. In *Proc. of the 23rd ACM Nat. Conf.* (1968), 517-523.
20. M. Sussman, E. Fatemi, P. Smereka & S. Osher: An improved level set method for incompressible two-phase flows. *Computers and Fluids* **27** (1998), 663-680.
21. M. Sussman, P. Smereka & S. Osher: A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.* **114** (1994), 146-159.
22. J. Towers: Finite difference methods for approximating heaviside functions. *J. Comput. Phys.* **228** (2009), 3478-3489.
23. D. Xiu & G. Karniadakis: A semi-Lagrangian high-order method for Navier-Stokes equations. *J. Comput. Phys.* **172** (2001), 658-684.

^aDEPARTMENT OF MATHEMATICAL SCIENCES, SEOUL NATIONAL UNIVERSITY, SEOUL 151-742, KOREA

Email address: mkang@snu.ac.kr

^bDEPARTMENT OF MATHEMATICS, EWHA WOMANS UNIVERSITY, SEOUL 120-750, KOREA

Email address: chohong@ewha.ac.kr