

JOURNAL NAME

Numerical Approach for Generic Three-Phases Flow Based on Cut-cell and Ghost Fluid Methods

Son Tung Dang^a, Ernst Arne Meese^b, John Christian Morud^b and Stein Tore Johansen^{a,b}

^a Norwegian University of Science and Technology, Trondheim, Norway; ^b SINTEF Industry, Trondheim, Norway

ARTICLE HISTORY

Compiled November 13, 2018

ABSTRACT

In this paper, we introduce numerical methods that can simulate complex multiphase flows. The finite volume method, applying Cartesian cut-cells is used in the computational domain, containing fluid and solid, to conserve mass and momentum. With this method flows in and around any geometry can be simulated without complex and time consuming meshing. For the fluid region, which involves liquid and gas, the ghost fluid method is employed to handle the stiffness of the interface discontinuity problem. The interaction between each phase is treated simply by wall function models or jump conditions of pressure, velocity and shear stress at the interface. The sharp interface method "Coupled Level Set and Volume of Fluid"(CLSVOF) is used to represent the interface between the two fluid phases. This approach will combine some advantages of both interface tracking/capturing methods, such as excellent mass conservation from the volume of fluid method and good accuracy of interface normal computation from the level set function. In order to resolve interacting lines created by gas-liquid-solid, the first CLSVOF will be generated to reconstruct the interface between solid and the other materials. The second one will represent the interface between liquid and gas.

KEYWORDS

1. Introduction

Simulating multiphase and multi-material flows are one of the most challenging problems in computational fluid dynamics. It is not only because of the presence of numerous phases or materials but also due to the difficulty of interface treatment. Therefore, in order to model accurately the physical phenomena between phases or materials, it is crucial to well predict flow fields close to the interfacial layer. The first challenge is the interaction between solid and fluid phase, when solid is considered as undeformable material. In recent years, applications of Cartesian grids with immersed boundaries between solid and liquid, have become increasingly popular. Unlike body fitting methods where solid boundary is treated explicitly, the immersed boundary method handles the boundary implicitly. The boundary condition may be represented by inserting body force terms into the cell containing solid so that the non-slip condition will be satisfied.

Corresponding Author: Son Tung Dang Email: son.tung.dang@ntnu.no

fied (Balaras, 2004; Fadlun et al., 2000; Gilmanov and Sotiropoulos, 2005; Kim et al., 2001). Other method like the ghost cell approach (Ehsan Khalili et al., 2018; Luo et al., 2016; Mittal et al., 2008) defines a virtual layer of nodes, which are located inside the body and have at least one adjacent cell in the fluid computational domain. The flow field variables at the ghost nodes are computed based on these values at the neighbouring cells and the boundary condition applying on the body surface. The ghost cells then will be used for spatial discretization of the governing equations inside the fluid domain. Although this method can predict accurately for a wide variety of flows, the local mass error due to the finite difference method can cause non-physical phenomena near the immersed boundary (Seo and Mittal, 2011). Cut-cell method (Schneiders et al., 2016; Ye et al., 2001), however, discretizes the governing equations based on the finite-volume scheme. Therefore, the mass and momentum is strictly conserved, which is crucial for multiphase flow simulations. In this paper, we have employed the cut-cell method introduced in Kirkpatrick et al. (2003) to compute convective and diffusive fluxes in the Navier-Stokes equations.

The second challenge is the interaction between liquid and gas when their interface is severely deformed. Besides, the variables' discontinuity across the free surface can cause spurious oscillation in the numerical solutions. To handle these difficulties, the embedded boundary method (Wang et al., 2013) introduces separate pressure value for each phase in a cell containing interface. The relation between these pressure are derived based on the jump conditions for velocity and force. In addition, two continuity equations for each phase are solved at the same time to obtain the pressure value. Due to irregular shape of the free surface, the matrix formed by discretizing the continuity equation may not be symmetric and diagonally dominant. This will slow down the convergence. The ghost fluid method (Desjardins et al., 2008; Kang et al., 2000), instead, only assigns a single phase pressure in a cell. The other phase pressures will be defined as ghost variables, computed based on jump conditions at the interface and the neighbour cells' pressures. Moreover, a continuity equation for both of the two phases is employed to construct a Poisson equation for pressure. As reported in Vukčević et al. (2017) and Gibou et al. (2002), the matrix resulting from discretizing the Poisson equation is symmetric, and therefore, will not restrict the convergence of the solution. In this study, we adopt the ghost-fluid interpolation approach (Vukčević et al., 2017) to discretize the mass equation. For simplicity, we for now neglect the viscous discontinuity across the interface, the surface tension and related forces.

The third challenge is the sharp representation of the contacting layer between two phases. There are two common parameters which are used to capture the interface, namely a volume of fluid(VOF) field and the level set(LS) function. The VOF represents the interface through the volume fraction of one phase in a cell. The movement of the free surface, hence, will be described by a transport equation of VOF function (Hirt and Nichols, 1981). A geometric reconstruction of the interface is applied (Gueyffier et al., 1999) to compute the exact amount of fluid which will pass through cell face during a time step. Therefore, the VOF method can conserve mass accurately. However, due to the discontinuity of the VOF field over the interface, the surface normal vector or curvature can be estimated less precisely based on spacial derivatives of the VOF function. In contrast, the LS function, which defines the signed normal distance from an interface (Sussman et al., 1994), is smooth across the free surface. As a result, the normal vector and curvature can be calculated accurately. A drawback of the LS method is that it violates mass conservation. In order to improve mass conservation, several methods has been introduced, such as the particle level set method (Enright et al., 2002) or the conservative level set method (Olsson and Kreiss, 2005). In our

study, we applied the coupled volume of fluid and level set(CLSVOF) (Griebel and Klitz, 2017; Wang et al., 2009), which combines advantages of both methods. Besides, the CLSVOF scheme fit quite well to our current approach. The VOF function provides volume fraction and the LS field gives cell area for the cut-cell method.

The paper is organised as follows: the governing equations are described in section 2. Section 3 presents the numerical discretization for the mass and momentum equations, and the application of cut-cell and ghost fluid methods, and CLSVOF with a new technique for normal vector computation. The validation and discussion of the computational works are given in section 4. Finally, the paper is concluded in section 5.

2. Governing Equations

The model equations are derived based on formal volume and ensemble averaging (Cushman, 1982; Gray and Lee, 1977; Quintard and Whitaker, 1995). An important element is however that based on volume fractions, accurate boundary positions can be located and correct boundary conditions can be applied at internal as well as external boundaries. The transport equation for the mass is given as follows:

$$\frac{\partial}{\partial t} \int_{V_F} \rho dV = - \int_{S_F} \rho \mathbf{u} \cdot \mathbf{n}_F dS - \int_{A_w} \rho (\mathbf{u} - \mathbf{u}_w) \cdot \mathbf{n}_{F,w} dS \quad (1)$$

Where, wall area A_w , fluid volume V_F , fluid surface S_F , wall velocity \mathbf{u}_w , surface normal vector \mathbf{n}_F and wall normal vector $\mathbf{n}_{F,w}$ are explained in figure 1. We will use α_F as fluid fraction of the control volume. After evaluating the volume and surface integrals, Eq.1 can be rewritten as below:

$$\Delta V \frac{\partial}{\partial t} (\alpha_F \rho) = - \sum_{S_F} \rho \mathbf{u} \cdot \mathbf{n}_F A_F - \rho (\mathbf{u} - \mathbf{u}_w) \cdot \mathbf{n}_{F,w} A_w \quad (2)$$

Where, ΔV is size of the control volume V , and A_F is the fluid surface area, which can be determined based on the level set function. The term $\rho (\mathbf{u} - \mathbf{u}_w) \cdot \mathbf{n}_{F,w} A_w$ is the change of mass in the cell due to the mass flow across the wall face. For the impermeable inert and steady wall, this term will be negligible. Similarly, the momentum equation reads:

$$\begin{aligned} \frac{\partial}{\partial t} \int_{V_F} \rho \mathbf{u} dV &= \int_{V_F} \rho \mathbf{g} dV - \int_{V_F} \nabla p \mathbf{I} dV + \int_{S_F \cap A_w} \tau \cdot \mathbf{n}_F dS - \int_{S_F} \rho \mathbf{u} \mathbf{u} \cdot \mathbf{n}_F dS \\ &\quad - \int_{A_w} \rho \mathbf{u} (\mathbf{u} - \mathbf{u}_w) \cdot \mathbf{n}_{F,w} dS \end{aligned} \quad (3)$$

The volume integrals are first evaluated, $\frac{\partial}{\partial t} \int_{V_F} \rho \mathbf{u} dV = \Delta V \frac{\partial}{\partial t} (\alpha_F \rho \mathbf{u})$ and $\int_{V_F} \rho \mathbf{g} dV = \Delta V \alpha_F \rho \mathbf{g}$, where \mathbf{g} is the gravity acceleration vector. Next, we do

the surface integrals:

$$\begin{aligned} \Delta V \frac{\partial}{\partial t} (\alpha_F \rho \mathbf{u}) = & -\alpha_F \Delta V \nabla p \mathbf{I} + \sum_{S_F} \tau \cdot \mathbf{n}_F A_F + \tau_w \cdot \mathbf{n}_{F,w} A_w + \Delta V \alpha_F \rho \mathbf{g} \\ & - \sum_{S_F} \rho \mathbf{u} \mathbf{u} \cdot \mathbf{n}_F A_F - \rho \mathbf{u} (\mathbf{u} - \mathbf{u}_w) \cdot \mathbf{n}_{F,w} A_w \end{aligned} \quad (4)$$

Where, τ_w is wall shear stress. In this study, a simple formulation is used to compute this term, which is

$$\begin{aligned} \tau_w \cdot \mathbf{n}_{F,w} \cdot \mathbf{e}_x & \approx \mu \frac{u - u_w}{\Delta h} \\ \tau_w \cdot \mathbf{n}_{F,w} \cdot \mathbf{e}_y & \approx \mu \frac{v - v_w}{\Delta h}, \end{aligned} \quad (5)$$

where, Δh is the distance from velocity nodes to the wall, which is the value of the level set function in the corresponding cells. $\rho \mathbf{u} (\mathbf{u} - \mathbf{u}_w) \cdot \mathbf{n}_{F,w} A_w$ is the momentum exchange term due to mass transfer. In the case of an inert wall surface, moving through space, we will have zero contribution from this term. It can be applied for a typical fluid-structure interaction problem.

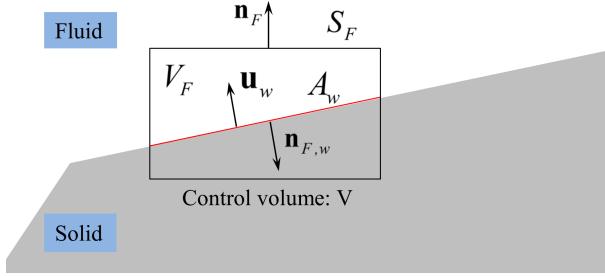


Figure 1. Control volume cut by solid

3. Numerical Methods

In figure 2, we depict a typical staggered grid layout in 2D. The level set function and volume fraction are stored at the pressure node. The boundary and solid cells are highlighted. While the location of pressure is unchanged for both standard and boundary cells, the velocity is located at the face centre of the open pressure cell surface. In this paper, we assume that the fluids are incompressible and there is no mass transfer across the interface. Therefore, Eq.2 can be discretized in time as follows:

$$\Delta V \frac{\alpha_F^{n+1} - \alpha_F^n}{\Delta t} = - \sum_{S_F} \mathbf{u}^{n+1} \cdot \mathbf{n}_F^{n+1} A_F^{n+1} \quad (6)$$

Where, n and $n + 1$ denotes the current and next time steps, Δt is the time step size. Note that the α_F^{n+1} is not computed from this equation. For a stationary wall, $\alpha_F^{n+1} = \alpha_F^n$. For a moving wall, we assign α_S^n as the solid volume fraction in the control volume

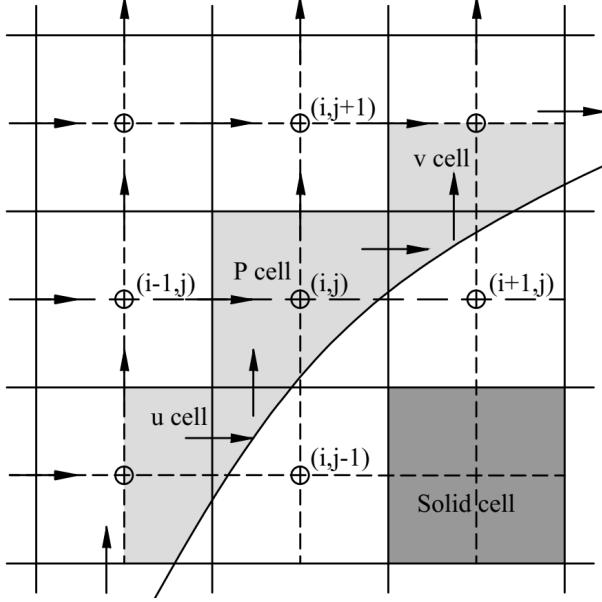


Figure 2. The staggered grid layout in 2D. The pressure node is marked by a circle(\oplus). Horizontal and vertical arrows represent the location of u and v nodes, respectively.

