A Hybrid Particle-Mesh Method for Viscous, Incompressible, Multiphase Flows

Jie LIU, The University of Tokyo, 2-22 Shirane, Shirakata, Tokai, Naka, Ibaraki 319-1188, Japan, liuj@utnl.jp
Seiichi KOSHIZUKA, The University of Tokyo, 2-22 Shirane, Shirakata, Tokai, Naka, Ibaraki 319-1188, Japan
Yoshiaki OKA, The University of Tokyo, 2-22 Shirane, Shirakata, Tokai, Naka, Ibaraki 319-1188, Japan

A hybrid method to simulate unsteady multiphase flows is described. One phase is represented by moving particles and the other phase is defined on stationary mesh. The flow field is discretized by a conservative finite volume approximation on the stationary mesh, and the interface is automatically captured by the distribution of particles moving through the stationary mesh. The effects of surface tension and wall adhesion are evaluated by the Continuum Surface Force (CSF) model. The method simplifies the calculation of interface interaction, enables accurate modeling of two- and three-dimensional multiphase flows.

1. Introduction

Accurate simulation of multiphase flows with a sharp interface has considerable difficulty in numerical methods. Problems with a moving interface are important in many technological applications in which moving interfaces play a dominant role. Capillarity phenomena, wetting effect, deformation of droplet or bubble, motion of boundaries between immiscible fluids are some of well-known examples in engineering and science. In order to describe quantitatively such problems, the understanding of physical processes taking place on the interfaces is necessary. In particular, successful simulation of the moving interfaces depends on the numerical method in which the corresponding interfaces can be captured effectively.

The most popular approach to compute multiphase flows is to capture the front directly on a regular, stationary mesh. For example, Marker-And-Cell (MAC) ⁽¹⁾ method uses marker particles to identify each fluid. The Volume-Of-Fluid (VOF) ⁽²⁾ method uses a marker function to identify the interface. The main difficulty in using these methods is the maintenance of a sharp boundary between two phases and the computation of surface tension. Further developments to the methods that capture fluid interfaces on a fixed grid include the CIP ⁽³⁾ method and the phase field method ⁽⁴⁾, etc.

Adaptive (moving) grid methods alter the computational grid so that the interface always coincides with one of the grid lines. The interface is then a well-defined, continuous curve. The information regarding the location and curvature of the interface is readily available. The review of techniques for the construction of boundary conforming coordinates is referred to ⁽⁵⁾. The main advantage of this approach is that it is possible to maintain sharp resolution of the interface, while the main disadvantage is the difficulty in adjusting the grids to follow the highly deformed interfaces.

The front tracking is another method where the interface itself is described by additional computational elements, and the fixed grid is kept unchanged. The major difficulty of direct front tracking is the question of how the interface grid interacts with the stationary grid, and vice versa. It is also necessary to restructure the interface grid dynamically as the calculations proceed. Computational points must be added in regions where the interface grid points become scattered, and eliminated from regions of concentration. Another problem in front tracking results from the interaction of a front with another front. Double interfaces have to be merged into one interface or eliminated. A merging algorithm is usually used. A development to this method made by G. Tryggvason et al. ⁽⁶⁾ is a hybrid between front capturing and front tracking technique. A stationary regular grid is used for the fluid flow, but the interface is tracked by a separated grid of lower dimension. However, unlike front tracking methods where each phase is treated separately, all phases are treated together by solving a single set of governing equations for the whole field. The explicit tracking of the interface reduces errors associated with the surface tension computations, and its flexibility makes it applicable to problems where complex interface physics must be accounted for.

From the point of view of numerical algorithms, the methods mentioned above are classified as Eulerian method. Lagrangian

method is another category, which is suited for moving boundary problems because it permits interfaces to be specifically delineated and precisely followed and it allows interface boundary conditions to be easily applied ⁽⁵⁾. The two main problems of the Lagrangian methods are mesh tangling and numerical inaccuracy due to highly irregular meshes. Particle method is another Lagrangian description of flows in which particles are explicitly associated with different materials and thus the interfaces can be easily followed. The well-known example is the Particle-In-Cell (PIC) ⁽⁷⁾ algorithm. The basic idea of PIC is to divide the region of interest into regular Eulerian cells for purposes of computing field variables such as pressure and fluid velocity, and to simulate the material transport from cell to cell in a Lagrangian fashion in the form of discrete simulation particles. The attractive features of the method are no mass diffusion and the ability of treating large distortions of fluids, large slippages, and colliding interfaces. The problems consist of a large numerical diffusion caused by transferring the momentum from the grid to the particle and back, numerical noise caused by the use a finite numerical of fluid particles, and limited spatial resolution because of a fixed uniform Eulerian grid being used.

