Manuscript

A variational approach to motion of triple junction of gas, liquid and solid

Kensuke Yokoi [a,b,c,1]

[a] *Mathematics Department, University of California, Los Angeles CA 90095-1555, USA*

[b] *Mathematics Department, Massachusetts Institute of Technology, Cambridge MA 02139-4307, USA*

[c] *Department of Electronics and Mechanical Engineering, Chiba University, Chiba 263-8522, Japan*

Abstract

We propose a simple and robust numerical algorithm to deal with motion of triple junctions of gas, liquid (or two kinds of fluid) and solid based on the level set method [1–4]. In Eulerian framework, to simulate interaction between a moving solid object and an interfacial flow, we need to define at least two functions (level set functions) to distinguish three materials. In such simulations, in general two functions overlap and/or disagree due to numerical errors such as numerical diffusion. In this paper, we resolved the problem using the idea of the active contour model [5–8] introduced in the field of image processing.

Key words: triple junction, multi-phase/material flow, level set method, active contour model

---

[1] Corresponding author: Kensuke Yokoi,
Postal address:
UCLA, Department of Mathematics

Preprint submitted to Elsevier Science *27 April 2006*

1 Introduction

Flow phenomena arising from the interaction of free surface and moving solid objects produce many interesting and practical problems such as biolocomotion on water [9,10]. However attempts of numerical simulations for these phenomena generate some difficulties. One of such difficulties can be found when dealing with triple junctions of gas, liquid and solid.

Numerical simulations for free surface flows with moving structures based on Eulerian grids have been conducted by many researchers [11–17]. In the numerical simulations of [11–14], Eulerian approach has been used for both solid and fluid. As another approach [15,16], Eulerian and Lagrangian approaches are used for fluid and solid, respectively. In [17], a hybrid method has been introduced. Namely Lagrangian particles and Eulerian grid are used for fluid, and Lagrangian approach is used for structure(thin object).

Fig. 1 shows a numerical result for a solid body that falls into a liquid by a method, for which the solid body is treated and the liquid is by a Lagrangian framework and an Eulerian framework, respectively. In the simulation, overlap of the liquid region and the solid region is observed as shown in Fig. 1. As time increases, the overlap region expands. Although this is an example of

Box 951555 Los Angeles, CA 90095-1555 USA
Tel. +1-310-825-2173
Fax. +1-310-206-6673,
E-mail: kensuke@kensukie.biz

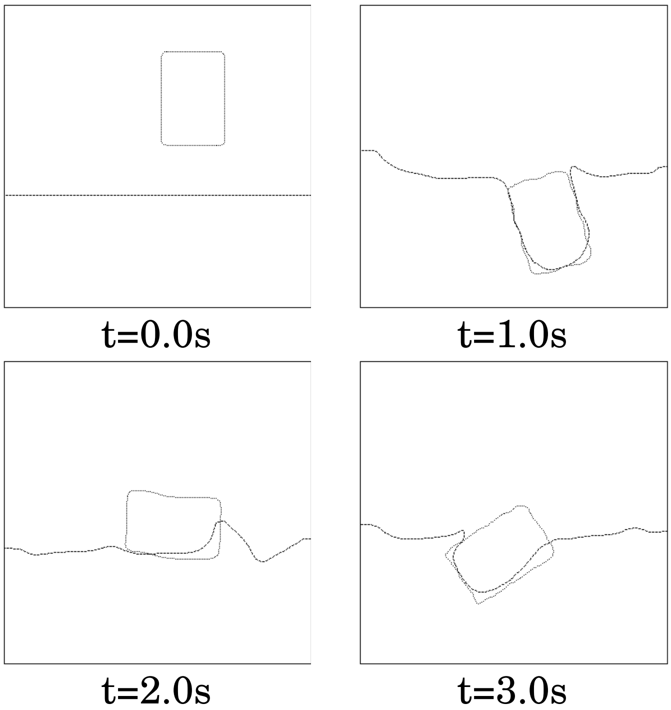

Fig. 1. Time series of a numerical simulation of interaction between the rigid body and the liquid surface by an old method. 40×40 grids are used. The rectangular rigid body is represented by 8×12 grids.

overlap, if the liquid region and the solid region disagree in the liquid, bubbles will be generated from the solid interface. This kind of disagreement of these interfaces disturbs accurate calculation.