. It can be calculated by

$$\Delta V \frac{\alpha_S^{n+1} - \alpha_S^n}{\Delta t} = - \sum_{S_S} \mathbf{n}_{F,w} \mathbf{u}_w^{n+1} A_w^{n+1}, \quad (7)$$

where, S_S is the solid surface. In addition, we also have

$$\alpha_S + \alpha_F = 1 \quad (8)$$

As a result,

$$\Delta V \frac{\alpha_F^{n+1} - \alpha_F^n}{\Delta t} = \sum_{S_S} \mathbf{n}_w \mathbf{u}_{F,w}^{n+1} A_w^{n+1} \quad (9)$$

The predicted velocity \mathbf{u}^* can be evaluated from the Navier-Stokes equations as follows:

$$\begin{aligned} \Delta V \frac{\alpha_F^{n+1} \mathbf{u}^* - \alpha_F^n \mathbf{u}^n}{\Delta t} &= \sum_{S_F} \frac{1}{2} \left(\frac{\tau^* \cdot \mathbf{n}_F^{n+1} A_F^{n+1}}{\rho^{n+1}} + \frac{\tau^n \cdot \mathbf{n}_F^n A_F^n}{\rho^n} \right) + \frac{\tau_w^*}{\rho^{n+1}} \cdot \mathbf{n}_{F,w}^{n+1} A_w^{n+1} \\ &\quad - \sum_{S_F} \left(\frac{3}{2} \mathbf{u}^n \mathbf{u}^n \cdot \mathbf{n}_F^n A_F^n - \frac{1}{2} \mathbf{u}^{n-1} \mathbf{u}^{n-1} \cdot \mathbf{n}_F^{n-1} A_F^{n-1} \right) \\ &\quad + \Delta V \alpha_F^{n+1} \mathbf{g} \end{aligned} \quad (10)$$

In Eq.10, the convective term is treated explicitly by the second-order Adams-Bashforth scheme. Besides, the semi-implicit Crank-Nicolson scheme is employed for discretizing the diffusive terms. For the wall shear stress, a fully implicit scheme is applied to eliminate the stability restriction from this term. The pressure contribution

is absent in this formulation due to the stiffness of the moving boundary problem. The velocity at the next time step \mathbf{u}^{n+1} is calculated as follows:

$$\Delta V \alpha_F^{n+1} \frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = - \frac{1}{\rho^{n+1}} \alpha_F^{n+1} \Delta V \nabla p^{n+\frac{1}{2}} \mathbf{I} \quad (11)$$

By inserting Eq.11 to Eq.6. A Poisson-like pressure equation reads,

$$\Delta V \frac{\alpha_F^{n+1} - \alpha_F^n}{\Delta t} + \sum_{S_F} \mathbf{u}^* \cdot \mathbf{n}_F^{n+1} A_F^{n+1} = \sum_{S_F} \Delta t \frac{1}{\rho^{n+1}} \nabla p^{n+\frac{1}{2}} \mathbf{I} \cdot \mathbf{n}_F^{n+1} A_F^{n+1} \quad (12)$$

After solving Eq.12 to obtain $p^{n+\frac{1}{2}}$, new velocity \mathbf{u}^{n+1} can be computed by applying Eq.6

3.1. Cut-cell method

3.1.1. Convective flux

In this section, we present a method to estimate the convective flux for the u -momentum equation. This method can be used for the v -momentum equation as well. Based on the cut-cell method in Kirkpatrick et al. (2003) the convective flux in the cut cell is calculated for the u -momentum equation as follows:

$$\begin{aligned} F_{conv} &= (u^2 A_F) && \text{for } x \text{ direction} \\ F_{conv} &= (uv A_F) && \text{for } y \text{ direction} \end{aligned} \quad (13)$$

In the x direction, for the standard cell, a typical central interpolation is used to compute the velocity at the centre of the cell face

$$u_e = [(1 - \theta) u_P + \theta u_E], \quad (14)$$

where, $\theta = \frac{\Delta x_e}{\Delta x_E}$. For a boundary cell the interpolated velocity is slight off the centre of the cell face as shown in the figures 3 and 4. Therefore, a modification is needed to correct the velocity at this position.

$$u_{ec} = \alpha_c (u_e - u_b) + u_b \quad (15)$$

with $\alpha_c = \frac{h_{ec}}{h_e}$. In y direction, v_n at the north face is calculated as:

$$v_n = [(1 - \theta_e) v_{ne} + \theta_w v_{nw}], \quad (16)$$

where, $\theta_e = \frac{\Delta x_w}{\Delta x_{we}}$ and $\theta_w = \frac{\Delta x_e}{\Delta x_{we}}$. Velocity u_n at the north face is given as:

$$u_n = [(1 - \theta) u_P + \theta u_N] \quad (17)$$

The correct velocity at the cell face centre reads

$$u_{nc} = \alpha_c (u_n - u_b) + u_b, \quad (18)$$

with $\alpha_c = \frac{h_{nc}}{h_n}$.

3.1.2. Diffusive flux

The diffusive flux for the u -momentum equation is given as follows:

$$F_{diff} = \begin{cases} \left(\mu A_F \frac{\partial u}{\partial x} \right) & \text{for } x \text{ direction} \\ \left(\mu A_F \frac{\partial u}{\partial y} \right) & \text{for } y \text{ direction} \end{cases} \quad (19)$$

As seen in figure 5, the vector, connecting new velocity locations E and P , may not be perpendicular to the cell face. Therefore, a modification from a convectional central difference is needed in order to compute the derivative $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$ at the cell face. Taking the derivative of u along the vector \mathbf{S} gives

$$\frac{\partial u}{\partial \mathbf{S}} = s_x \frac{\partial u}{\partial x} + s_y \frac{\partial u}{\partial y}, \quad (20)$$

where, s_x and s_y are component of \mathbf{S} . Using the central difference to approximate $\frac{\partial u}{\partial s}$ yields

$$\frac{u_E - u_P}{|\mathbf{S}|} \approx s_x \frac{\partial u}{\partial x} + s_y \frac{\partial u}{\partial y} \quad (21)$$

Therefore,

$$\frac{\partial u}{\partial x} \approx \frac{1}{s_x} \left(\frac{u_E - u_P}{|\mathbf{S}|} - s_y \frac{\partial u}{\partial y} \right), \quad (22)$$

with

$$\frac{\partial u}{\partial y} \approx n_y \frac{(u_e - u_b)}{h_e} \quad (23)$$

Where, n_y is a component of normal vector \mathbf{N} at the surface which passes through e . The velocity u_e is evaluated by Eq.14. As a result, Eq.22 becomes

$$\frac{\partial u}{\partial x} \approx \frac{u_E - u_P}{s_x} - \left[\frac{(1 - \theta) u_P + \theta u_E - u_b}{s_x h_e} s_y n_y \right] \quad (24)$$

3.1.3. Small cell problem

The presence of an interface creates several velocity cells which connect to only one pressure cell. Those cells are marked as small cells (slave cells) and are linked to master cells as shown in figure 6. The details of this method was presented in Kirkpatrick et al. (2003).

3.1.4. Extension of cut-cell method for two-fluid flow

As reported in (Tryggvason et al., 2011, chapter 4), the cut-cell method can be applied in fluid-fluid interaction directly. In line with this suggestion, we apply the present cut-cell approach for two-fluid flow. We adopt the embedded boundary method(EBM) in

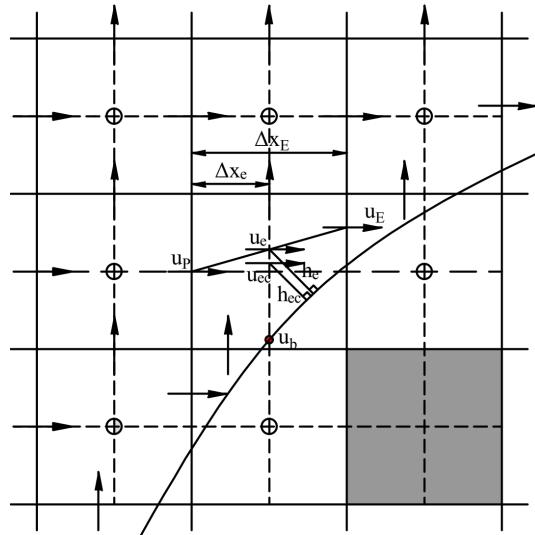


Figure 3. The snapshot of interpolation and correction method in x direction

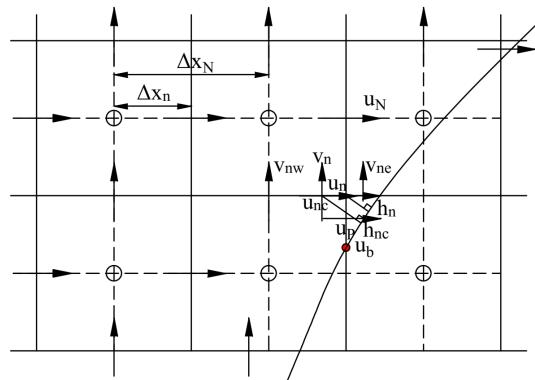


Figure 4. The snapshot of interpolation and correction method in y direction

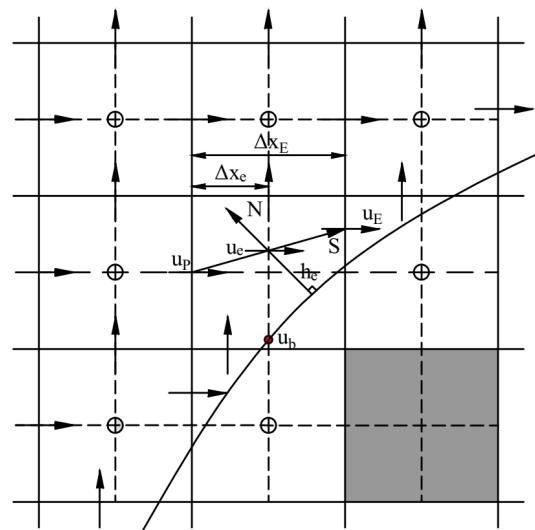


Figure 5. The vector \mathbf{S} connects two cells and the normal vector \mathbf{N} from the surface through the point e

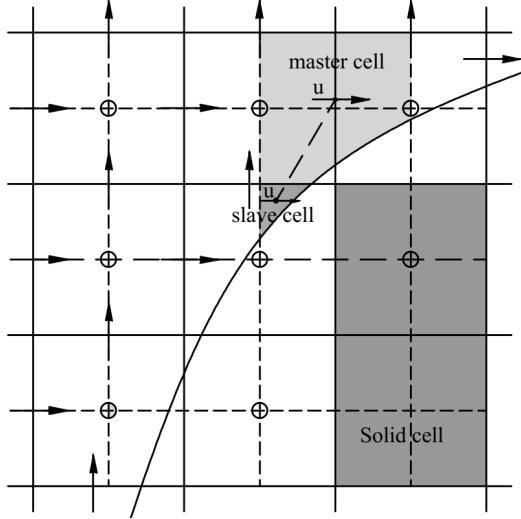


Figure 6. Linking between a slave cell and a master cell

Wang et al. (2013) to solve the Poisson-like pressure equation. We assign separate flow variables for each fluid as shown in figure 7. The predicted velocities for each phase are obtained from the momentum equations. The jump conditions across the interface are used to find a relation between fluid pressures. Then, the mass equations are solved for each fluid to compute the corrected value of pressures and velocities. However, the resulted matrix from discretizing the mass equation is non-symmetric. In addition, the huge variation in fluid densities cause the matrix coefficients to vary significantly. Therefore, the linear system is very stiff and difficult to solve by well-known methods such as BiCGSTAB or GMRES. Furthermore, it is non-trivial to construct a polynomial for evaluating the pressure gradient at the free surface. We could not find a solution for these challenges. As a result, we choose a ghost fluid method as an alternative approach.

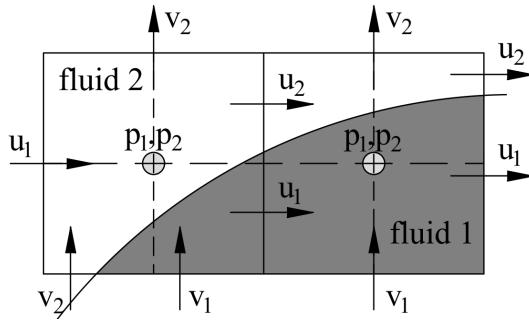


Figure 7. Location of flow variables for fluid 1(u_1, v_1, p_1) and fluid 2(u_2, v_2, p_2). The velocities locate at face centre of corresponding fluid cells. The two fluid pressures are placed at the same location

3.2. Ghost-fluid method

The presence of gas phase in V_F can create some problems, because each pressure cell carries only one pressure value. When we try to solve the Poisson-like pressure equation for the cell containing gas and liquid, the same pressure gradient will be applied for both phases. However, the huge difference between liquid and gas densities

will cause large variation in the corrected velocity. To tackle this problem, a ghost fluid method is adopted. The idea behind this approach is that we give the distinct pressure gradient for each phase. This gradient is computed basing on a ghost pressure value at the other phase pressure cell. By using the interface jump conditions, the ghost value can be derived from the actual pressure locating at that cell. For simplicity, we present the method as $1D$, while an extension for $2D$ and $3D$ can be done by a similar technique. We define the fluid level set function ϕ_F , which denotes the free surface as $\phi_F = 0$. We assume that the centroid of cell i is inside the gas field($\phi_F > 0$) and the centroid of cell $i + 1$ is inside the liquid field($\phi_F < 0$).

$$\begin{aligned}\frac{u_{i,l}^{n+1} - u_{i,l}^*}{\Delta t} &= - \frac{1}{\rho_l} \frac{p_{l,i+1} - p_{l,i}^{ghost}}{\Delta x} \\ \frac{u_{i,g}^{n+1} - u_{i,g}^*}{\Delta t} &= - \frac{1}{\rho_g} \frac{p_{g,i+1}^{ghost} - p_{g,i}}{\Delta x}\end{aligned}\quad (25)$$