The Lagrangian particle method called Smooth Particle Hydrodynamics (SPH) ⁽⁸⁾ abandons grid completely. The approximation of spatial derivatives is derived from the interpolation among particles. Moving Particle Semi-implicit (MPS) ⁽⁹⁻¹⁷⁾ method is another meshless method where the fluids are represented by a finite number of moving particles. Governing equations are discretized based on particle interaction models representing gradient, Laplacian, and free surface. Grids are not necessary at all. Thus, the method is free from the grid distortion. A semi-implicit algorithm is employed to analyze incompressible flows. The numerical diffusion derived from the convection terms does not arise because of fully Lagrangian description. The MPS method was applied to multi-phase and multi-fluid thermal-hydraulic problems such as free surface flows ⁽¹⁰⁾, droplet breakup ⁽¹¹⁾, single bubble rise and nucleate boiling ^(12, 13), jet injection ⁽¹⁴⁾, fragmentation ^(15, 16), large deformation and fracture of solid ⁽¹⁷⁾, etc.

For the accurate prediction of multiphase flows, the numerical method must possess the ability of evaluation surface tension forces at interface. In the Continuum Surface Force (CSF) model ⁽¹⁸⁾, a surface force is modeled with a localized volume force. Instead of a surface tensile force or a surface pressure boundary condition applied at a discontinuity, a volume force acts on fluid interfaces within finite thickness continuously. Surface tension modeled with the continuum method eliminates the need for interface reconstruction, and can be easily calculated by applying an extra body force in the momentum equation.

In this article we present a hybrid method for incompressible, viscous, multiphase flows by coupling MPS method with mesh method. One fluid (or phase) is represented by moving particles, while another fluid (or phase) is defined on the mesh. The interfaces are automatically determined by the distribution of particles without specific front tracking algorithms, and no numerical diffusion arises. The original underlying mesh is retained through the simulations and no restructuring is needed. The Continuum Surface Force (CSF) model will be used to evaluate the

surface tension force, and all phases are treated together by solving a single set of governing equations for the whole flow field. The primary advantage of this approach is that the interfaces carry the jump of properties and that, at each time step, the property field is easily calculated from the information carried by the distribution of particles. The interactions of the interfaces automatically taken care of in the present method is considered one of the major differences from the previous front-tracking methods.

2. Numerical Method

(1) Description of Multiphase Flow by Particle and Mesh

In the present method, the multiphase flow region of interest is divided into regular Eulerian mesh for purposes of computing field variables such as pressure and fluid velocity. Some part of the region is occupied by a finite ensemble of particles, which represents one phase (liquid phase, for example, in gas-liquid two-phase flow). The rest part of the region not occupied by the particles is another phase (gas phase, for example). The general setup of particle and mesh is sketched in Fig. 1.

The use of particles is the same way as that in MPS method. Each particle has a set of attributes, such as mass and position. Since the particles are explicitly associated with one phase, the interfaces between different phases can be easily followed by the distribution of particles. The whole multiphase flow system will be calculated on the mesh, and the advection of the density and the viscosity fields is achieved by following the motion of particles. The velocities calculated on the mesh are used to advance the particles. The attractive features of the method are that the mass conservation is preserved, and the interaction of interfaces is possible. Furthermore, the continuous change of physical quantities to eliminate the numerical noise is ensured. The transfer of information between the particles and the grid will be given in detail in the following sessions.

Fig. 1 The setup of mesh and particle.

(2) Governing Equations

The equations governing the motion of unsteady, viscous, incompressible multiphase flows are the laws of conservation of mass

$$\oint_S \bar{v} \cdot \bar{n} dS = 0, \quad (1)$$

and momentum

$$\frac{\partial}{\partial t} \int_V \rho \bar{v} dV + \oint_S \rho \bar{v} (\bar{v} \cdot \bar{n}) dS = - \oint_S p \bar{I} \cdot \bar{n} dS - \oint_S \bar{\tau} \cdot \bar{n} dS, \quad (2)$$

$$+ \int_V \bar{f}_v dV$$

where t represents time, $\bar{v}$, p , and ρ are velocity vector, the pressure and the density of the fluid, respectively. $\bar{I}$ is the identity matrix, V is the volume occupied by the fluid, S is the surface area formed by the boundary of this volume, and $\bar{n}$ is the outward pointing unit vector, normal to S . $\bar{f}_v$ is the sum of forces acting

on the fluid. $\bar{\tau}$ is the stress tensor, which for a Newtonian fluid in two dimensions is

$$\bar{\tau} = \begin{bmatrix} -2\mu \frac{\partial u}{\partial x} & -\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \\ -\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & -2\mu \frac{\partial v}{\partial y} \end{bmatrix}, \quad (3)$$

where μ is the dynamic viscosity, and u and v are the horizontal and vertical components of the velocity, respectively.

Although $\bar{f}_v$ has the form of a volume force, it also includes the effect of the surface tension in the present method, which will be discussed in the following session.

(3) Surface Tension Model

In the present method, the interfaces are represented by the interfacial particles. The interfacial particles can be easily determined by the particle number density, which was defined originally in the MPS method⁽⁹⁾. The particle number density n of particle i is defined as

$$n_i = \sum_{j \neq i} w(|\bar{r}_j - \bar{r}_i|) \quad (4)$$

where $\bar{r}$ is the coordinate, and w is weight function defined by

$$w(r) = \begin{cases} \frac{r_e - r}{r_e} & r \leq r_e \\ 0 & r > r_e \end{cases}, \quad (5)$$

here r is the distance between two particles, r_e is radius of interaction area. The particle number density is proportional to the fluid density. The mass conservation law of incompressible flow requires that the fluid density should be constant. This is equivalent to the particle number density being constant, n^0 . A particle whose particle number density satisfies

$$n_i < \beta n^0 \quad (6)$$

is regarded as the interface, where β is a parameter below 1.0.

