International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:5, No:4, 2011

GPU implementation for solving incompressible two-phase flows

Sheng-Hsiu Kuo, Pao-Hsiung Chiu, Reui-Kuo Lin, and Yan-Ting Lin

Abstract—A one-step conservative level set method, combined with a global mass correction method, is developed in this study to simulate the incompressible two-phase flows. The present framework do not need to solve the conservative level set scheme at two separated steps, and the global mass can be exactly conserved. The present method is then more efficient than two-step conservative level set scheme. The dispersion-relation-preserving schemes are utilized for the advection terms. The pressure Poisson equation solver is applied to GPU computation using the *pCDR* library developed by National Center for High-Performance Computing, Taiwan. The SMP parallelization is used to accelerate the rest of calculations. Three benchmark problems were done for the performance evaluation. Good agreements with the referenced solutions are demonstrated for all the investigated problems.

Keywords—conservative level set method; two-phase flow; dispersion-relation-preserving; Graphics Processing Unit (GPU); Multi-threading.

I. Introduction

THE most common incompressible flow algorithms that have been applied to track the air/water interfaces include vortex method [1], boundary integral method [2], volume of fluid (VOF) method [3], front tracking method [4], and level set method [5], [6]. The VOF method can conserve the volume exactly, but the reconstruction of the interface is a main issue. The interface reconstruction is also the problem for the front tracking method. For the level set method, interface can be implicitly defined with the zero-contour of the level set function. However, how to obtain the mass-conserving solutions are the main tasks.

Recently, the CLSVOF (coupled level set and volume-of-fluid) method [7], THINC (tangent of hyperbola for interface capturing) method [8], [9], VOSET (volume-of-fluid and level set) method [10], and conservative level set method [11] have been proposed to resolve the problems arisen from the VOF or/and level set method. These schemes are can obtain the mass-conserving and accurate-interface solutions.

There were many applications using GPU implementation in computational fluid dynamics (CFD). Krüger and Westermann [12] proposed a framework for the implementation of direct solvers for sparse matrices, and applied to 2D wave equation and the incompressible Navier-Stokes equations. Goodnight, Woolley, Lewin, Luebke and Humphreys [13] presented the solver for the boundary value heat and fluid flow problems using GPU implementation. A Navier-Stokes flow solver for structured grids using GPU was also presented in [14]. Hagen, Lie, and Natvig [15] presented the implementations to compressible fluid flows using GPU. Brandvik and Pullan [16] presented 2D and 3D Euler equations solvers on GPU and focus on performance comparisons between GPU and CPU codes based on considerable speed-ups using exclusively structured grids. Corrigan, Camelli, Löhner, and Wallin [17] presented a GPU solver for inviscid, compressible flows on 3D unstructured grids. All the presneted resutls have shown that considerably computational time can be reduced by using the GPU implementation.

In this paper, we combine the GPU and multi-CPU cores to develop a global mass correction method based one-step conservative level set method. Good performance can be obtained by involving GPU to accelerate the Poisson solver with using multi-CPU cores for rest of calculations.

This paper is organized as follows. Section II presents the derivation for the conservative level set method. This is followed by the presentation of the differential equations governing the motion of two fluids and the dispersion-relation-preserving schemes. Section IV and V present the simulated results to show the applicability and efficiency for the proposed framework. Finally, we draw some concluding remarks in Section VI.

II. Conservative level set equation

The conservative level set method is firstly proposed by Olsson and Kreiss [11]. Instead of using sign distance function [5], they proposed the following smoothed heaviside step function Φ in their previous work:

$$\Phi = \frac{1}{1+\exp(-s/\varepsilon)} \tag{1}$$

By solving the following two equations, the interface profiles can be tracked by the conservative level set method:

$$\frac{\partial \Phi}{\partial t} + \nabla \cdot (\underline{u}\Phi) = 0 \tag{2}$$

$$\frac{\partial \Phi}{\partial \tau} + \nabla \cdot (\Phi(1-\Phi)\underline{n}) = \varepsilon \nabla \cdot (\nabla \Phi) \tag{3}$$

where $\underline{u}$ is the divergence-free velocity vector, $\underline{n}(=\frac{\nabla\phi}{|\nabla\phi|})$ is the unit vector normal to the interface, and ε is the coefficient of the transition region which makes Φ vary between 0 to 1 with the width of $3\sqrt{2}\varepsilon$. Eq. (2) is a advection equation for Φ, and Eq. (3) is the compression-diffusion equation, which is proposed by Olsson and Kreiss [11] to stably preserve the