$p_{l,i}^{ghost}$ and $p_{g,i+1}^{ghost}$ are ghost values of liquid and gas pressure at adjacent cells. Without the present of mass transfer between two phases, the velocity is continuous across free surface. Therefore, the velocity jump condition at the interface yields

$$[u^{n+1}]_I = u_l^{n+1} - u_g^{n+1} = 0 \quad (26)$$

In the homogeneous model, the predicted gas and liquid velocities are identical($u_g^* = u_l^*$). As a result, inserting Eq.26 to Eq.25 gives

$$\frac{p_{l,i+1} - p_{l,i}^{ghost}}{\rho_l} - \frac{p_{g,i+1}^{ghost} - p_{g,i}}{\rho_g} = 0 \quad (27)$$

The liquid and gas pressure at interface are calculated as follows:

$$\begin{aligned}p_{l,I} &= p_{l,i+1} - \left(p_{l,i+1} - p_{l,i}^{ghost} \right) \frac{|\phi_{F,i+1}|}{|\phi_{F,i}| + |\phi_{F,i+1}|} \\ p_{g,I} &= p_{g,i} + \left(p_{g,i+1}^{ghost} - p_{g,i} \right) \frac{|\phi_{F,i}|}{|\phi_{F,i}| + |\phi_{F,i+1}|}\end{aligned}\quad (28)$$

In our study, we neglect the surface tension and the possible discontinuity of shear stress. Therefore, the pressure is continuous across the interface, which gives the pressure jump condition

$$[p]_I = p_{l,I} - p_{g,I} = 0 \quad (29)$$

From Eq.28 and Eq.29, a relation between two ghost values is expressed as follows:

$$\varepsilon p_{l,i+1} + p_{l,i}^{ghost} - p_{g,i} - \varepsilon p_{g,i+1}^{ghost}, \quad (30)$$

where, $\varepsilon = \frac{|\phi_{F,i}|}{|\phi_{F,i+1}|}$. From Eq.27 and Eq.30 the gas ghost value can be computed as:

$$p_{g,i+1}^{ghost} = \frac{(\varepsilon + 1)\rho_g p_{l,i+1} - (\rho_g - \rho_l)p_{g,i}}{\varepsilon\rho_g + \rho_l} \quad (31)$$

Inserting the gas ghost value from Eq.31 to Eq.25, the updated velocity is calculated as:

$$\frac{u_{i,g}^{n+1} - u_{i,g}^*}{\Delta t} = - \frac{\varepsilon + 1}{\varepsilon \rho_g + \rho_l} \frac{p_{l,i+1} - p_{g,i}}{\Delta x} \quad (32)$$

In this study, we assume that liquid and gas share the same velocity ($u_{i,g} = u_{i,l} = u_i$). Therefore, we can use Eq.32 for solving the pressure equation in Eq.12.

3.3. Density-based computation of convective flux

There are several proposed methods which solve the problem of the high density ratio in the momentum equations such as the ones reported in Raessi (2008), Bussmann et al. (2002) and Rudman (1998). In order to fit the Adams-Barthford scheme for the convective term, the flux correction method in Desjardins and Moureau (2010) is adopted. We denote C^n as the convective term at time step n . From Eq.10, C^n would be expressed as:

$$C^n = \sum_{S_F} \mathbf{u}^n \mathbf{u}^n \cdot \mathbf{n}_F^n A_F^n \quad (33)$$

The new convective flux is given in Desjardins and Moureau (2010)

$$\hat{C}^n = \frac{1}{\hat{\rho}^{n+1}} \sum_{S_F} \rho^n \mathbf{u}^n \mathbf{u}^n \cdot \mathbf{n}_F^n A_F^n + \frac{1}{\hat{\rho}^{n+1}} \frac{\hat{\rho}^{n+1} \mathbf{u}^n - \hat{\rho}^n \mathbf{u}^n}{\Delta t}, \quad (34)$$

where, $\hat{\rho}^n$ and $\hat{\rho}^{n+1}$ are multiphase densities in the velocity cell at time step n and $n + 1$ respectively. The previous density is calculate as $\hat{\rho}^n = \alpha_l^n \rho_l + \alpha_g^n \rho_g$ with α_g^n and α_l^n are volume fractions of gas and liquid inside the velocity cell. The new density $\hat{\rho}^{n+1}$ is updated based on the continuity equation

$$\hat{\rho}^{n+1} = \hat{\rho}^n - \frac{\Delta t}{\Delta V} \sum_{S_F} \hat{\rho}^n \mathbf{u}^n \mathbf{n}_F^n A_F^n \quad (35)$$

Note that the primary volume fraction is stored at the pressure cell. Therefore, an interpolation technique is needed to compute this parameter in the velocity cell. An example for how to compute the volume fraction in the u cell is given in figure 8. It is computed by combining the amount of fluid in the left haft of the pressure cell i and right half of the pressure cell $i + 1$. At the cell face, the density is interpolated based on the first order upwind scheme. More complex scheme can be found in Rudman (1998)

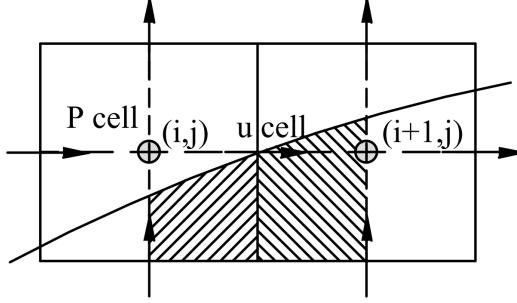


Figure 8. The volume of fluid in a u cell

3.4. Coupled level set and volume of fluid method

The transport equation for liquid volume fraction(α) and level set function(ϕ) is given as:

$$\begin{aligned}\frac{\partial \alpha}{\partial t} + \mathbf{u} \cdot \nabla \alpha &= 0 \\ \frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi &= 0\end{aligned}\quad (36)$$

And it is equivalent to

$$\begin{aligned}\frac{\partial \alpha}{\partial t} + \nabla \cdot (\mathbf{u} \alpha) &= (\nabla \cdot \mathbf{u}) \alpha \\ \frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u} \phi) &= (\nabla \cdot \mathbf{u}) \alpha\end{aligned}\quad (37)$$

Note that the RHS in Eq.37 is zero for incompressible fluids, but is retained in case of an imperfect fluid volume conservation($\nabla \cdot \mathbf{u} \neq 0$). The second order operator split in Puckett et al. (1997); Strang (1968) for the level set function and the volume of fluid, is employed in this study. Firstly, we solve for the x direction and then the y direction. In the next iteration, we switch calculation order. Time and space discretization in the x direction

$$\begin{aligned}\frac{\alpha_{i,j}^* - \alpha_{i,j}^n}{\Delta t} + \frac{u_{i+1/2,j} \alpha_{i+1/2,j}^n - u_{i-1/2,j} \alpha_{i-1/2,j}^n}{\Delta x} &= \alpha_{i,j}^* \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} \\ \frac{\phi_{i,j}^* - \phi_{i,j}^n}{\Delta t} + \frac{u_{i+1/2,j} \phi_{i+1/2,j}^n - u_{i-1/2,j} \phi_{i-1/2,j}^n}{\Delta x} &= \phi_{i,j}^* \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x},\end{aligned}\quad (38)$$

and in the y direction

$$\begin{aligned}\frac{\alpha_{i,j}^{n+1} - \alpha_{i,j}^*}{\Delta t} + \frac{v_{i,j+1/2} \alpha_{i,j+1/2}^* - v_{i,j-1/2} \alpha_{i,j-1/2}^*}{\Delta y} &= \alpha_{i,j}^* \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y} \\ \frac{\phi_{i,j}^{n+1} - \phi_{i,j}^*}{\Delta t} + \frac{v_{i,j+1/2} \phi_{i,j+1/2}^* - v_{i,j-1/2} \phi_{i,j-1/2}^n}{\Delta y} &= \phi_{i,j}^* \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y}\end{aligned}\quad (39)$$

The level set function at a cell face is calculated by taking an average of this parameter over the adjacent cells. In order to compute the volume of fluid that passes a cell face at a specific time, an interface reconstruction is employed to generate the interface

geometry within a given cell. The mathematical form of free surface reads

$$n_x x + n_y y + d = 0, \quad (40)$$

where, $\mathbf{n}(n_x, n_y)$ are interface normal vector. d is the distance from free surface to the cell centre. The interface orientation could be calculated by discretization of $\mathbf{n} = \frac{\nabla \alpha}{|\nabla \alpha|}$ or $\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|}$. However, due to the discontinuous nature of the volume of fluid, the α based computation may give an incorrect normal approximation. Therefore, the level set function which is continuous across the interface is chosen to estimate the interface normal vector. Usually, a central second order difference discretization $\nabla \phi$ is used for normal calculation. However, irregular shapes of the free surface can sometimes degenerate the solution. Figure 9 gives an example of when the central difference of $\mathbf{n} = \frac{\Delta \phi}{|\Delta \phi|}$ fails to compute the normal vector accurately at (x_i, y_j) . As we can see from the figure, the level set function at (x_{i+1}, y_j) gives a distance with respect to interface II , while the one at (x_i, y_j) gives a distance with respect to interface I . Therefore, a central difference in the x direction will give a wrong estimation of ϕ_x . We set “extrema band” as name of the points. In this study, direction difference (Macklin and Lowengrub, 2005) is employed to minimize the error in normal computing. In Macklin and Lowengrub (2005), “normal quality function” is defined as, $Q(v) = |1 - |\mathbf{v}||$, where, \mathbf{v} is the central second order difference of $\nabla \phi$. The $Q_{i,j}$ denotes the evaluation of $\nabla \phi$ at (x_i, y_j) . η is the threshold value which is used to determine whether the point (x_i, y_j) belongs to extrema band ($Q_{i,j} \geq \eta$). In the present work, $\eta = 0.1$ and $\eta = 0.075$ are used for 2D and 3D applications. The direction vector $\mathbf{D} = (D_x, D_y)$ is defined based on $Q(v)$ as follows:

$$D_x = \begin{cases} -1 & \text{if } Q_{i-1,j} < \eta \text{ and } Q_{i+1,j} \geq \eta \\ 1 & \text{if } Q_{i-1,j} \geq \eta \text{ and } Q_{i+1,j} < \eta \\ 0 & \text{if } Q_{i-1,j} < \eta \text{ and } Q_{i,j} < \eta \text{ and } Q_{i+1,j} < \eta \\ 0 & \text{if } Q_{i-1,j} \geq \eta \text{ and } Q_{i,j} \geq \eta \text{ and } Q_{i+1,j} \geq \eta \\ \text{undetermined} & \text{otherwise} \end{cases} \quad (41)$$

D_y is determined similarly. The discretization of ϕ_x is calculated by

$$\phi_x = \begin{cases} \frac{\phi_{i,j} - \phi_{i-1,j}}{\Delta x} & \text{if } D_x = -1 \\ \frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta x} & \text{if } D_x = 1 \\ \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x} & \text{if } D_x = 0 \end{cases} \quad (42)$$

For the case when D_x is underdetermined, the central difference is used instead of the method reported in Macklin and Lowengrub (2005). For the y direction, ϕ_y is computed accordingly. After obtaining the normal vector, the distance d is determined based on volume fraction α . A detail about this technique is given in Griebel and Klitz (2017). After solving Eq.38 and Eq.39, the LS function will be reinitialized as shortest distance by using the interface configuration in Eq.40. For details of the algorithm, the reader is referred to (Griebel and Klitz, 2017; Wang et al., 2009).

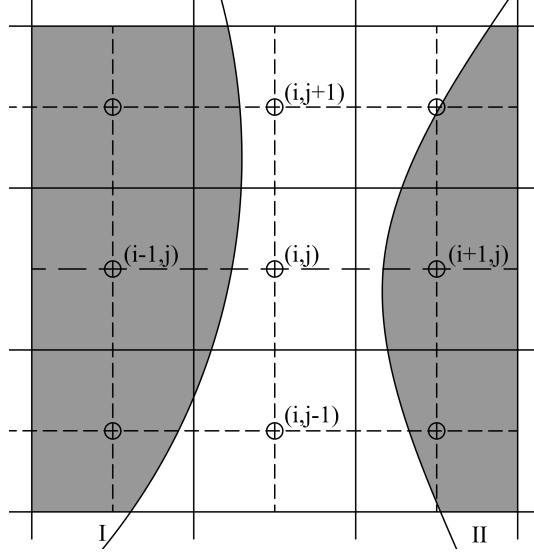


Figure 9. A scenario which give an inaccurate estimation of normal vector basing on central difference

4. Results and Discussions

4.1. Taylor-Couette flow

This test is performed to check the order of accuracy of the cut-cell method. A diagram of Taylor-Couette flow is represented in figure 10. It consists of two cylinders with a gap between them, which allows the fluid to move freely. The inner cylinder rotates with angular velocity ω , while the outer cylinder is stationary. The Taylor number Ta which presents characterization of the Taylor-Couette flow is defined by

$$Ta = \omega^2 \frac{(R_1 + R_2)(R_2 - R_1)^3}{2\nu^2}, \quad (43)$$

where, R_1 and R_2 are the inner and outer cylindrical radius respectively. In this test, we set $R_1 = 1$ and $R_2 = 4$. As reported by Dou et al. (2008), the flow fields are stable with Ta smaller than 1708. According to Cheny and Botella (2010), the exact flow fields in steady state are given as follows:

$$\begin{aligned} u^e(x, y) &= -K \left(\frac{R_2^2}{r^2} - 1 \right) (y - y_c) \\ v^e(x, y) &= K \left(\frac{R_2^2}{r^2} - 1 \right) (x - x_c) \\ p^e(x, y) &= K^2 \left(\frac{r^2}{2} - \frac{R_2^4}{2r^2} - R_2^2 \log r^2 \right), \end{aligned} \quad (44)$$

where, $K = \frac{\omega R_1^2}{R_2^2 - R_1^2}$ and $r = \sqrt{(x - x_c)^2 + (y - y_c)^2}$. The infinity norm(L_∞) and 2-norm(L_2) of the primitive variables' error is computed by

$$\|\epsilon\|_2 = \sqrt{\frac{1}{N} \sum_{i=1}^N |PV_i^n - PV_i^e|^2}, \quad \|\epsilon\|_\infty = \max_{i=1,N} |PV_i^n - PV_i^e|, \quad (45)$$

where, PV^n and PV^e are numerical and exact solution of flow field variables. In this paper, the Ta is equal to 1000 and the centre of the cylinder(x_c, y_c) is at (0.023, 0.013). The computational domain is from -5 to 5 in each direction. The grid spacing h is approximated by $10/N$, which N is grid size. The figure 4.1 shows the convergence rate of the numerical scheme for L_2 and L_∞ of the velocity field error. Whereas the current method shows second order of accuracy for the L_2 of the u and v velocity errors, the L_∞ is slightly off from the 2^{nd} order slope. The quadratic convergence is also achieved for L_2 of the pressure error, as indicated in figure 12(a). However, its L_∞ only shows a superlinear convergence.