The surface tension force is calculated on the particles that are regarded as interface. In the present method, surface tension is modeled as a volume force derived from the Continuum Surface Force (CSF) model⁽¹⁸⁾. The surface force per unit interfacial area can be written as

$$\bar{f}_{sa} = \sigma \kappa \hat{n}, \quad (7)$$

where σ is the surface tension coefficient, κ is the curvature of the interface and $\hat{n}$ is unit normal to the interface. The curvature is calculated form

$$\kappa = -(\nabla \cdot \hat{n}) \quad (8)$$

and the unit normal is derived from normal vector

$$\hat{n} = \frac{\bar{n}}{|\bar{n}|}. \quad (9)$$

The normal vector can be obtained by a gradient of particle number density at the interface. As shown in Fig. 2, a gradient vector between two particles i and j possessing n_i and n_j at coordinates $\bar{r}_i$ and $\bar{r}_j$ is simply defined by

$(n_j - n_i)(\bar{r}_j - \bar{r}_i)/|\bar{r}_j - \bar{r}_i|^2$. The gradient vectors between particle i and its neighboring particles j are weighted with the kernel function and averaged to obtain a gradient vector at particle i :

$$\bar{n}_i = (\nabla n)_i = \frac{d}{n_i} \sum_{j \neq i} \left[\frac{n_j - n_i}{|\bar{r}_j - \bar{r}_i|^2} (\bar{r}_j - \bar{r}_i) w(|\bar{r}_j - \bar{r}_i|) \right], \quad (10)$$

where d is the number of space dimensions⁽⁹⁾.

The divergence of unit normal vector is calculated by

$$(\nabla \cdot \hat{n})_i = \frac{d}{n_i} \sum_{j \neq i} \left[\frac{(\hat{n}_j - \hat{n}_i)(\bar{r}_j - \bar{r}_i)}{|\bar{r}_j - \bar{r}_i|^2} w(|\bar{r}_j - \bar{r}_i|) \right], \quad (11)$$

where $\hat{n}$ is unit normal vector, n_i is particle number density.

Fig. 2 Definition of vector normal and contact angle.

The surface force $\vec{f}_{sa}$ per unit interfacial area acting on particle i can be transferred to volume force $\vec{f}_{sv}$. If the average distance between particles is d , the volume force is estimated by $\vec{f}_{sv} = \vec{f}_{sa} / d$ in two dimensions because the area of a particle is d^2 .

With the volume force, surface tension effects at interface are modeled as a body force included in the momentum equation (2), instead of a surface tensile force or a surface pressure boundary condition applied at a discontinuity.

(4) Boundary Conditions: Wall Adhesion

Wall adhesion is the surface force acting on fluid interfaces at points of contact with “walls”, which are static, rigid boundaries. Wall adhesion forces are calculated in the same manner as volume forces due to surface tension are calculated, using Eq. (7) for $\vec{f}_{sa}$, except that a boundary condition is applied to the interface unit normal $\hat{n}$ prior to evaluating Eq. (7). The condition is applied only to those particles near a rigid boundary, as shown in Fig. 2. The wall adhesion boundary condition becomes an expression for the unit interface normal $\hat{n}$ at points of contact $\vec{r}_w$ along the wall (18).

$$\hat{n} = \hat{n}_w \cos \theta_{eq} + \hat{n}_t \sin \theta_{eq}, \quad (12)$$

where θ_{eq} is the static contact angle between the fluid and the wall, $\hat{n}_w$ is the unit wall normal directed into the wall, and $\hat{n}_t$ is tangent to the wall, normal to the contact line between the interface and the wall at $\vec{r}_w$. The unit tangent $\hat{n}_t$ is directed into the fluid.

The angle θ_{eq} is not a fluid material property, but a system property, depending also on properties of the wall itself. The value of θ_{eq} is measured experimentally when the fluid is at rest. So in the present method the treatment of wall adhesion is static because θ_{eq} is assumed to be a constant.

(5) Mesh Calculation by Finite Volume Method

Computational solutions of Eqs. (1) and (2) are obtained on a staged mesh (Fig. 3). The pressure, density, and viscosity are defined at the center of each cell; the horizontal component of the velocity is placed at the center of the vertical cell faces; and the vertical component of the velocity is located at the center of the horizontal cell faces.

In the present method, the interface is automatically determined by the distribution of particles. The mesh is stationary and the particle is moving, so information transfer is required from each other. The transfer is as follows: (1) a resultant surface force is

computed on the interfacial particles and must be distributed to the background mesh; (2) the density and viscosity of the particles are distributed to the background mesh; (3) the velocity of the fluid is computed on the background mesh and must be interpolated to the particles.

Fig. 3 Staggered mesh.