However it is difficult to avoid this kind of overlap and disagreement in the framework in which Eulerian and Lagrangian approaches are used for liquid and solid, respectively. For instance, we consider the case of a triple junction as shown in Fig. 2. Fig. 2 (a) shows the initial condition and does not have overlap and disagreement of the solid and the liquid. The gas, liquid and solid are evolved by an appropriate velocity field. If the Lagrangian approach is used for solid, the shape of the solid is kept in time evolution. However, in Eulerian

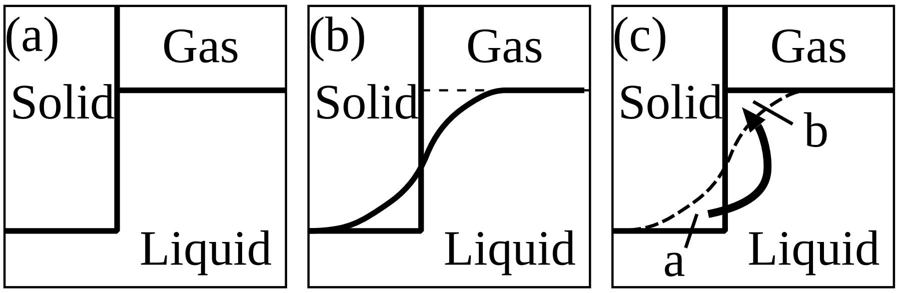

Fig. 2. Problem in triple junction of gas, liquid and solid(b), and the solution for the problem(c). (a) is the initial condition.

framework, numerical diffusion at the interface, especially at sharp corners, is not avoidable as shown in Fig. 2 (b). Even in the cases of no sharp corners, it would be difficult to avoid the accumulation of overlap and/or disagreement in subgrid scale in each time step. One may then suggest that Lagrangian approach should be used for liquid, however it is difficult to deal with large deformation of liquid. If we use Eulerian approach for both liquid and solid, the problem of overlap and disagreement becomes worse.

Altough to avoid such problem, some methods [18,19] to deal with triple junction have been proposed, we introduce another approach based on the active contour model [5–8] using in the field of image processing. The concept is quite simple. We just transport the liquid overlapping with the solid (region a in Fig. 2 (c)) to around the triple junction (region b in Fig. 2 (c)). The basic idea in the active contour model is to evolve a curve, subject to constraints from a given image, in order to detect objects in that image. The active contour model involves an edge detector, which depends on the gradient of the image, to stop the evolving curve at the boundary of the target object. In the paper, the active contour model is used to match a curve of the liquid surface with the solid surface, and only edge detector is used because the mismatch of two level set functions are always within one mesh in a time step because of the CFL condition. We modified the active contour model so that the conservation is satisfied. More precisely we enforce the areas of a and b in Fig. 2 (c) to be the same.

2 Motion of gas, liquid and solid

Two level set functions ψ_{liquid} and ψ_{solid} are defined for the liquid and the solid body, respectively. As a level set function, we used the signed distance function

$$\begin{aligned} \psi &= 0 \text{ at the interface} \\ |\nabla\psi| &= 1 \text{ for the whole region.} \end{aligned} \tag{1}$$