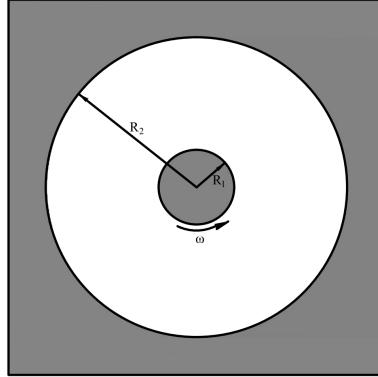


Figure 10. The schematic of Taylor-Couette flow

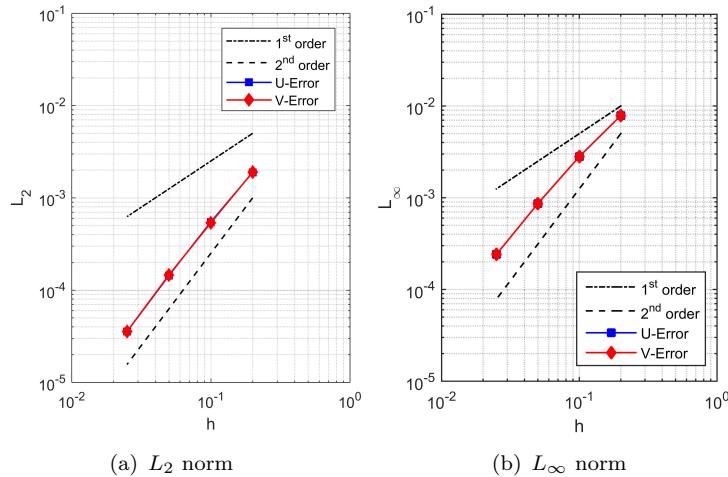


Figure 11. Calculated error norms of velocity components

4.2. Flow past a circular cylinder

The second test is aimed at examining the capability of the numerical method in terms of predicting physical quantities. The circular cylinder is chosen in this case due to numerous numerical and experimental research reported in the literature. Furthermore, its curved surface creates a wide range of cell shapes, which is notable for investigating the robustness of the cut-cell method. For cylinder flow, its solution strongly depends on Reynolds number(Re). The flow will transform from steady to unsteady state when

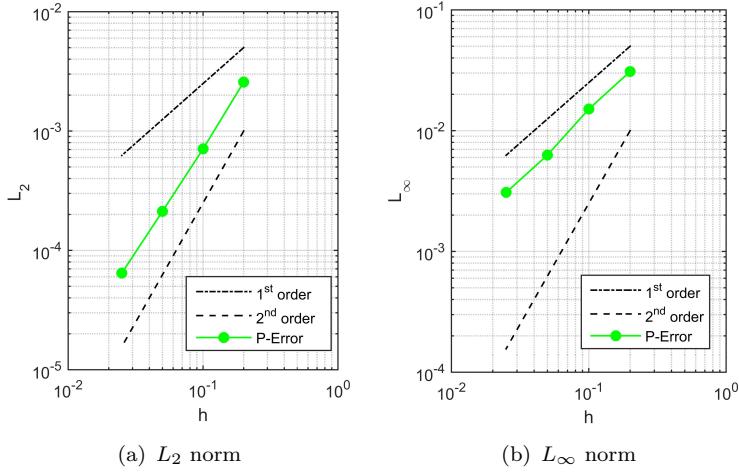


Figure 12. Calculated error norms of pressure

the Reynold number is greater than $40 - 50$ (Kirkpatrick et al., 2003). A diagram of the computational domain is given in figure 13. A cylinder of diameter D is placed at a distances $20D$ and $40D$ from inlet and outlet in order to minimize the effect of the boundary conditions on the flow field around the cylinder. Slip walls are located $10D$ from the body centre, at the sides of computational domain. For all our computations, the inlet velocity U_{inlet} is calculated based on Reynolds number, object diameter and fluid kinetic viscosity as: $U_{inlet} = \frac{Re\nu}{D}$. Due to the big ratio of computational domain to cylinder, the mesh is generated non-uniformly except for region surrounding the object. The details of grid sizes using for our simulation is given in table 1. The pressure, drag and maximum lift coefficients are calculated by this following formulation:

$$C_p = 2\frac{p - p_0}{\rho U_\infty}, \quad C_D = 2\frac{F_D}{\rho U_\infty^2 D}, \quad C_L = 2\frac{F_L}{\rho U_\infty^2 D} \quad (46)$$

Here, p_0 , U_∞ are inlet pressure and reference velocity, and F_D , F_L are drag and maximum lift force acting on the cylinder. The present results of the pressure coefficient for $Re = 40$ achieved on the group of meshes, are compared with experiment data(Grove et al., 1964) in figure 14. As seen from the figure, good agreement with these reference results is observed. In addition, the current method can predict the pressure distribution quite accurately, even for the coarse grid A. At $Re = 40$, the flow is steady and creates a vortex ring at the wake. Several characteristics of the wake such as separation angle and length are shown in table 2. A comparison between our work and other established research works from the literature indicates that our numerical scheme can predict quite accurately physical quantities at steady state. When the Reynold number reaches 100 and above, the flow become unsteady. The vortices are shed from behind the cylinder and oscillates with a time period of T_{vs} . The Strouhal number(St) are calculated by $St = \frac{fD}{U_\infty T_{vs}}$. We compare our computed results of C_D , $C_{L,max}$ and St with reference data in table 3 and 4. In general, our results fit well in the range of other simulation data reported in the literature.

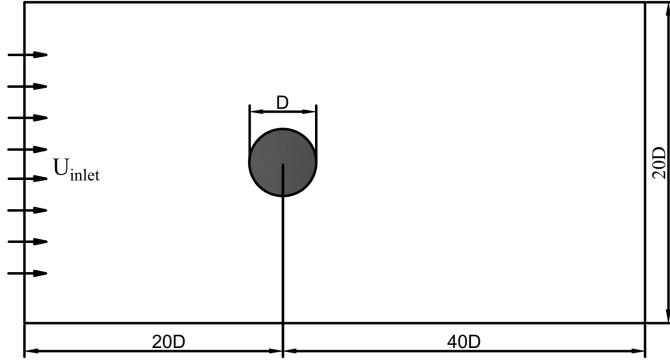


Figure 13. The computational domain

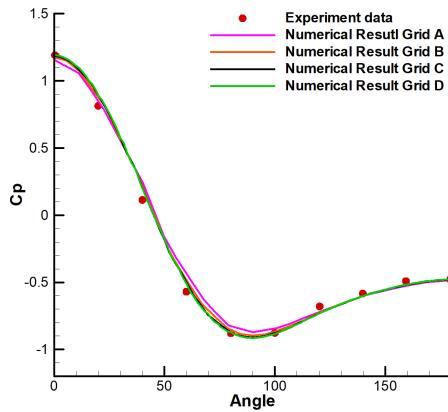


Figure 14. The pressure coefficient over cylinder at $Re = 40$

Table 1. The computational mesh

Grid	$N_x \times N_y$	Cells/Diameter
A	81×61	10
B	121×81	20
C	201×141	40
D	321×241	80

Table 2. The drag coefficient C_D , the separation angle θ and the length of recirculation bubble L/D behind cylinder at $Re = 40$

Study	C_D	θ	L/D
(Linnick and Fasel, 2005)	1.54	53.6	2.28
(Taira and Colonius, 2007)	1.54	53.7	2.30
(Kirkpatrick et al., 2003)	1.542	53.7	2.27
Present Simulation, Grid C	1.55	53.5	2.26

4.3. Transversely oscillating cylinder in a free-stream

To validate our numerical method in case of a moving boundary problem, we performed a computational simulation for flow over the cylinder with prescribed cylinder oscillation in the vertical direction. The computational configurations are chosen similarly to the previous test in section 4.2. In this section, all our computations are

Table 3. The drag coefficient C_D , the maximum lift coefficient $C_{L,max}$ and Strouhal number St at $Re = 100$

Study	C_D	$C_{L,max}$	St
(Linnick and Fasel, 2005)	1.34 ± 0.009	0.333	0.166
(King, 2007)	1.41	—	—
(He et al., 2000)	1.353	—	0.167
Present Simulation, Grid C	1.374 ± 0.01	0.337	0.169

Table 4. The drag coefficient C_D , the maximum lift coefficient $C_{L,max}$ and Strouhal number St at $Re = 200$

Study	C_D	$C_{L,max}$	St
(Linnick and Fasel, 2005)	1.34 ± 0.044	0.69	0.197
(Taira and Colonius, 2007)	1.35 ± 0.048	0.68	0.196
(He et al., 2000)	1.356	—	0.198
Present Simulation, Grid C	1.346 ± 0.046	0.7	0.196

implemented on the grid C. The translational motion of the cylinder centre (x_c, y_c) is given by:

$$x_c = 0, \quad y_c = A \cos \left(2\pi f_e \frac{U_\infty}{D} t \right), \quad (47)$$

where, $A = 0.2D$ is an oscillation amplitude, and f_e is an excitation frequency. The free-stream U_∞ is estimated so that the Reynolds number is equal to 185. Firstly, we run a simulation for the stationary cylinder in order to obtain the natural vortex shedding frequency, $f_o = 0.195$. Afterwards, the excitation frequency can be computed by: $f_e = kf_o$, which $k \in [0.8, 1.2]$. Figure 15 shows the vortex shedding formed behind the cylinder when the cylinder is at its maximum vertical position. The shed vortices behind the cylinder become numerically dispersed due to the coarser grid downstream. The time evolution of computed drag and lift coefficients for a variety of forcing frequencies is presented in figure 16. For f_e smaller than f_o , both C_D and C_L show an uniform oscillation after a certain time. When f_e is greater than f_o , the force coefficients shows a non-uniform fluctuating pattern due to the impact of the higher excitation frequency. The computed mean value of drag and lift coefficients are in figure 17 compared with other numerical results from both a cut-cell method (Cheny and Botella, 2010) and body-fitted grid (Guilmeneau and Queutey, 2002). While the root mean square of the drag coefficient shows a good agreement with the reference data, the other quantities show a slightly over-prediction. Figure 18 shows a comparison of the pressure coefficient over the cylinder surface. Clearly, our numerical result marks well with the body fitted grid (Guilmeneau and Queutey, 2002).