The resultant surface force is distributed to the Cartesian mesh using an area-weighted extrapolation, and incorporated into Eq. (2) through the force term, $\vec{f}_v$. The procedure is as follows: (1) the cell that encloses the center of the interfacial particle is found; (2) the neighbors of the cell are found; (3) the fractional areas that the particle occupied on the neighbor cells are computed; (4) these fractional areas are used to distribute the surface force $\vec{f}_{sv}$ to the four cells. This procedure is sketched in Fig. 4. The expression for the force assigned to each cell is

$$(\vec{f}_v)_k = \frac{1}{V} \vec{f}_{sv} A_k \quad (13)$$

where k indicates each cell around the center of the particle, V is the volume of cell, A_k is the fractional area associated with the k th cell ($\sum_{k=1}^4 A_k = 1$). The distribution is performed for all interfacial particles, and the contributions of different particles to the same cell are added.

Fig. 4 Extrapolation of surface force from particle to mesh.

A similar expression is used for the density and viscosity of particles. The cell neighbors are included in the averaging as a form of smoothing to avoid sharp jumps in the density and viscosity, which can produce spurious spikes in the solution.

The description of fluid by particles prohibits intersected interfaces, which is a serious problem in front tracking method. If the size of particle is the same as the computational cell, at least one particle per cell is guaranteed. Since the particles move independently of one another, special measures must be taken to prevent particles from clustering. This will be discussed in the

following session.

The finite-volume discretization of Eqs. (1) and (2) is used,

$$\sum_{\text{faces}} \bar{v} \cdot \bar{n} \Delta S = 0 \quad (14)$$

$$\frac{(\rho \bar{v} V)^{n+1} - (\rho \bar{v} V)^n}{\Delta t} = - \sum_{\text{faces}} (\bar{F}_{\text{conv}} \cdot \bar{n} \Delta S)^n - \sum_{\text{faces}} (\bar{F}_{\text{press}} \cdot \bar{n} \Delta S)^{n+1} + \sum_{\text{faces}} (\bar{F}_{\text{visc}} \cdot \bar{n} \Delta S)^n + \bar{f}_v V \quad (15)$$

where V is the size of the control volume; ΔS is the size of a face; $\bar{n}$ is the outward normal to the face; and the total flux has been divided into three components: convective fluxes ($\bar{F}_{\text{conv}}$), pressure fluxes ($\bar{F}_{\text{press}}$), and viscous fluxes ($\bar{F}_{\text{visc}}$), which are defined in two dimensions as

$$\begin{aligned} \bar{F}_{\text{conv}} &= \begin{bmatrix} \rho u^2 & \rho uv \\ \rho uv & \rho v^2 \end{bmatrix}, \\ \bar{F}_{\text{press}} &= \begin{bmatrix} p & 0 \\ 0 & p \end{bmatrix}, \\ \bar{F}_{\text{visc}} &= \begin{bmatrix} 2\mu \frac{\partial u}{\partial x} & \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \\ \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & 2\mu \frac{\partial v}{\partial y} \end{bmatrix}, \end{aligned} \quad (16)$$

where u and v are the x - and y -components of the velocity vector, respectively.

In the staggered mesh, the control volume used for conservation of mass is centered at the cell centroid, the one for the conservation of momentum in the x -direction is centered at the right face, and the one for the conservation of momentum in the y -direction is centered at the upper face (Fig. 3).

The equations are solved by means of a projection method. The momentum equation is split into two parts:

$$\frac{\rho^{n+1} \bar{v}^* V - (\rho \bar{v} V)^n}{\Delta t} = - \sum_{\text{faces}} (\bar{F}_{\text{conv}} \cdot \bar{n} \Delta S)^n + \sum_{\text{faces}} (\bar{F}_{\text{visc}} \cdot \bar{n} \Delta S)^n + \bar{f}_v V \quad (17)$$

and

$$\frac{(\rho \bar{v} V)^{n+1} - \rho^{n+1} \bar{v}^* V}{\Delta t} = - \sum_{\text{faces}} (\bar{F}_{\text{press}} \cdot \bar{n} \Delta S)^{n+1}, \quad (18)$$

where $\bar{v}^*$ is temporal velocity. By using the mass conversion equation at new time,

$$\sum_{\text{faces}} \bar{v}^{n+1} \cdot \bar{n} \Delta S = 0, \quad (19)$$

the pressure equation is found by eliminate $\bar{v}^{n+1}$ from Eq. (18) as

$$\sum_{\text{faces}} \left[\frac{1}{\rho^{n+1}} \sum_{\text{faces}} (\bar{F}_{\text{press}} \cdot \bar{n} \Delta S)^{n+1} \right] \cdot \bar{n} \Delta S = \frac{V}{\Delta t} \sum_{\text{faces}} \bar{v}^* \cdot \bar{n} \Delta S. \quad (20)$$

The updated velocity is then found from equation (18). Since the density is variable, the pressure equation cannot be solved by traditional fast Poisson solvers designed for separable elliptic equations. Here a simple Successive Over Relaxation (SOR) method is used. The over-relaxation parameter by which the pressure equation is solved depends generally on the density ratio of different fluids.

(6) Particle Calculation

Since the fluid velocities are computed on the fixed mesh and particles move with the fluid velocities, the velocity of particles must be found by interpolating from the fixed mesh. The interpolation starts by identifying the grid points that are closest to the particle. The particle velocity is then interpolated by area-weighted interpolation which incorporates most of the features

developed for the distribution of the surface force described above. Once the velocity of each particle has been found, its new position can be found by integration

$$\bar{r}_i^* = \bar{r}_i^n + \bar{v}_i^n \Delta t, \quad (21)$$

where $\bar{r}$ is the particle position, $\bar{v}$ is the particle velocity, and Δt is the time step.