The density (color) functions to distinguish liquid and solid are generated by using a smoothed Heaviside function

$$\phi_{liquid} = H_\alpha(\psi_{liquid}), \tag{2}$$

$$\phi_{solid} = H_\alpha(\psi_{solid}), \tag{3}$$

where

$$H_\alpha(\psi) = \begin{cases} 0 & \text{if} \;\; \psi < -\alpha \\ \frac{1}{2}[1 + \frac{\psi}{\alpha} + \frac{1}{\pi}\sin(\frac{\pi\psi}{\alpha})] & \text{if} \;\; |\psi| \le \alpha \\ 1 & \text{if} \;\; \psi > \alpha \end{cases} \tag{4}$$

Here 2α represents the distance of the transition region. In this paper, we set $2\alpha = \Delta x$. The density function of the air is calculated as follows:

$$\phi_{air} = 1 - \phi_{liquid} - \phi_{solid}. \tag{5}$$

The time evolution of ψ_{liquid} is calculated in an Eulerian framework by the advection equation

$$\frac{\partial \psi_{liquid}}{\partial t} + (\mathbf{u} \cdot \nabla)\psi_{liquid} = 0. \tag{6}$$

In the paper, the CIP method [20–22] is used for (6). Other methods such as ENO [23,24] and WENO [25,26] would be also useful. To maintain the property of the sign distance function, reinitialization is done by solving the following problem to a steady state,

$$\frac{\partial \psi_{liquid}}{\partial t_l} = S(\psi_0)(1 - |\nabla \psi_{liquid}|), \tag{7}$$

where t_l is a fictitious time, ψ_0 is the value of ψ_{liquid} immediately after the calculation for (6) and S is a smoothed sign function $S(\psi_0) = \dfrac{\psi_0}{\sqrt{\psi_0^2 + \varepsilon^2}}$. See [2] for more details.

The motion of a solid ψ_{solid} is computed using a Lagrangian framework and determined by

$$\mathbf{u}_{solid} = \mathbf{u}_{center} + \mathbf{r} \times \omega, \tag{8}$$

where $\mathbf{u}_{center}$ is the velocity at the mass center, $\mathbf{r}$ is the vector from the mass center, and ω is the angular velocity at the mass center. To couple the Lagrange solid object with the Eulerian grid, the level set function for the Lagrange solid object is constructed by using a method such as the fast marching method [27,3,28,13].

However in such formulation, two level set functions overlap and/or disagree as shown before. To avoid the problem, we introduce the active contour model.

3 Active contour model

3.1 Active contour model for image

The active contour model has been used for image processing such as image segmentation. Let us consider an edge penalty function $g(x, y)$:$[0, 1] \to R^2$, where for example