4.4. Flow past a sphere

This test is used to validate our cut-cell method for a 3D application. The computational domain and numerical mesh are shown in 19. The sphere is placed $20D$ from the inlet and $40D$ from the outlet, where D is sphere diameter. Four slip walls are located symmetrically at the sides of the domain. A non-uniform mesh of $121 \times 81 \times 81$ grid points is employed in this simulation. We cluster an uniform grid in the vicinity of the sphere with $20Cells/D$, so that the physical phenomena around the body can be captured accurately. As reported in Johnson and Patel (1999), for Reynolds number

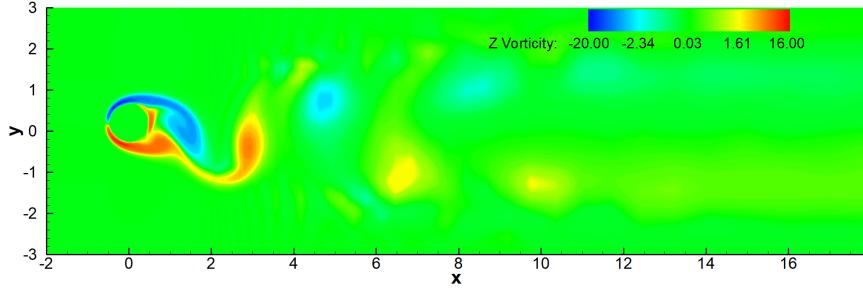


Figure 15. The vorticity contour

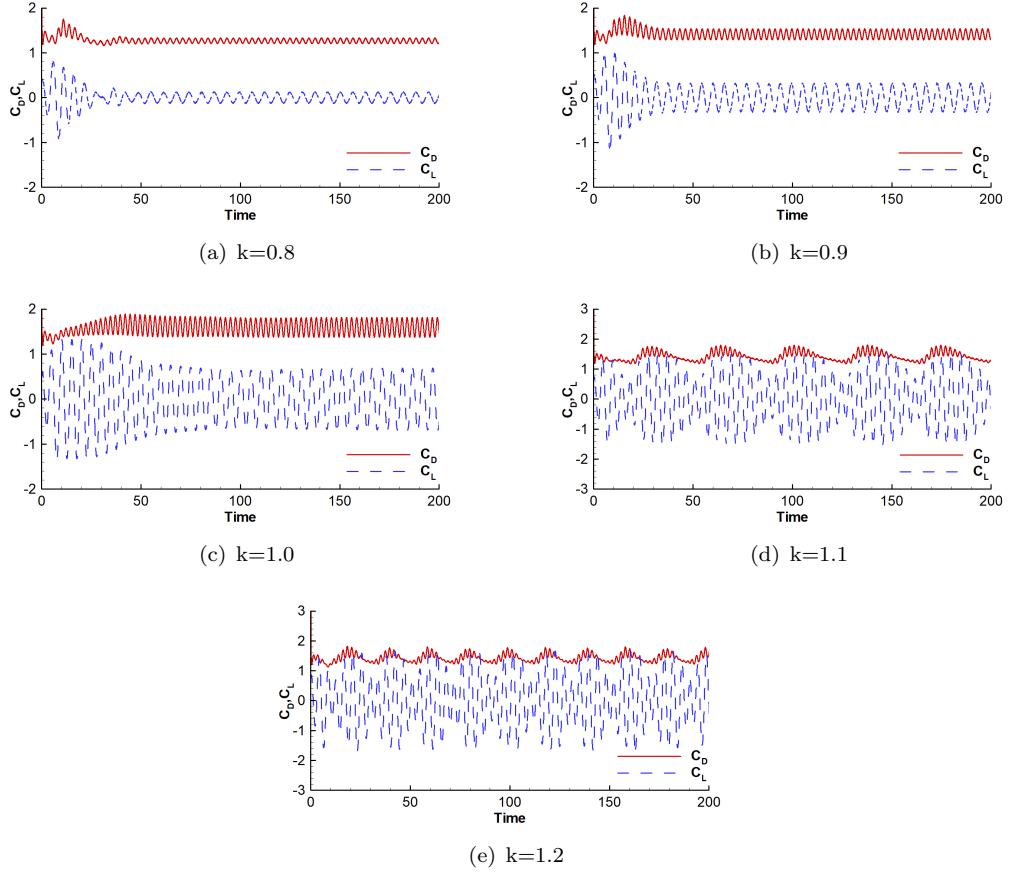


Figure 16. The drag and lift coefficient versus non-dimensional time for different ratios of f_e/f_0

smaller than 210, the flow is steady and symmetric, as can be observed from figure 20(a). A separation length L/D and vortex position(x_c, y_c) for two specific Reynolds number obtained in various studies are given in table 5. Clearly, our numerical method can predict quite accurately physical quantities for steady flow over a sphere. When the Reynolds number increases, the flow becomes unsteady and non-symmetric as indicated in figure 20(b). Furthermore, the time evolution of the drag and lift force coefficients, for $Re = 300$, are given in figure 21(a). After $t^* = 100$ these quantities show a stationary oscillation. As a result, the lift coefficient is chosen to calculate the Strouhal number. The computed result and comparison with well reported studies in the literature is presented in table 5. It can be seen that our result is very close to the

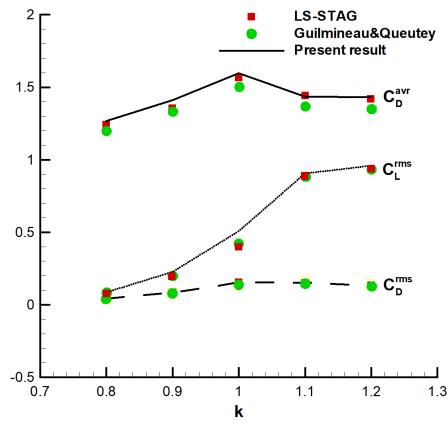


Figure 17. Computed time averaged force coefficients and a comparison with other numerical results for the test of prescribed oscillating cylinder in free stream

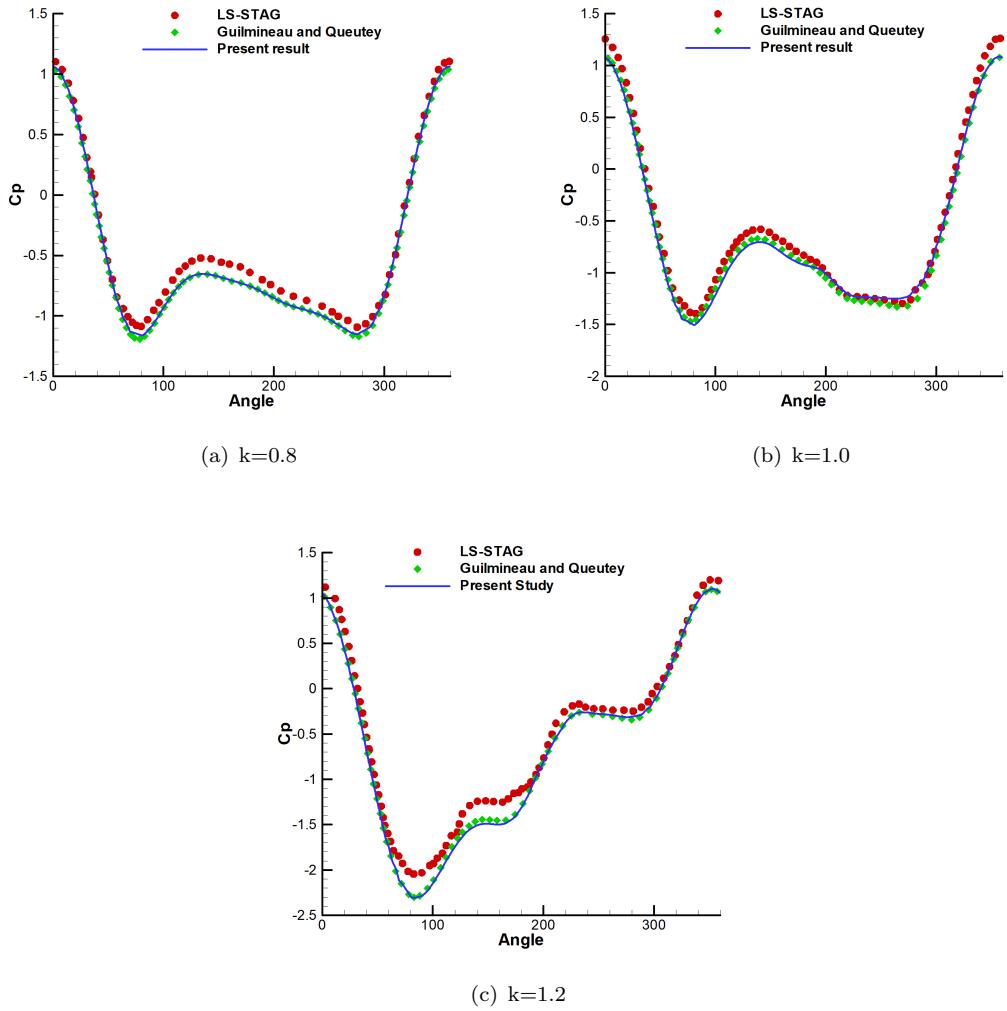


Figure 18. The pressure coefficient distributed over the cylinder's surface when it peaks the upper position

reference data. Figure 21 compares the computed drag coefficient with other numerical results (Johnson and Patel, 1999; Mittal et al., 2008) and theoretical data (Clift et al., 1978). In general, our method can predict the drag coefficient well and in agreement with previous studies.

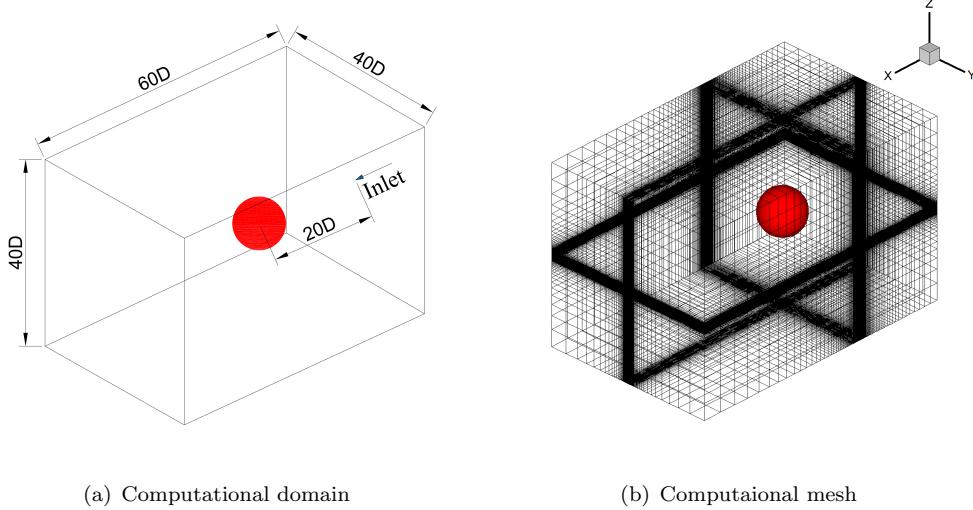


Figure 19. The computational configuration

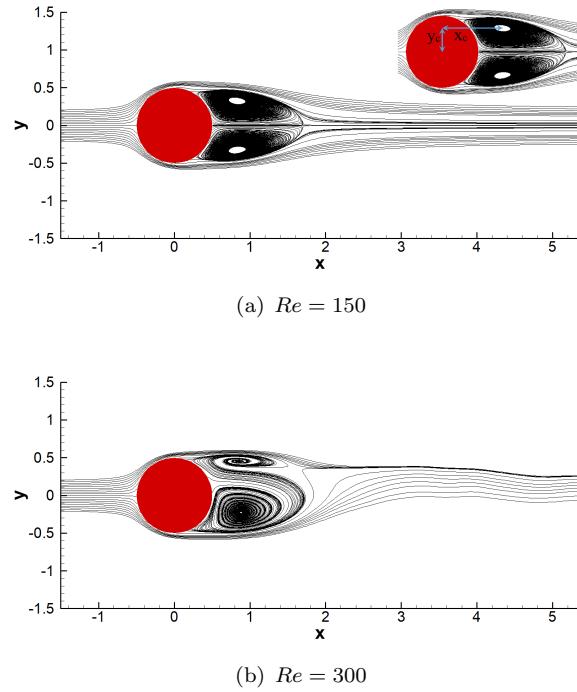
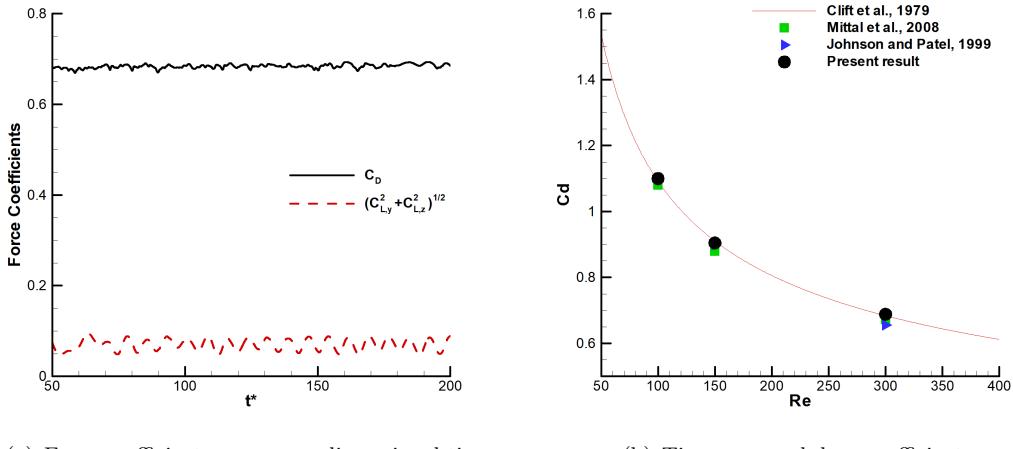


Figure 20. The streamline of flow in $X - Y$ plane

Table 5. Computed characteristic parameters and comparison with established numerical and experimental data in literature

Study	Re			150			300
	x_c/D	y_c/D	L/D	x_c/D	y_c/D	L/D	St
(Johnson and Patel, 1999)	0.75	0.29	0.88	0.83	0.33	1.2	0.137
(Taneda, 1956)	0.745	0.28	0.8	0.82	0.29	1.2	—
(Mittal et al., 2008)	0.742	0.278	0.84	0.81	0.3	1.17	0.135
Present computation	0.754	0.288	0.87	0.82	0.33	1.2	0.137



(a) Force coefficients versus nondimensional time

(b) Time averaged drag coefficients

Figure 21. Quantitative study of force coefficients

4.5. Advection tests

The tests are designed to validate the ability of CLSVOF scheme to deal with complex topological transformations. In these tests, the transport equations for the VOF and the level set fields will be solved with a specified velocity field. Therefore, the resulting accuracy will only depend on the interface tracking method itself rather than the numerical methods applied for the governing equations.