The continuity equation requires that the fluid density should be constant. This is equivalent to the particle number density being constant, n^0 . When the particle number density n^* is not n^0 , it is corrected to n^0 by

$$n^* + n' = n^0 \quad (22)$$

where n' is the correction value. This is related to the velocity correction value $\bar{v}'$ through the mass conservation equation⁽⁹⁾:

$$\frac{1}{\Delta t} \frac{n'}{n^0} = -\nabla \cdot \bar{v}' \quad (23)$$

The velocity correction value is derived from a correction pressure gradient term as

$$\bar{v}' = -\frac{\Delta t}{\rho} \nabla p' \quad (24)$$

where p' is the correction pressure, ρ is the density of fluid represented by particles. With Eqs. (22), (23) and (24), a Poisson equation of correction pressure is obtained:

$$\langle \nabla^2 p' \rangle_i = -\frac{\rho}{\Delta t^2} \frac{\langle n^* \rangle_i - n^0}{n^0}. \quad (25)$$

The Poisson equation is solved by the incomplete Cholesky conjugate gradient (ICCG) method⁽⁹⁾ with a Dirichlet boundary condition ($p' = 0$) given to the particles at interfaces. The velocity correction is computed by Eq. (24), and the position of particle is modified by the velocity correction value:

$$\bar{r}_i^{n+1} = \bar{r}_i^* + \bar{v}_i' \Delta t. \quad (26)$$

In each time step a well-distributed particles is guaranteed by keeping the particle number density. After the particle's position is adjusted, the particle's velocity is omitted. Only the velocities defined on mesh remain.

(7) Solution Algorithm

The overall algorithm is described in Fig. 5. First, the particle's density and viscosity, and the surface force calculated on the interfacial particles are extrapolated to mesh, and then the velocity and pressure fields are calculated on mesh by finite volume method. Next, the particle's velocity is interpolated from the mesh, and then the particle moves with this velocity. Finally, the position of particle is modified by the particle number density.

3. Computational Examples

Here present the results for several standard static and dynamic problems with surface tension to illustrate the flexibility and accuracy of the hybrid method.

3.1. Equilibrium Rod

A static liquid drop will become spherical under surface tension in the absence of gravitational or other external forces. For an infinite cylinder surrounded by a background fluid at zero pressure, the internal drop pressure, p_{drop} , will be

$$p_{\text{drop}} = \sigma \kappa = \sigma / R, \quad (27)$$

where R is the drop radius. Here the calculation of equilibrium rod with surface tension is carried out on a two-dimensional computational domain. A regular orthogonal 66×66 mesh ($\Delta x = \Delta y = 0.001\text{m}$) partitions the domain, with background density $\rho = 500(\text{kg/m}^3)$. A fluid drop with radius $R = 0.02$ (m), density $\rho = 1000(\text{kg/m}^3)$, and surface tension coefficient $\sigma = 0.02361(\text{N/m})$, is centered at the point (0.03, 0.03). The fluid drop is represented by 2020 particles with distance $D = 0.001(\text{m})$ between them. The configuration of mesh and particle is shown in Fig. 6. Since the initial distribution of particles does not form exactly a rod, the particle's position will be adjusted under surface tension to form a

more circular one. Fig. 7 shows the position of particles at about 0.5s. From Eq. (27), the pressure inside the fluid drop is 1.1805 (Pa) under the surface tension coefficient given above. This value is compared with the mean drop pressure computed by an average process defined as

$$\langle p \rangle = \frac{1}{N_p} \sum_{i=1}^{N_p} p_i, \quad (28)$$

where the sum is over the N_p particles lying within the drop that have particle number density $n_i > \beta n^0$. Fig. 8 shows the computed mean drop pressure by Eq. (28) and its variation with time. Since the particles on the interface oscillate around the equilibrium position, the mean drop pressure waves around the analytical value. The result suggests that smoothing algorithm for the surface tension calculation is necessary to obtain an accurate, uniform drop pressure. Without smoothing process, relative errors in the curvature of interface are nearly 80%. However, the convergence of surface tension model is demonstrated by the result. The result also shows that the definition of interfacial by Eq. (6) is not sensitive to the parameter β .

Fig. 5 Algorithm of the hybrid method.

Fig. 6 Initial configuration of mesh and particles.

Fig. 7 Mesh and particles at time 0.5s.

Fig. 8 Mean pressure of the liquid rod.