S. H. Kuo is with the National Center for High-Performance Computing , Hsinchu, Taiwan, 30076 ROC e-mail: sheng_hsiu@nchc.narl.org.tw.
P. H. Chiu is with the Institute of Nuclear Energy Research , Longtan, Taiwan, 32546 ROC e-mail: phchiu@iner.gov.tw.
R. K. Lin is with the Taiwan Typhoon and Flood Research Institute , Taipei, Taiwan, 10093 ROC e-mail: rklin@ttfri.narl.org.tw.
Y. T. Lin is with the Institute of Nuclear Energy Research , Longtan, Taiwan, 32546 ROC e-mail: yantinglin@iner.gov.tw.

sharp interface. These two equations can be seen as the first step (advection step) and second step (re-initialization step) for the original level set method [5]. Note that theoretically we need to solve Eq. (3) until the steady state is reached.

The conservative level set method is known to have a better mass conservation, as shown by Olsson and Kreiss [11]. Moreover, the total mass can be exactly conserved if one uses a conservative discretization. However, The second step equation is computational intensive, due to the fact that we need to solve it until steady-state is reached. Although only few time steps are sufficient in practice [11], [18], the computational cost is still high. This motivated us to propose a one-step conservative level set method.

A. One-step conservative level set equation

The idea for the one-step conservative level set method is to combine the advection step (Eq. (2)) and the compression-diffusion step (Eq. (3)) into one equation. This will lead the following one-step conservative level set equation :

$$\frac{\partial \Phi}{\partial t} + \nabla \cdot (\underline{u}\Phi) = \gamma \nabla \cdot (\Phi(1-\Phi)\underline{n} - \varepsilon \nabla \Phi) \tag{4}$$

where introduced γ is the reinitialization parameter coefficient. In the present study, ε and γ are chosen as $0.7\Delta x$ and $|\underline{u}|$. It can be expected that the present method is more efficient than the original conservative level set method due to the fact that the proposed method is a one-step method.

B. Global mass correction method

Even we uses a conservative discretization for the conservative level set method, the truncation errors will make the total mass change with small magnitude. In order to conserve the global mass exactly, we propose a global mass correction method in present study. The idea is to re-distribute the summation of the loss/increase values in the transition region. Here is the summary for the global mass correction method.

(i)compute the total mass M at present time t;

$$M = \int_{\Omega} \Phi d\Omega \tag{5}$$

(ii)compute the difference G for the mass between the initial time t_0 and present time t

$$G = M_0 - M \tag{6}$$

where $M_0 = \int_{\Omega} \Phi | t = 0 d\Omega$ is the total mass at initial time t_0.

(iii) uniformly distributed G at N_G meshes as G/N_G, where N_G is the total numbers cells of the transition region ($0.001 < \Phi < 0.999$).

By using the above procedures, the total mass can then be exactly conserved.

III. Two-phase flow solver

For the two immiscible and incompressible fluids, the equations of motion for this two-phase fluids in a gravitational vector field $\underline{g}$ can be represented by the incompressible flow equations given below:

$$\frac{\partial \underline{u}}{\partial t} + (\underline{u} \cdot \nabla)\underline{u} = \frac{1}{\rho}\left(-\nabla p + \nabla \cdot (2\mu \underline{\underline{D}}) + \underline{F_s} + \rho \underline{g}\right) \tag{7}$$

where ρ is the density, μ is the viscosity, $\underline{\underline{D}}$ is the rate of deformation, p is the pressure, and the $\underline{F_s}$ is the surface tension force which is denoted as $\underline{F_s} = \sigma\kappa\delta\underline{n}$. In present study, the surface tension force is approximated using the continuum surface force model (CSF) of Brackbill et al. [19] as $\underline{F_s} = \sigma\kappa\nabla\Phi$. The above equations can be casted in the dimensionless equations as