$$g(x, y) = \frac{1}{1 + |\nabla I(x, y)|^2}, \tag{9}$$

here $I(x, y)$ is the data image. The function g indicates the presence of edges

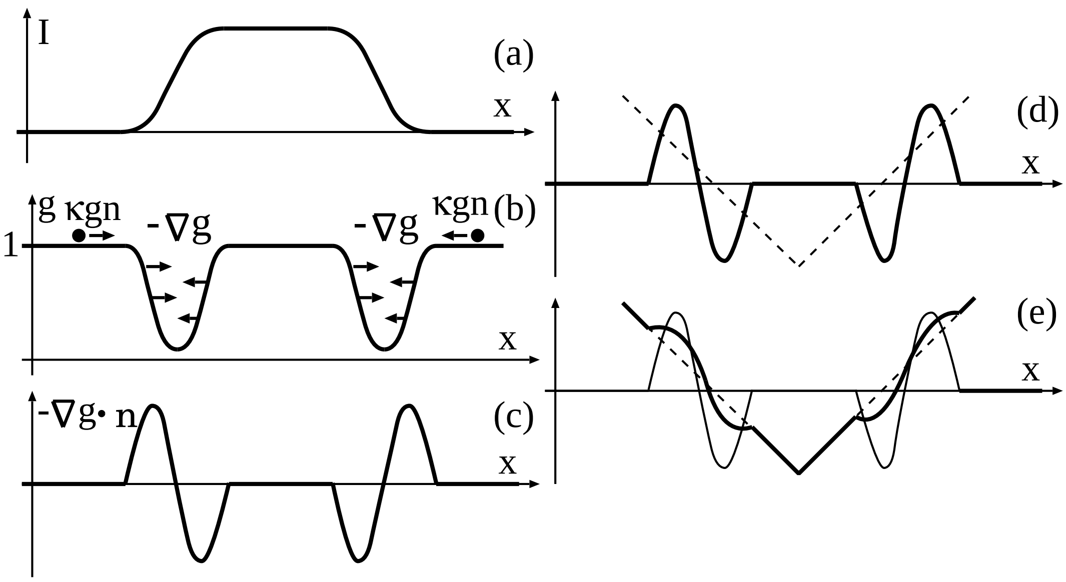

Fig. 3. Schematic figure of the active contour model.

in the image. That is, high gradient magnitudes in the image that indicate the possible presence of an edge are mapped by g to small values, while flat regions in the image are mapped to one as illustrated in Fig. 3 (b). The model minimised an energy functional along a general parametric curve $C(p)$:$[0,1] \to R^2$ given by

$$E(C) = \int_0^1 g(C(p))dp. \tag{10}$$

for which the first variation is given by

$$\frac{\delta E(C)}{\delta C} = C_t = -(\kappa g - \nabla g \cdot \mathbf{n})\mathbf{n}, \tag{11}$$

where κ is the curvature and $\mathbf{n}$ is the inward normal for C. The detected object is then given by the steady state that is $C_t = 0$.

We can derive a level set formulation of (11) by using the variational level set formulation [19]

$$\psi_t = \mp(|\nabla\psi|g\kappa + \nabla g \cdot \nabla\psi), \tag{12}$$

where C is represented by the zero level set, "$-$" sign is for that ψ of inside of C is less than 0, and "$+$" sign is for that ψ of inside C is larger than 0. Hereafter we assume that ψ of inside C is always larger than 0. First term of the right hand side of (11) and (12) means the well-known mean curvature motion (or shortening). This term works mainly on the region of $g = 1$, i.e. except near the edge. This flow decreases the value of maxima/minima curvature. Therefore, it has the properties of shortening (an initial curve shrinks to a point in finite time with asymptotically circular shape) as well as smoothing (points with high curvature evolve faster and disappear asymptotically). Near the edge, the influence of the first term is reduced because g becomes small and the second term attracts C to the edge. In the level set formulation, C is attracted as shown in Fig.3 (d)(e). If the gradient of g is added to the level set function, the zero level set approach to the edge.

3.2 Active contour model for triple junction

Our purpose is to adjust the mismatch of the liquid interface and the solid interface. The mismatch is always within one mesh spacing because of the

CFL condition. In this case, the initial C of the liquid surface is always near the target solid shape. Therefore we omit the first term of (12) and rewrite (12) as follows:

$$\psi_t = \nabla g_1 \cdot \nabla \psi_{fluid}. \tag{13}$$

To make the formulation simple, g of (9) is also rewritten as follow:

$$g_1(x, y) = -|\nabla \phi_{solid}|, \tag{14}$$

where

$$\phi_{solid} = H_\beta(\psi_{solid}). \tag{15}$$

The 2β is the width of g_1. The meanings of (9) and (14) are almost same except the magnitude. Although (13) depends on the gradient of g, it does not depend on the magnitude. Therefore this rewriting does not change the meaning. Although we can use the original g of (9), the reformulation will then be slightly complicated.