3.2. Non-equilibrium Rod

An initially square drop is numerically calculated to observe the behavior responding to unbalanced surface tension forces. A square ethanol drop with $\rho = 797.88(\text{kg/m}^3)$ and $\sigma = 0.02361(\text{N/m})$ is centered on a domain with 96×96 mesh ($\Delta x = \Delta y = 0.0025\text{m}$). The drop consists of 30×30 particles ($D = 0.0025\text{m}$). A small density ($\rho = 1.0\text{kg/m}^3$) and a small viscosity ($\mu = 10^{-5}\text{Pa}\cdot\text{s}$) are given to the background fluid defined on the mesh. The gravitational and other external forces are neglected. The oscillations of the surface of the drop about its equilibrium shape are shown in Fig. 9 at a time sequence, $t = 0, 0.575, 1.250$, and 1.825s . The initially square shape of the drop results in very strong surface forces at the high-curvature corners, setting the drop into oscillation. The period of oscillation is approximately 1.25s . In the numerical calculation, the oscillations damp because of viscosity of fluids. At about $t = 4.0\text{s}$, the drop is nearly circular in cross section. Animations of computational results indicated that the distribution of particles at interface cause discontinuous changes of interface location. These provide a small impulse through the surface tension to sustain the oscillations.

3.3. Equilibrium Contact Angle

By definition, wetting is the ability of liquids to form a boundary surface with solids. It plays an important role in many technical processes such as adhesives, coating, oil recovery, gluing cleaning, etc. The phenomenon of wetting or non-wetting of a solid wall by a liquid is illustrated by studying what is known as contact angle. The contact angle θ describes the shape of a liquid drop adsorbed at the solid wall. If $\theta = 0$, the solid wall preferentially adsorbs a layer of liquid, which is described as wetting. While if $0 < \theta < 90^\circ$, the solid wall is wetted with only cap-shaped liquid drop adsorbed. If $\theta > 90^\circ$ it is a layer of the gas phase which is adsorbed at the solid wall, sometimes described as non-wetting.

A liquid drop lying on a solid wall is numerically simulated to observe the equilibrium shape of drop under different contact angles. A square ethanol drop with $\rho = 797.88(\text{kg/m}^3)$ and $\sigma = 0.02361(\text{N/m})$ is initially put on the solid wall. The drop consists of 30×30 particles ($D = 0.002\text{m}$). A small density ($\rho = 1.0\text{kg/m}^3$) is given to the background fluid defined on the 146×46 mesh ($\Delta x = \Delta y = 0.002\text{m}$). The gravitational and other external forces are neglected. A large viscosity ($\mu = 10^{-2}\text{Pa}\cdot\text{s}$) is given for the background fluid. The shapes of the drop at equilibrium state are shown in Fig. 10 under different static contact angles, $\theta_{eq} = 30, 60, 90, 120$, and 150° . The initially square shape of the drop deforms under the strong surface forces at the high-curvature corners and the adhesive forces of solid wall. The oscillations of drop shape damp because of large viscosity, and finally an equilibrium state of drop shape is obtained.

Fig. 9 Vibration of square ethanol drop under surface tension.

Fig. 10 Effects of static contact angle on drop shapes.

3.4. Flow Induced by Wall Adhesion

As an example of the effects of wall adhesion, consider a shallow pool of water located at the bottom of a two-dimensional tank. Assume that the water interface wants to attain a specified contact angle θ_{eq} with the tank wall, different from the initial angle of 90° (a horizontal interface). Two different cases are computed in Fig. 11-12, one in which the water wets the wall ($\theta_{eq} < 90^\circ$), and one in which the water does not wet the wall ($\theta_{eq} > 90^\circ$). The results are computed on a 56×76 mesh ($\Delta x = \Delta y = 0.002\text{m}$) with particles representing the water. For the wetting case, the pool of water is 0.05m deep and $\theta_{eq} = 5^\circ$. For the non-wetting case, the pool of water is 0.02m deep and $\theta_{eq} = 175^\circ$. All external forces such as gravity are set to zero.

Figure 11 displays the motion of flow induced by wall adhesion in the wetting case. The wall adhesion boundary condition of Eq. (12) is applied to the normal vectors $\hat{n}$ along the tank wall, resulting in the upward surface forces at contact location. From the sequence of times in Fig. 11, $t = 0, 0.5, 1.0, 1.5$, and 2.0s , it is evident that the flow field due to the wall adhesion forces for $\theta_{eq} = 5^\circ$ causes the water to move up the tank walls until the boundary condition in Eq. (12) is satisfied.

There exists a problem in the calculation. When the water film on the tank wall becomes very thin (for example, only 2 or 3 particles' thick), here the particle has very few neighbors and the calculation based on particle interactions is not accurate enough. For example, the calculation of gradient by Eq. (10) or divergence by Eq. (11) is an average process over the neighbors of one particle. This problem causes the interface unstable after $t = 2.0\text{s}$ in Fig. 11.

In the non-wetting case, a "ball" of water evolves from an initial shallow pool when the wall adhesion forces are specified on every boundary with $\theta_{eq} = 175^\circ$. The water behaves in this case like mercury, wanting to separate itself from the walls since contact angle is obtuse. The shape of water is displayed at times of $t = 0, 0.5, 1.0, 2.0$, and 3.0s in Fig. 12. A net upward momentum, evident in Fig. 12e, is imported to the water ball by the wall adhesion force, eventually causing it to separate from the bottom wall.

Fig. 11 Flow induced by wall adhesion for the wetting case.

Fig. 12 Flow induced by wall adhesion for non-wetting case.