4.5.1. A Circle in 2D Deformation Field

A fluid circular disk of radius 0.15 is located with its centre at (0.5,0.75), inside a unit computational domain. A deforming velocity field is prescribed as follows:

$$\begin{aligned} u &= -\sin^2(\pi x) \sin(2\pi y) \cos(\pi t/T) \\ v &= \sin^2(\pi y) \sin(2\pi x) \cos(\pi t/T), \end{aligned} \quad (48)$$

where, $T = 8s$ is time period that the fluid disk needs to come back to its original location. Figure 22 shows the development of the interface at $t = 0s, 4s$ and $8s$ with the grid size of 128×128 . The fluid circle starts moving in the clock-wise direction from its initial position at $t = 0s$, achieves maximum stretching at half of the time period, $t = 4s$, before returning to its initial position at $t = 8s$. As shown in figure 22(b), the interface is not continuous and is broken into several small pieces at the tail. This

phenomenon is due to the limitation of the VOF field which only accepts a single interface inside a cell. Therefore, when filaments become thinner their cross section will be smaller than the grid resolution. The result is interface cracking. This problem could be reduced by refining the mesh. Deformations of the circular disc, calculated by two different schemes for the normal vector approximation, at $t = 4$ and $t = 8$, are shown in figure 23. Clearly, the new proposed scheme can preserve the interface shape better at both time marks.

A grid convergence study for the deforming flow is conducted by performing numerical computation for three different mesh sizes, which are 64×64 , 128×128 and 256×256 respectively. The level set function and area losses are examined in this study. The infinity norm (L_∞), 1-norm (L_1) and 2-norm (L_2) of the level set function errors are computed as follows:

$$\|\epsilon\|_1 = \frac{1}{N} \sum_{i=1}^N |\phi_i^n - \phi_i^e|, \quad \|\epsilon\|_2 = \sqrt{\frac{1}{N} \sum_{i=1}^N |\phi_i^n - \phi_i^e|^2}, \quad \|\epsilon\|_\infty = \max_{i=1,N} |\phi_i^n - \phi_i^e| \quad (49)$$

Here, ϕ_i^n and ϕ_i^e are the numerical and the exact level set function at $t = 8$. N is number of grid points located inside the *band_width* of cells from the interface. The formulation for area loss calculation at $t = 8$ is given as

$$AreaLoss = \left(\sum_{i=1}^{I_{max}} \sum_{j=1}^{J_{max}} \alpha_{i,j} \right) \Delta x \Delta y - \pi R^2, \quad (50)$$

where, I_{max} and J_{max} are the number of grid points in x and y direction respectively, and R is circle radius. The table 6 shows the values of error norms, area loss and their convergent orders. As we can see, the convergence rate for the level set function reduces from second order to first order when we refine grid, while it remains second order for area loss. This phenomenon is due to the mathematical technique computing the level set function from the volume of fluid, when the grid point at the farthest distance from the interface would be the less accurate.

Table 6. Computed error norms, area loss and convergence order.

Mesh	Error				Order			
	L_1	L_2	L_∞	$AreaLoss$	L_1	L_2	L_∞	$AreaLoss$
64x64	$5.945E-3$	$5.879E-4$	$5.318E-2$	$6.405E-5$	—	—	—	—
128x128	$1.186E-3$	$4.958E-5$	$8.934E-3$	$1.671E-5$	2.33	3.56	2.57	1.94
256x256	$5.271E-4$	$2.028E-5$	$5.005E-3$	$4.047E-6$	1.17	1.29	0.84	2.05

4.5.2. A Sphere in 3D Deformation Field

The next test case is implemented in three-dimensional coordinates, which is aimed to verify the capability of CLSVOF for a 3D interface stretching problem. A fluid sphere of radius 0.15 is located at (0.35,0.35,0.35) and is deformed by a flow field being given

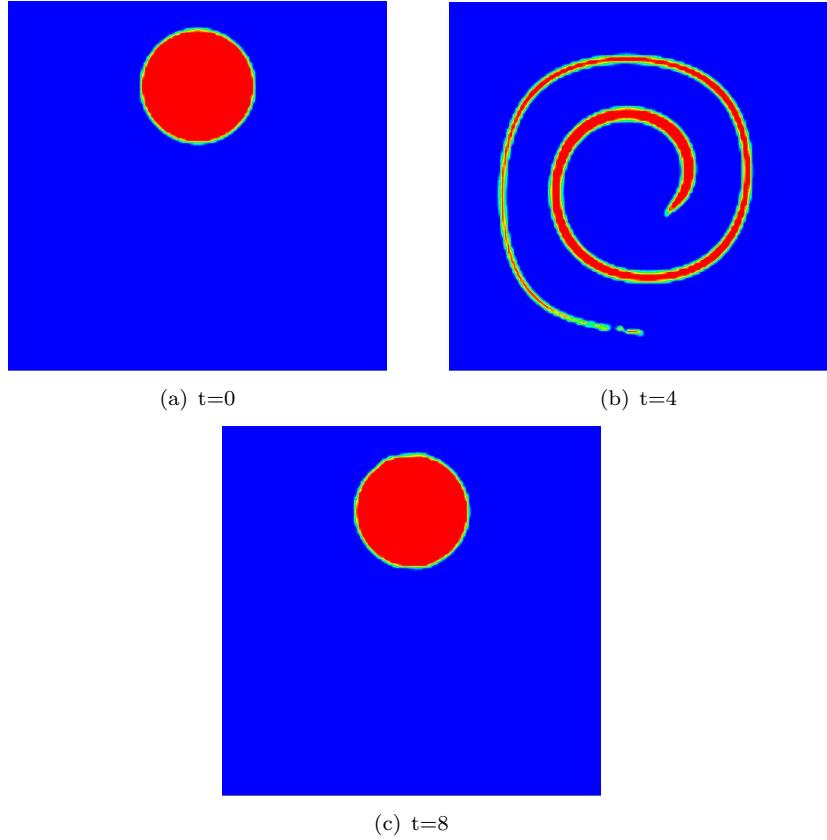


Figure 22. The evolution of circular disk versus time

as follows:

$$\begin{aligned} u &= 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) \cos(\pi t/T) \\ v &= -\sin^2(\pi y) \sin(2\pi x) \sin(2\pi z) \cos(\pi t/T) \\ w &= -\sin^2(\pi z) \sin(2\pi x) \sin(2\pi y) \cos(\pi t/T), \end{aligned} \quad (51)$$

which $T = 3s$ is the time period a sphere needed to return to its initial location. A computational domain is restricted to an unit cubic box with 128 grid points in each direction. Figure 24 illustrates the evolution of fluid interface from $t = 0s$ to $t = 3s$. In general, the numerical scheme can preserve the shape of fluid zone quite well at the end of the computation. In addition, the interface is not smooth at the half period, which is also indicated in the 2D test case.

4.6. Dam-break flow

Dam-break flow describes the instantaneous movement of a water column in a tank caused by the sudden removal of a vertical obstacle. The wave tip is driven by gravity, and travels freely downstream to a steep wall at the end of reservoir. After colliding with the wall, the wave rolls back and impacts with underlying water. The complex physical behaviour and severe interface deformation of the flow is a good test case for validating our numerical method. An overview over the simulated dam-break flow configuration is given in figure 25. The tank dimensions and the water level measuring

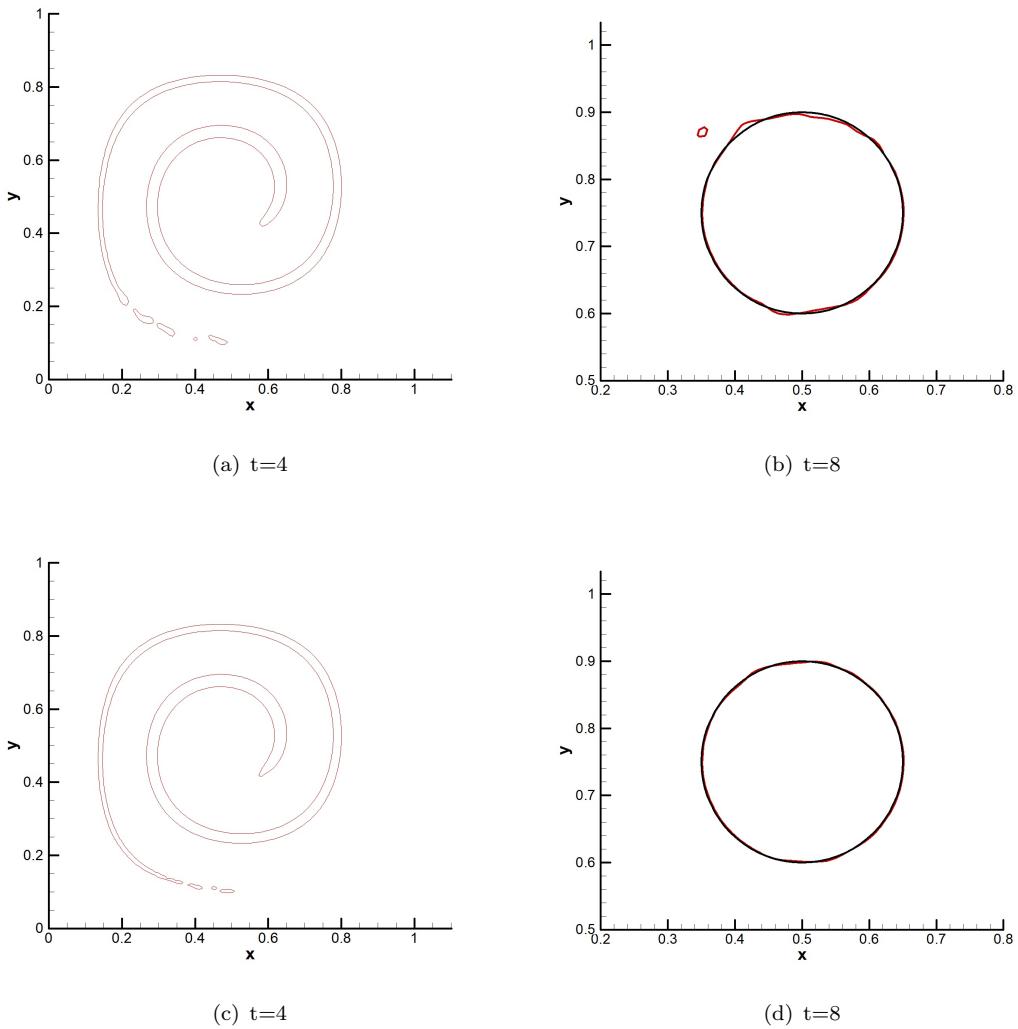


Figure 23. The comparison of interface location between two normal vector calculation schemes (a,b: central difference scheme; c,d: normal quality based scheme (Macklin and Lowengrub, 2005))

positions used in the experiment are given in figure 25(a). The computational domain is illustrated in figure 25(b), the liquid column is initialized with the height of $0.3m$ and width of $0.6m$ at the left side of the tank. The remaining reservoir is occupied by air. The liquid and air densities are $998.2kg/m^3$ and $1.205kg/m^3$, respectively. The grid size of 161×60 is employed in this computation. Figure 26 compares the free surface profile obtained by two different numerical methods calculating the convective flux. In general, a typical scheme can produce spurious interface shapes at particular moments of the simulation, as represented in figure 26(a),(b). This phenomenon could be eliminated by applying a new method in section 3.3 to compute the convective flux, as depicted in figure 26(c),(d). The time development of the wave tip, computed by our numerical method, and comparison with other results reported in the literature (Dressler, 1954; Lobovský et al., 2014; Martin et al., 1952) is given in figure 27. Clearly, our result agree well with reference data. Figure 28 shows temporal variation of water level at specific locations $H1, H2, H3$ and $H4$ (given at figure 25(a)) from current results and measured data (Buchner, 2002; Lee et al., 2001; Lobovský et al., 2014)

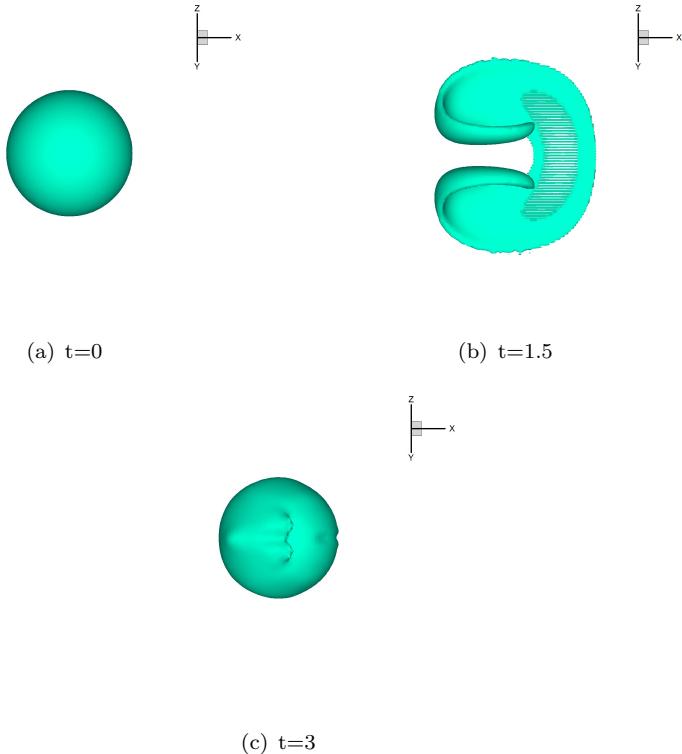
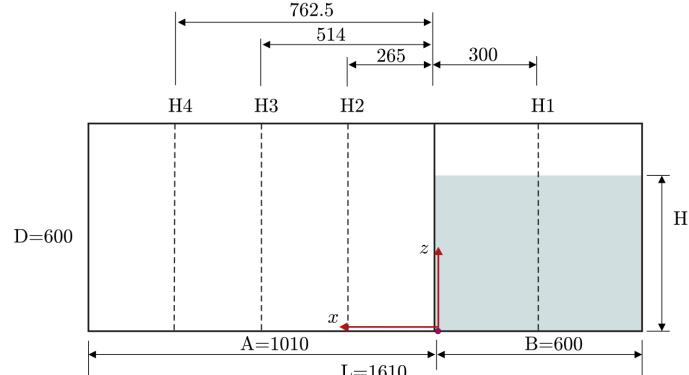


Figure 24. Snapshots of the fluid sphere versus time

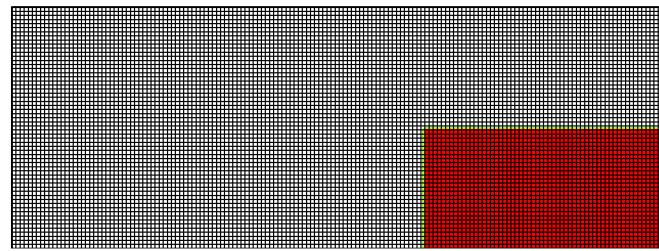
for the case of water height $H = 0.3m$. It can be seen that our present method can predict the arrival time of the first wave across the water level measurements $H2$, $H3$ and $H4$ quite accurately. Our model also predicts the development of the liquid height, after arrival of the primary wave, relatively precise. When the second wave arrives and passes through the given locations $H1$, $H2$, $H3$ and $H4$, there is difference between our computations and the results previously published in the literatures in both its arrival time and its elevation. We may note that the dam-break flow involves several complex phenomena such as turbulence or fluid-structure interaction. These phenomena differ from case to case and it affects considerably the flow field properties. As a result, it creates random variations of the elevation in the experiments and there is a need to perform ensemble averaging of experiments. In our studies which did not investigate the effects of of turbulence and fluid-solid interaction.