In this case, the level set function of the solid (target shape) is given. Therefore we can calculate directly g_1 and ∇g_1 from the level set function ψ_{solid} as follows:

$$g_1 = -|\phi_{gac}| = \begin{cases} -\frac{1}{2\beta}(1 + cos(\frac{\pi}{\beta}\psi)) & \text{if } -\beta \leq \psi \leq \beta \\ 0 & \text{if otherwise} \end{cases} \tag{16}$$

$$\nabla g_1 \cdot \nabla \psi_{liquid} = \begin{cases} -\frac{\pi}{2\beta^2} sin(\frac{\pi}{\beta}\psi) \text{ if } & -\beta \leq \psi \leq \beta \\ 0 & \text{if otherwise} \end{cases} \quad (17)$$

here we used $|\nabla \psi_{liquid}| = 1$ for (17). In the formulation, ∇g is determined directly from the level set function without discretization.

3.3 Conservation

Although the liquid surface is attracted to the solid surface in the above formulation, conservation is not taken into account. To take into account it, we modify (13) as follow:

$$\frac{\partial \psi_{fluid}}{\partial t_g} = \begin{cases} \nabla g_1 \cdot \nabla \psi_{fluid}, & \text{if} \quad 0 \leq \psi_{solid} \leq \beta \quad \text{(inside of solid)} \\ \frac{\Delta A}{|\Delta A + \epsilon_1|} \nabla g_1 \cdot \nabla \psi_{fluid}, & \text{if} \quad -\beta \leq \psi_{solid} < 0 \text{ (outside of solid)} \end{cases} \quad (18)$$

with

$$A(t_g, t) \equiv \int \int H_\beta(\psi_{fluid}(t_g, t)) dxdy, \quad (19)$$

$$\Delta A = -(A(t_g, t) - A(0, t)), \quad (20)$$

here ϵ_1 is a small positive constant to avoid 0 division ($\epsilon = 10^{-7}$ is used in our calculation) and $A(t_g, t)$ is the total area of the liquid. t_g is an artificial time for (18) and t are the time for fluid and solid calculation. (18) is solved to a steady state i.e. $\Delta A = 0$. However actually we use $|\Delta A| < \epsilon_2$ for computation, here ϵ_2 is a small positive constant and $\epsilon_2 = 0.001 \Delta x \Delta y$ is used. This means that the error is less than 0.1% of one grid size in each time step. (18) removes the overlap region (region a in Fig. 2 (c)) unconditionally and generate the new liquid area around triple junctions outside the solid (region b in Fig. 2 (c)) as conservation is satisfied.

In the formulation, although conservation is guaranteed in the calculation of (18), it is not guaranteed in the calculation of fluid, because $A(0,t)$ of (20) may have errors in fluid calculation. These errors may increase as the time increases. To guarantee the conservation from (18), (20) is replaced by

$$\Delta A = -(A(t_g, t) - A(0,0)). \tag{21}$$

Then $A(t_g, t)$ is always attracted to the initial value $A(0,0)$. If we assume that most of errors arise around the solid interface, (21) may be suitable.

The width of g_1, β of (18), should be a small value, because the new area should be generated near the solid interface as much as possible. However if we use β of a small constant, such as $\beta = \Delta x$ and $\beta = 1.5\Delta x$, oftentimes (18) does not converge especially in the case when the convergence tolerance ϵ_2 is so small. Cancellation of significant digits also happens rarely, because ∇g_1 becomes so small or 0, depends on the positions of the grid and the solid interface. If this happens at the triple junctions, (18) can not converge. However if we use a rather large β such as $2\Delta x$ and $3\Delta x$, then (18) converges well. Therefore in our calculation, we used $\beta(t_g)$ which depends on t_g. As an example, we used $\beta(t_g) = (1 + 0.5int(itr/50))\Delta x$, here $int(a)$ is the integer part of a and itr is a number of iterations of (18). First $\beta = \Delta x$ is used. If it does not converge in 50 iterations, we change β into $\beta + 0.5\Delta x$. If this $\beta(t_g)$ is used, the calculation becomes robust.