3.5. Rayleigh-Taylor Instability

The Rayleigh-Taylor instability is created when a heavy fluid is put upon a lighter one and when the equilibrium state is perturbed. As a basic two-phase flow phenomenon, the Rayleigh-Taylor instability is numerically simulated by using the present method. In the initial state of the phenomenon, the heavier fluid $\rho_1 = 3$ lies above the lighter one $\rho_2 = 1$ in a channel with the width 1 and height 2. The gravity acceleration is given as $g = 10$. Both fluids have the same kinematical viscosity $\nu_1 = \nu_2 = 0.01$. Two disturbance cases, one is upward and another is downward, are given by the sinusoidal interfaces with the amplitude 0.06 as shown in Fig. 13. The heavy fluid is represented by particles with distance $D = 0.02$, and the lighter one is defined on mesh ($\Delta x = \Delta y = 0.02$). All quantities mentioned above are non-dimensional.

The various behaviors of the two-phase system are observed at equivalent times and the effects of surface tension are illustrated. Fig. 14 shows the evolutions ($t = 0.4, 0.8$, and 1.2) of instabilities

of the two cases without surface tension ($\sigma = 0$). In upward disturbance case, the lighter fluid moves upward and a bubble is formed. While in downward disturbance case, a jet of heavier fluid falls down. Further developments of the phenomenon mix the fluids to the dispersed state, because there is no surface tension which would limit the dispersion.

Fig. 13 Initial states of Rayleigh-Taylor instability.

Fig. 14 Evolutions of instabilities ($t = 0.4, 0.8$, and 1.2) without surface tension.

Fig. 15 shows the developments of instability of the two cases with a surface tension ($\sigma = 0.127$) at times of $t = 0.5, 1.0$, and 1.5 . The effect of surface tension is effectively observed that the surface force tends to make the interface as flat as possible. The curvatures of the interfaces are less than those without surface tension.

Moreover, when the surface tension is much higher ($\sigma = 0.253$), the instability behaves in a different way as shown in Fig. 16 at times of $t = 1.0, 2.0$, and 3.0 . Although the disturbance initiates the

instability, the surface tension prohibits it from developing. Due to the numerical error, the symmetric state of the system is lost, which results in another disturbance on the balance of interface. The perturbation develops and finally the heavy fluid flows downward near one sidewall of tank.

Fig. 15 Evolutions of instabilities ($t = 0.5, 1.0$, and 1.5) with surface tension ($\sigma = 0.127$).

3.6. Kelvin-Helmholtz Instability

The Kelvin-Helmholtz instability is a fundamental instability of incompressible fluid flow, which will arise, for example, from two parallel fluids of different densities moving at different velocities. The likelihood of Kelvin-Helmholtz instability can be evaluated by means of the Richardson's number (Ri)⁽¹⁹⁾:

$$\text{Richardson Number} = \frac{\text{Stabilizing Buoyancy}}{\text{Destabilizing Shear}} \quad (29)$$

The instability occurs when the destabilizing shear is strong enough to break up the stable layer across the interface between two fluids.

Here a simple way is considered to generate the Kelvin-Helmholtz instability. In a long narrow tank a lighter freshwater rests on a heavier saltwater with a stable boundary between the two fluids. If the tank is tilt at an angle, the saltwater flows down and the freshwater flows upward, causing a current speed difference at the boundary of the two types of water. When the current speed difference exceeds a certain level, the current becomes unstable. The nonlinear growth and evolution of this interface are governed by only the competing effects of the buoyancy and shear.

The calculation is carried out on a two-dimensional geometry shown in Fig. 17a with the present method. To tilt the tank is equivalent to give the gravity in the direction different from downward vertically (with an angle $\theta = 30^\circ$). The densities of freshwater and saltwater are 1000kg/m^3 and 1050kg/m^3 , respectively. The gravity acceleration is given as $g = 9.8\text{m/s}^2$. Surface tension is neglected. The saltwater is represented by particles with distance $D = 0.002\text{m}$, and the freshwater is defined

on mesh ($\Delta x = \Delta y = 0.002\text{m}$), shown in Fig. 17b.

Fig. 18 shows the evolution of instability at times of $t = 1.2, 1.4, 1.6, 1.8$, and 2.0 . The interface rolls up and forms a wave which grows quickly. In the present calculations, the freshwater and saltwater flow to either side without mixing. However, in the experiments⁽¹⁹⁾ the two kinds of fluid mix violently and the mixing finally ends up with a turbulent state.

Fig. 16 Evolutions of instabilities ($t = 1.0, 2.0$, and 3.0) with surface tension ($\sigma = 0.253$).

Fig. 17 Calculation conditions.

4. Conclusions

A hybrid numerical method that coupled the mesh and particle has been developed for the unsteady, incompressible, multiphase flows. One phase (usually the liquid phase) is represented by the particles and another phase (usually the gas phase) is defined on mesh. The formulation of discontinuous interface problem as a continuum problem is realized by extrapolating the density and viscosity of particles onto mesh. The conservation equations are

solved on the stationary mesh with finite-volume method, and the sharp interface is automatically tracked by the moving particles. This prevents both numerical diffusion and oscillations and allows surface tension to be incorporated in a natural way. The original Eulerian mesh is retained through the simulation. A simple Successive Over Relaxation (SOR) method is used in solving the pressure equation, which is successful in large density ratio problems. Since a conservative finite volume scheme is used, no special treatment is needed near the interface. The use of a regular fixed mesh makes coding the method relatively simple. Several test cases suggest that the method is both robust and computationally efficient.