4.7. Water entry flow

The next test is a water entry simulation which is used to demonstrate the capability of our numerical method to handle three-phase flows. The shape of the object is given in figure 29(a). The characteristics of water entry flow are illustrated in figure 29(b). α is the wedge angle, being 90° and 120° in our computations. V is the entry velocity. For simplicity, the item is kept standstill and the free surface is let moving with a speed being equivalent to a water entry velocity. V_{so} is water jet speed, η_o is the angle between splash and a horizontal interface. The computational domain is illustrated in figure 30(a). The size of the rectangular domain is $0.51m$ of width and $0.32m$ of height, as reported in Vincent et al. (2018). The wedge width $d = 0.036m$ is held constant

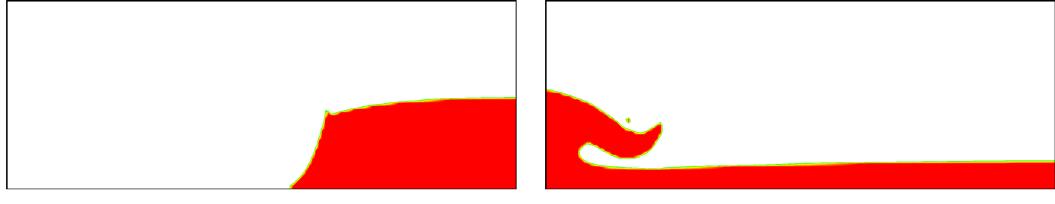


(a)



(b)

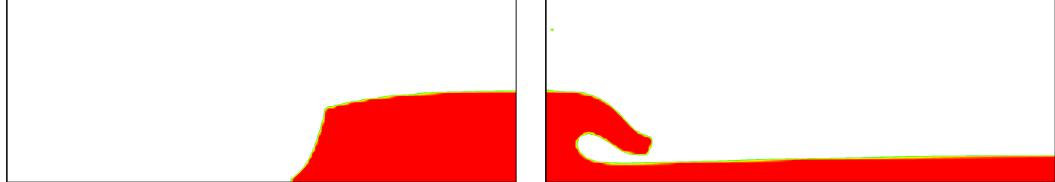
Figure 25. The diagram of dam break flow configurations (a): Experimental set-up (Lobovský et al., 2014), (b): Computational domain



(a)



(b)



(c)

(d)

Figure 26. The comparison of liquid interface computed by typical convection scheme(a,b) and applying density-based convection scheme(c,d)

for the whole simulation. The non-uniform grid of 260×180 is shown in figure 30(b). The mesh is clustered in the wedge vicinity with $100\text{cells}/d$ so that it can resolve the thin geometry of the splash. The jet velocity versus the entry velocity for the case of $\alpha = 90^\circ$ and $\alpha = 120^\circ$ is plotted in figure 31. Generally, our method overpredicts the jet velocity and underpredict the jet angle in comparison with reference data from (Vincent et al., 2018) for $\alpha = 90$. When $\alpha = 120$, our computed jet quantities fit quite well with the experimental and theoretical results. Figure 32 shows the time

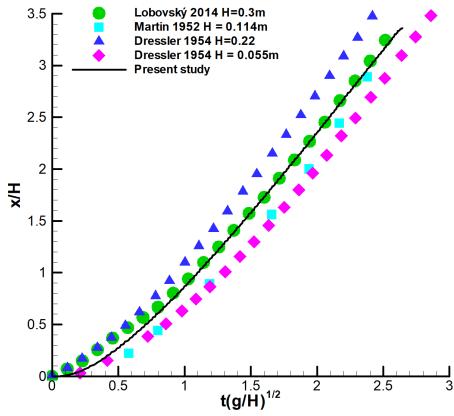


Figure 27. Temporal variation of wave tip movement

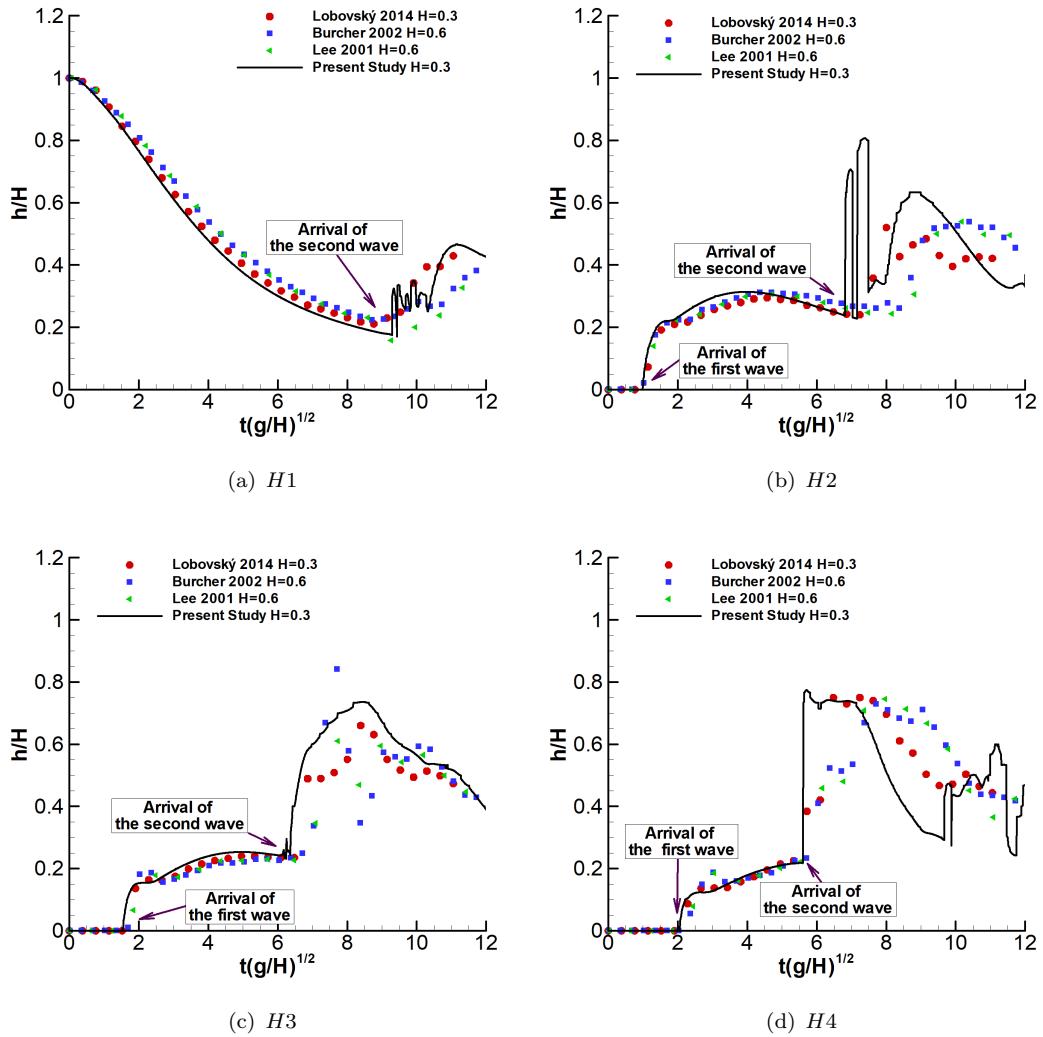


Figure 28. The water level evaluated at given positions (H_1, H_2, H_3 and H_4)

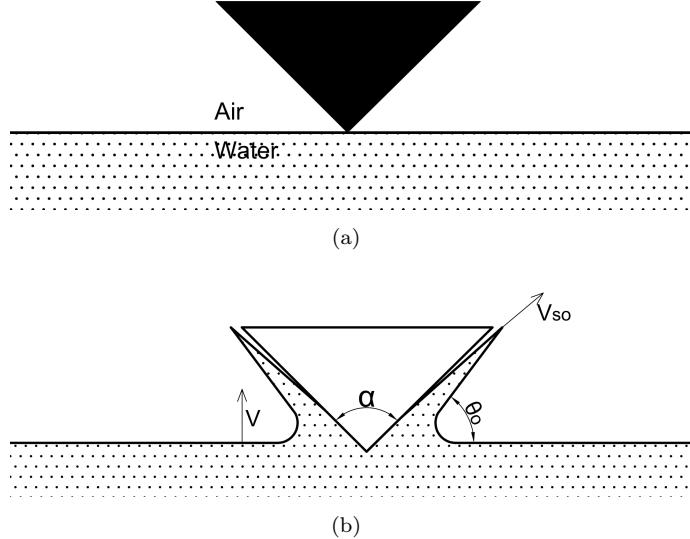


Figure 29. The schematic of water entry problem

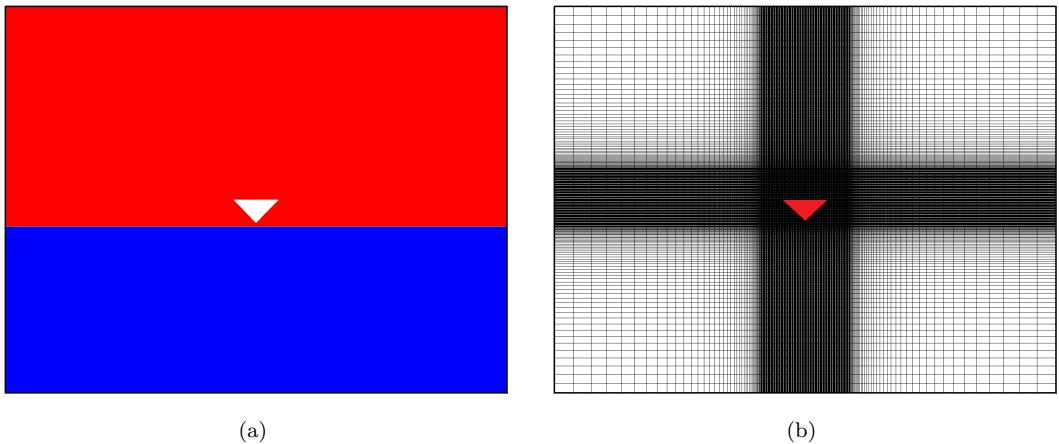


Figure 30. The snapshots of computational domain(a) and computational grid(b)

development of the splash from time of impact $t^* = 0.127$ to the moment when the wedge is fully submerged with regard to a flat water surface.

5. Conclusions

In this paper, we have introduced a numerical approach for generic three phase flow, including two fluids and one solid. The cut-cell method showed good mass conservation when it was applied to regions containing fluid and solid. In the fluid field the ghost-fluid method is adopted to resolve the discontinuity across the gas-liquid interface. In addition, the density based convection flux is employed to minimize the effect of the high density ratio(large effective density gradients) in the momentum equations. The numerical results show that the proposed scheme can reduce the unphysical deformation of a free surface. For tracking the interface, we applied the CLSVOF method which yields an excellent mass conservation. A new proposed technique for computing the normal vector indicates an improvement in preserving the interface shape.