4 Numerical result

We applied the present method to the same problem described in Fig. 1. Fig. 4 shows the time series of the result using the present method with (20). We can see the present method can prevent overlap well. Fig. 5 represents the

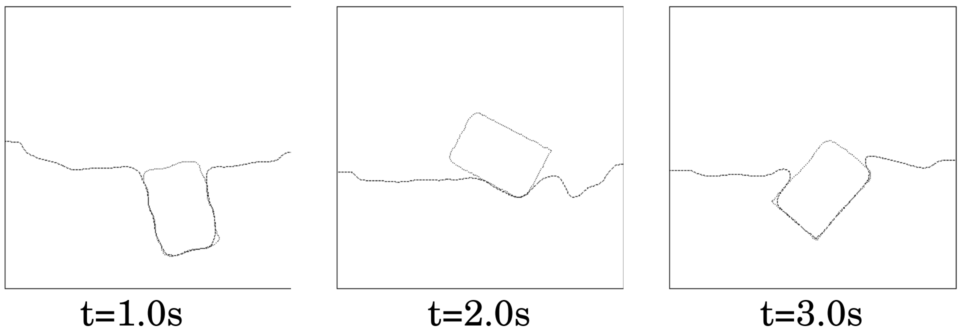

Fig. 4. Time series of a numerical simulation of interaction between the rigid body and the liquid surface by the present method with (20). 40×40 grids are used. The rectangular rigid body is represented by 8×12 grids.

result using the present method with (21). This formulation can also prevent overlap well.

Fig. 6 shows the time evolution of the total area of the liquid. The present method with (20) has the error of a few percent. However the error is not a result of the calculation of (18). It is from the fluid calculation via $A(0,t)$. If we use (21) instead of (20), then the error is less than 10^{-5} when the convergence

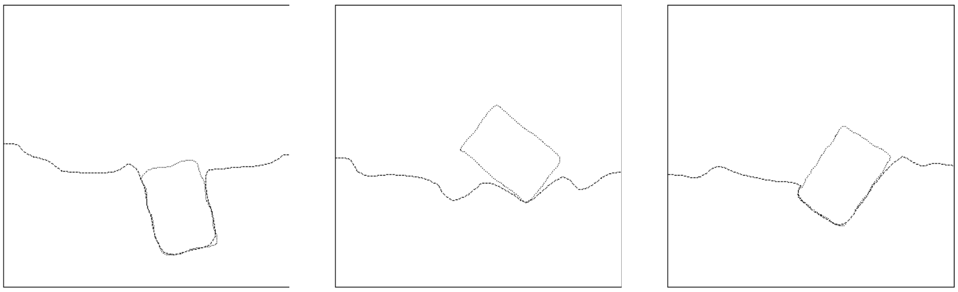

Fig. 5. Time series of a numerical simulation of interaction between the rigid body and the liquid surface by the present method with (21).