Fig. 18 Kelvin-Helmholtz instability.

The Continuum Surface Force (CSF) model is used to calculate the surface tension effects at the interfaces and the wall adhesion. The surface tension model has been validated on both static and dynamic interfaces having surface tension. It has been applied successfully to a number of fluid flows driven by surface tension.

There is no limitation of the method for three-dimensional simulations. Furthermore, the interface is determined by the distribution of particles, no special algorithm is necessary to deal with interface interactions. The mass conservation of different phases is also preserved by the Lagrangian particles.

Although the present method has been tested extensively on two-dimensional problems, the major purpose of developing this method is to simulate fully three-dimensional multiphase flows. The test cases presented here suggest that large simulations involving interacting interfaces are entirely feasible. The implementation of the method in three dimensions appear to be straightforward and the method would be extended to many new and physically interesting problems.

Acknowledgments

This research was partly funded by Japan Society for the Promotion of Science (FY2002 JSPS Postdoctoral Fellowship for Foreign Researchers, ID No. P02121).

References

- (1) Harlow, F. H. and Welch, J. E., "Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface," *Phys. Fluids*, 8 (1965), pp. 2182-2189.
- (2) Hirt, C. W. and Nichols, B. D., "Volume of fluid (VOF) method for the dynamics of free boundaries," *J. Comput. Phys.*, 39 (1981), pp. 201-225.
- (3) Yabe, T., Xiao, F. and Utsumi, T., "The constrained interpolation profile method for multiphase analysis," *J. Comput. Phys.*, 169 (2001), pp. 556-593.
- (4) Jacqmin, D., "Calculation of two-phase Navier-Stokes flows using phase-field modeling," *J. Comput. Phys.*, 155 (1999), pp. 96-127.
- (5) Floryan, J. M. and Rasmussen, H., "Numerical methods for viscous flows with moving boundaries," *Appl. Mech. Rev.*, 42(12) (1989), pp.323-341.
- (6) Unverdi, S. O. and Tryggvason, G., "A front-tracking method for viscous, incompressible, multi-fluid flows," *J. Comput. Phys.*, 100 (1992), pp.25-37.
- (7) Amsden, A. A., "The particle-in-cell method for the calculation of the dynamics of compressible fluids," Los Alamos Scientific Laboratory Report LA-3466 (1966).
- (8) Gingold, R. A. and Monaghan, J. J., "Kernel estimates as a basis for general particle methods in hydrodynamics," *J. Comput. Phys.*, 46 (1982), pp. 429-453.
- (9) Koshizuka, S., Okamoto, K. and Furuta, K., "Development of computational techniques for nuclear engineering," *Prog. in Nucl. Energy*, 32(1/2) (1998), pp. 209-222.
- (10) Koshizuka, S., Nobe, A. and Oka, Y., "Numerical analysis of breaking waves using the moving particle semi-implicit method," *Int. J. Numer. Methods Fluids*, 26 (1998), pp. 751-769.
- (11) Nomura, K., Koshizuka, S., Oka, Y. and Obata, H., "Numerical analysis of droplet breakup behavior using particle method," *J. Nucl. Sci. Technol.*, 38(12) (2001), pp. 1057-1064.
- (12) Yoon, H. Y., Koshizuka, S. and Oka, Y., "Direct calculation of bubble growth, departure, and rise in nucleate pool boiling," *Int. J. Multiphase Flow*, 27 (2001), pp. 277-298.
- (13) Heo, S., Koshizuka, S. and Oka, Y., "Numerical analysis of boiling on high heat-flux and high subcooling condition using MPS-MAFL," *Int. J. Heat Mass Transfer*, 45 (2002), pp. 2633-2642.
- (14) Ikeda, H., Koshizuka, S., Oka, Y., Park, H. S. and Sugimoto, J., "Numerical analysis of jet injection behavior for fuel-coolant interaction using particle method," *J. Nucl. Sci. Technol.*, 38(3) (2001), pp. 174-182.
- (15) Koshizuka, S. and Oka, Y., "Moving-particle semi-implicit method for fragmentation of incompressible fluid," *Nucl. Sci. Eng.*, 123 (1996), pp.421-434.
- (16) Koshizuka, S., Ikeda, H. and Oka, Y., "Numerical analysis of fragmentation mechanisms in vapor explosions," *Nucl. Eng. Des.*, 189 (1999), pp. 423-433.
- (17) Chikazawa, Y., Koshizuka, S. and Oka, Y., "A particle method for elastic and visco-plastic structure and fluid-structure interactions," *Comput. Mech.*, 27 (2001), pp. 97-106.
- (18) Brackbill, J. U., Kothe, D. B. and Zemach, C., "A continuum method for modeling surface tension," *J. Comput. Phys.*, 100 (1992), pp. 335-354.
- (19) De Silva, I. P. D., Fernando, H. J. S., Eaton, F. and Hebert, D., "Evolution of Kelvin-Helmholtz billows in nature and laboratory," *Earth Planet. Sci. Lett.*, 143 (1996), pp. 217-231.