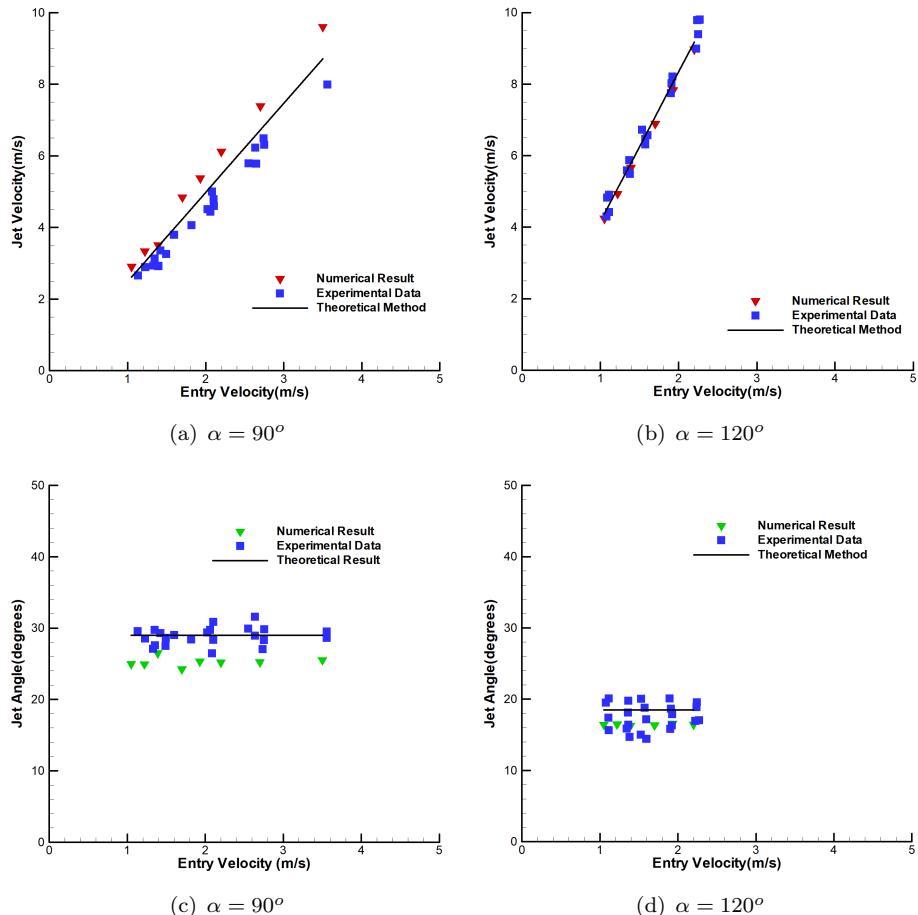


Figure 31. The jet velocity(a,b) and jet root expansion angle(c,d) in respect of entry velocity

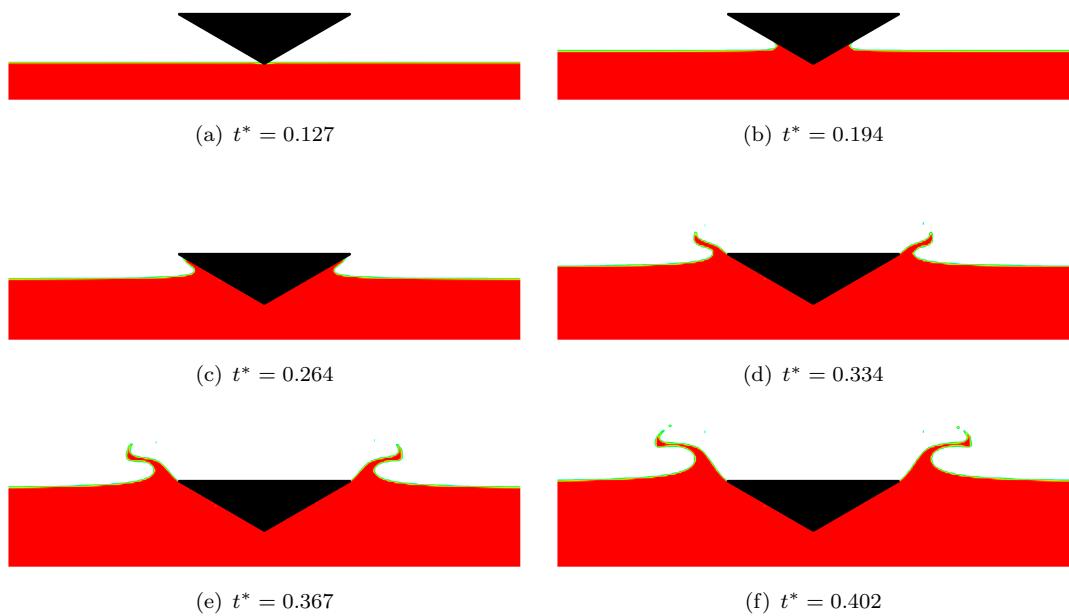


Figure 32. The evolution of water splash versus non-dimensional time $t^* = t/(d/V)$ for the case $\alpha = 120^\circ$

6. Acknowledgement

The Norwegian Research Council, through the researcher programs SIM-COFLW(proj. 234126) and SprayIce(proj. 244386), is acknowledged for supporting this work.

References

- Balaras, E. (2004). Modeling complex boundaries using an external force field on fixed cartesian grids in large-eddy simulations. *Computers & Fluids*, 33(3):375 – 404.
- Buchner, B. (2002). *Green Water on Ship-type Offshore Structures*. PhD dissertation, Delft University of Technology.
- Bussmann, M., Kothe, D. B., and Sicilian, J. M. (2002). Modeling high density ratio incompressible interfacial flows. In *ASME 2002 Joint US-European Fluids Engineering Division Conference*, pages 707–713. American Society of Mechanical Engineers.
- Cheny, Y. and Botella, O. (2010). The LS-STAG method: A new immersed boundary/level-set method for the computation of incompressible viscous flows in complex moving geometries with good conservation properties. *Journal of Computational Physics*, 229(4):1043–1076. bibtex: cheny2010.
- Clift, R., Grace, J. R., and Weber, M. E. (1978). *Bubbles, drops, and particles*. Academic Press.
- Cushman, J. H. (1982). Proofs of the volume averaging theorems for multiphase flow. *Advances in Water Resources*, 5(4):248–253.
- Desjardins, O. and Moureau, V. (2010). Methods for multiphase flows with high density ratio. *Center for Turbulent Research, Summer Programm*, 2010:313–322.
- Desjardins, O., Moureau, V., and Pitsch, H. (2008). An accurate conservative level set/ghost fluid method for simulating turbulent atomization. *Journal of Computational Physics*, 227(18):8395–8416.
- Dou, H.-S., Khoo, B. C., and Yeo, K. S. (2008). Instability of Taylor–Couette flow between concentric rotating cylinders. *International Journal of Thermal Sciences*, 47(11):1422–1435.
- Dressler, R. F. (1954). Comparison of theories and experiments for the hydraulic dam-break wave. In *Proc. Intl Assoc. of Scientific Hydrology Assemblée Générale*, volume 3, pages 319–328, Rome, Italy.
- Ehsan Khalili, M., Larsson, M., and Muller, B. (2018). Immersed boundary method for viscous compressible flows around moving bodies. *Computers & Fluids*, 170:77–92.
- Enright, D., Fedkiw, R., Ferziger, J., and Mitchell, I. (2002). A Hybrid Particle Level Set Method for Improved Interface Capturing. *Journal of Computational Physics*, 183(1):83–116.
- Fadlun, E. A., Verzicco, R., Orlandi, P., and Mohd-Yusof, J. (2000). Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *Journal of Computational Physics*, 161(1):35 – 60.
- Gibou, F., Fedkiw, R. P., Cheng, L.-T., and Kang, M. (2002). A Second-Order-Accurate Symmetric Discretization of the Poisson Equation on Irregular Domains. *Journal of Computational Physics*, 176(1):205–227.
- Gilmanov, A. and Sotiropoulos, F. (2005). A hybrid Cartesian/immersed boundary method for simulating flows with 3d, geometrically complex, moving bodies. *Journal of Computational Physics*, 207(2):457–492. bibtex: gilmanov2005.
- Gray, W. G. and Lee, P. C. Y. (1977). On the theorems for local volume averaging of multiphase systems. *International Journal of Multiphase Flow*, 3(4):333–340.
- Griebel, M. and Klitz, M. (2017). CLSVOF as a fast and mass-conserving extension of the level-set method for the simulation of two-phase flow problems. *Numerical Heat Transfer, Part B: Fundamentals*, 71(1):1–36.

- Grove, A. S., Shair, F. H., and Petersen, E. E. (1964). An experimental investigation of the steady separated flow past a circular cylinder. *Journal of Fluid Mechanics*, 19(01):60.
- Gueyffier, D., Li, J., Nadim, A., Scardovelli, R., and Zaleski, S. (1999). Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows. *Journal of Computational Physics*, 152(2):423–456.
- Guilmineau, E. and Queutey, P. (2002). A NUMERICAL SIMULATION OF VORTEX SHEDDING FROM AN OSCILLATING CIRCULAR CYLINDER. *Journal of Fluids and Structures*, 16(6):773–794.
- He, J.-W., Glowinski, R., Metcalfe, R., Nordlander, A., and Periaux, J. (2000). Active Control and Drag Optimization for Flow Past a Circular Cylinder. *Journal of Computational Physics*, 163(1):83–117.
- Hirt, C. W. and Nichols, B. D. (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201 – 225.
- Johnson, T. A. and Patel, V. C. (1999). Flow past a sphere up to a reynolds number of 300. *Journal of Fluid Mechanics*, 378:19–70.
- Kang, M., Fedkiw, R. P., and Liu, X.-D. (2000). A Boundary Condition Capturing Method for Multiphase Incompressible Flow. *Journal of Scientific Computing*, 15(3):323–360.
- Kim, J., Kim, D., and Choi, H. (2001). An immersed-boundary finite-volume method for simulations of flow in complex geometries. *Journal of Computational Physics*, 171(1):132 – 150.
- King, R., editor (2007). *Active Flow Control*, volume 95 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design (NNFM)*. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Kirkpatrick, M. P., Armfield, S. W., and Kent, J. H. (2003). A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid. *Journal of Computational Physics*, 184(1):1–36. bibtex: kirkpatrick2003.
- Lee, T.-h., Zhou, Z., and Cao, Y. (2001). Numerical simulations of hydraulic jumps in water sloshing and water impacting. *Journal of Fluids Engineering*, 124(1):215–226.
- Linnick, M. N. and Fasel, H. F. (2005). A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. *Journal of Computational Physics*, 204(1):157–192.
- Lobovský, L., Botia-Vera, E., Castellana, F., Mas-Soler, J., and Souto-Iglesias, A. (2014). Experimental investigation of dynamic pressure loads during dam break. *Journal of Fluids and Structures*, 48:407–434.
- Luo, K., Zhuang, Z., Fan, J., and Haugen, N. E. L. (2016). A ghost-cell immersed boundary method for simulations of heat transfer in compressible flows under different boundary conditions. *International Journal of Heat and Mass Transfer*, 92:708–717.
- Macklin, P. and Lowengrub, J. (2005). Evolving interfaces via gradients of geometry-dependent interior Poisson problems: application to tumor growth. *Journal of Computational Physics*, 203(1):191–220.
- Martin, J. C., Moyce, W. J., Martin, J. C., Moyce, W. J., Penney, W. G., Price, A. T., and Thornhill, C. K. (1952). Part IV. an experimental study of the collapse of liquid columns on a rigid horizontal plane. *Phil. Trans. R. Soc. Lond. A*, 244(882):312–324.
- Mittal, R., Dong, H., Bozkurttas, M., Najjar, F. M., Vargas, A., and von Loebbecke, A. (2008). A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *Journal of Computational Physics*, 227(10):4825–4852.
- Olsson, E. and Kreiss, G. (2005). A conservative level set method for two phase flow. *Journal of Computational Physics*, 210(1):225–246.
- Puckett, E. G., Almgren, A. S., Bell, J. B., Marcus, D. L., and Rider, W. J. (1997). A High-Order Projection Method for Tracking Fluid Interfaces in Variable Density Incompressible Flows. *Journal of Computational Physics*, 130(2):269–282.
- Quintard, M. and Whitaker, S. (1995). AEROSOL FILTRATION: AN ANALYSIS USING THE METHOD OF VOLUME AVERAGING. *J. Aerosol Sci.*, 26(8):1227–1255.
- Raessi, M. (2008). A level set based method for calculating flux densities in two-phase flows. *Annual Research Briefs*, pages 467–478.

- Rudman, M. (1998). A volume-tracking method for incompressible multifluid flows with large density variations. *International Journal for numerical methods in fluids*, 28(2):357–378.
- Schneiders, L., Günther, C., Meinke, M., and Schröder, W. (2016). An efficient conservative cut-cell method for rigid bodies interacting with viscous compressible flows. *Journal of Computational Physics*, 311:62–86.
- Seo, J. H. and Mittal, R. (2011). A sharp-interface immersed boundary method with improved mass conservation and reduced spurious pressure oscillations. *Journal of Computational Physics*, 230(19):7347 – 7363.
- Strang, G. (1968). On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis*, 5(3):506–517.
- Sussman, M., Smereka, P., and Osher, S. (1994). A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114(1):146 – 159.
- Taira, K. and Colonius, T. (2007). The immersed boundary method: A projection approach. *Journal of Computational Physics*, 225(2):2118–2137.
- Taneda, S. (1956). Experimental investigation of the wake behind a sphere at low reynolds numbers. *Journal of the Physical Society of Japan*, 11(10):1104–1108.
- Tryggvason, G., Scardovelli, R., and Zaleski, S. (2011). *Direct Numerical Simulations of Gas-Liquid Multiphase Flows*. Cambridge University Press, Cambridge.
- Vincent, L., Xiao, T., Yohann, D., Jung, S., and Kanso, E. (2018). Dynamics of water entry. *Journal of Fluid Mechanics*, 846:508–535.
- Vukčević, V., Jasak, H., and Gatin, I. (2017). Implementation of the Ghost Fluid Method for free surface flows in polyhedral Finite Volume framework. *Computers & Fluids*, 153:1–19.
- Wang, S., Glimm, J., Samulyak, R., Jiao, X., and Diao, C. (2013). An Embedded Boundary Method for Two Phase Incompressible Flow. *arXiv preprint arXiv:1304.5514*.
- Wang, Z., Yang, J., Koo, B., and Stern, F. (2009). A coupled level set and volume-of-fluid method for sharp interface simulation of plunging breaking waves. *International Journal of Multiphase Flow*, 35(3):227–246.
- Ye, T., Shyy, W., and Chung, J. N. (2001). A Fixed-Grid, Sharp-Interface Method for Bubble Dynamics and Phase Change. *Journal of Computational Physics*, 174(2):781–815.