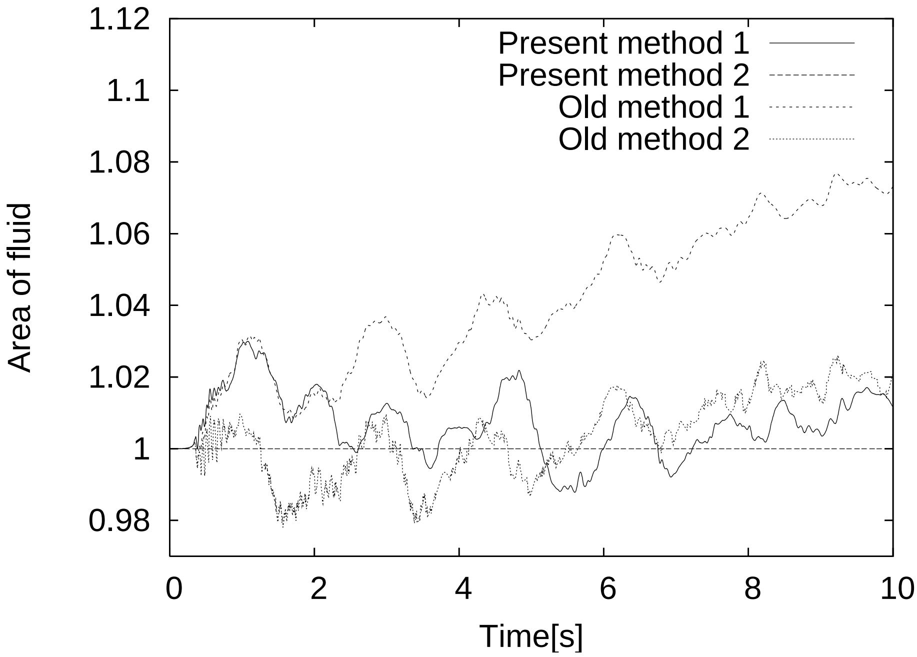

Fig. 6. Time series of the total area of the liquid. Present methods 1 and 2 are results by the present methods with (20) and (21), respectively. Old methods 1 and 2 are results by the old method. In old method 1 and 2, the total area of the liquid are calculated including overlap region and not including it, respectively.

tolerance of (21) $\epsilon_2 = 0.001\Delta x\Delta y$ is used. This means that (18) does not make an error associated with the conservation. As comparison, the results by the method of Fig. 1 is also plotted as old method 1 and 2 in Fig. 6. In old method 1 and 2, the total area of the liquid are calculated including overlap region and not including it, respectively.

5 Summary

We have proposed a numerical algorithm to prevent overlap and disagreement of solid and fluid interfaces. The method is constructed based on the level set method and the active contour model. The method has been shown to simulate the interaction of gas, liquid and solid robustly.

6 Acknowledgements

We would like to thank Luminita Vese for the lecture of Mathematical Models for Image Analysis in UCLA Mathematics department. We would also like to thank Stanley Osher, Young-Ju Lee, Jason Chung, Triet Le, Linh Lieu, Shing-yu Leung, He Lin, Shin Matsutani and Hao Liu for their useful discussions and comments. This work is supported in part by a Grant-in-Aid for JSPS fellow (17-4063) of Japan Society for the Promotion of Science. Numerical computations in this work was partially carried out at the Yukawa Institute for Theoretical Physics, Kyoto University.

A Dynamics of fluid and solid

The dynamics of fluid is determined by the continuum equation and the Navier Stokes equation:

$$\nabla \cdot \mathbf{u} = 0, \tag{A.1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{\nabla p}{\rho} + \frac{\nabla \cdot \tau}{\rho} + \frac{\mathbf{f}}{\rho}, \tag{A.2}$$

where $\mathbf{u}$ is the velocity, p the pressure, ρ the density, τ the viscous stress tensor and $\mathbf{f}$ the body force. The set of equation solved by the fractional step method [29].

The dynamics of solid is determined by the equations

$$\frac{d}{dt}(M\mathbf{u}_{center}) = \mathbf{F} = \int \rho \frac{d\mathbf{u}}{dt} \phi_{solid} dV \tag{A.3}$$

and

$$\frac{d}{dt}(\mathbf{I} \cdot \omega) = \mathbf{T} = \int (\mathbf{r} \times \rho \frac{d\mathbf{u}}{dt}) \phi_{solid} dV, \tag{A.4}$$

here M is the mass of the solid, $\mathbf{F}$ is the force for the solid body, $\mathbf{I}$ the tensor of inertia moment and $\mathbf{T}$ the torque. The force and the torque are calculated by a volume force formulation. See [11,12] for more details.

References

[1] S. Osher and J.A. Sethian, Front propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulation, J. Comput. Phys. **79**, 12 (1988).

[2] M. Sussman, P. Smereka and S. Osher, A level set approach for capturing solution to incompressible two-phase flow, J. Comput. Phys. **114**, 146 (1994).

[3] J.A. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press (1999).

[4] S. Osher and R. Fedkiw, Level Set Methods and Dynamics Implicit Surface, Applied Mathematical Sciences 153, Springer (2003).

[5] M. Kass, A. Witkin and D. Terzopoulos, Snakes: Active Contour Models, International Jonrnal of Computer Vision **1**, 321 (1988).

[6] V. Caselles, R. Kimmel, G. Sapiro, Geodesic Active Contours, International Jonrnal of Computer Vision **22**, 61 (1997).

[7] G. Sapiro, Geometric Partial Differential Equations and Image Analysis, Cambridge University Press (2001).

[8] R. Kimmel, Numerical Geometry of Images: Theory, Algorithms, and Applications, Springer-Verlag (2003).

[9] D.L. Hu, B. Chan and J.W.M Bush, The hydrodynamics of water strider locomotion, Nature, **424** 663 (2003).

[10] D.L. Hu and J.W.M Bush, Meniscus-climbing insects, Nature, **437** 733 (2005).

[11] F. Xiao et al., Comput. Phys. Commun. **102** 147 (1997).

[12] F. Xiao, J. Comput. Phys. **155**, 348 (1999).

[13] K. Yokoi, Numerical method for complex moving boundary problems in a Cartesian fixed grid, Phys. Rev. E **65** 055701(R) (2002).

[14] K. Yokoi, Numerical method for moving solid object in flows, Phys. Rev. E **67** 045701(R) (2003).

[15] T. Takahashi, et al., Realistic animation of fluid with splash and foam, Computer Graphics Forum(In Eurographics 2003 Proceedings) **22**, 716 (2003).

[16] K. Takizawa, et al., Simulation and experiment on swimming fish and skimmer by CIP method, Computers & Structures **83**, 397 (2005).

[17] E. Guendelman, et al., Coupling Water and Smoke to Thin Deformable and Rigid Shells, SIGGRAPH 2005, ACM TOG **24**, 973 (2005).

[18] B. Merriman, J. Bence and S. Osher, Motion of Multiple Junctions: A Level Set Approach, J. Comput. Phys. **112** 334 (1994).

[19] Hong-Kai Zhao, T. Chan, B. Merriman and S. Osher, A Variational Level Set Approach to Multiphase Motion, J. Comput. Phys. **127**, 179 (1996).

[20] T. Yabe and T. Aoki, A universal solver for hyperbolic-equation by cubic-polynomial interpolation. I. One-demensional solver, Comput. Phys. Commun. **66** 219 (1991).

[21] T. Yabe et al, A universal solver for hyperbolic-equation by cubic-polynomial interpolation. II. Two- and three dimensional solvers, Comput. Phys. Commun. **66** 233 (1991).

[22] T. Yabe, F. Xiao and T. Utsumi, The Constrained Interpolation Profile Method for Multiphase Analysis, J. Comput. Phys. **169**, 2 (2001).

[23] A. Harten, B. Engquist, S. Osher and S. Chakravarthy, Uniformly High-Order Accurate Essentially Non-Oscillatory Schemes III, J. Comput. Phys. **71**, 231 (1987).

[24] C.W. Shu and S. Osher, Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes, J. Comput. Phys. **77**, 439 (1988).

[25] X.D. Liu, S. Osher and T. Chan, Weighted Essentially Non-Oscillatory Schemes, J. Comput. Phys. **115**, 200 (1994).

[26] G.S. Jiang and C.W. Shu, Efficient Implementation of Weighted ENO Schemes, J. Comput. Phys. **126**, 202 (1996).

[27] D. Adalsteinsson and J.A. Sethian, The Fast Construction of Extension Velocities in Level Set Methods, J. Comput. Phys. **148**, 2 (1999).

[28] Y.R. Tsai, Rapid and accurate computation of the distance function using grids, J. Comput. Phys. **178** 175 (2001).

[29] J. Kim and P. Moin, Applications of a fractional step method to incompressible Navier-Stokes equations, J. Comput. Phys. **59**, 308 (1985).