$$\frac{\partial \underline{u}}{\partial t} + (\underline{u} \cdot \nabla)\underline{u} = \frac{1}{\rho}\left(-\nabla p + \frac{1}{Re}\nabla \cdot (2\mu \underline{\underline{D}}) + \frac{1}{We}\underline{F_s}\right) + \frac{1}{Fr^2}\underline{e}_g \tag{8}$$

where $Re = \frac{\rho_r V_r L_r}{\mu_r}$ is the Reynolds number, $We = \frac{\rho_r V_r^2 L_r}{\sigma}$ is the Weber number and the $Fr = \frac{V_r}{\sqrt{gL_r}}$ is the Froude number. The density and viscosity are approximated as $\rho = \rho_{GL} + (1-\rho_{GL})\Phi$ and $\mu = \mu_{GL} + (1-\mu_{GL})\Phi$, where ρ_{GL} and μ_{GL} are the ratios for the density and viscosity.

A. Semi-implicit Gear scheme and projection method

In present study, the two phase flow equations is discretized by the Gear scheme as:

$$\begin{aligned}\frac{3\underline{u}^{n+1,*} - 4\underline{u}^{n} + \underline{u}^{n-1}}{2\Delta t} &= -2[(\underline{u}\cdot\nabla)\underline{u}]^n + [(\underline{u}\cdot\nabla)\underline{u}]^{n-1} \\ +2\frac{1}{\rho^{n+1}}[\frac{1}{Re}\nabla\cdot(2\mu\underline{\underline{D}} - \nabla^2\underline{u})]^n &- \frac{1}{\rho^{n+1}}[\frac{1}{Re}\nabla\cdot(2\mu\underline{\underline{D}} - \nabla^2\underline{u})]^{n-1} \\ &+ \left((\frac{1}{\rho}(-\nabla p + \frac{1}{Re}\nabla^2\underline{u} + \frac{1}{We}\underline{F_s}) + \frac{1}{Fr^2}\underline{e}_g\right)^{n+1,*}\end{aligned} \tag{9}$$

the above equation is then solved by the standard iterative scheme such as successive overrelaxation (SOR) method. No non-linear iteration is needed for the present semi-implicit scheme.

The intermediate velocity $\underline{u}^{n+1,*}$ is generally not divergence-free. It is then needed to solve correct the velocity and the corresponding pressure to satisfy the divergence-free condition:

$$\frac{3(\underline{u}^{n+1} - \underline{u}^{n+1,*})}{2\Delta t} = -\frac{1}{\rho}\nabla p' \tag{10}$$

$$p^{n+1} = p^{n+1,*} + p' \tag{11}$$

where p' is the pressure correction. Take the divergence on Eq. (10), the following Poisson equation for the pressure correction can be derived

$$\nabla \cdot (\frac{1}{\rho}\nabla p') = \frac{3(\nabla \cdot \underline{u}^{n+1,*})}{2\Delta t} \tag{12}$$

After solving Eq. (12), one can get the corrected velocity $\underline{u}^{n+1} (= \underline{u}^{n+1,*} - \frac{2}{3}\Delta t(\frac{1}{\rho}\nabla p'))$, and the corresponding corrected pressure $p^{n+1} (= p^{n+1,*} + p')$.

B. Multi-dimensional advection scheme for advection terms

In this subsection we show the way of approximating $\underline{u}_x$, which can accommodate the dispersion-relation-preserving property, under the uniform grid ($\Delta x = \Delta y = h$). Referring to Fig. 1, $\underline{u}_x$ at the nodal point (i, j) is assumed to the expressed as

$$\begin{aligned}\underline{u}_x(x,y) \simeq \frac{1}{h}\big(&d_1\underline{u}_{i-1,j-1} + d_2\underline{u}_{i,j-1} + d_3\underline{u}_{i+1,j-1}\\ &+ d_4\underline{u}_{i-1,j} + d_5\underline{u}_{i,j} + d_6\underline{u}_{i+1,j}\\ &+ d_7\underline{u}_{i-1,j+1} + d_8\underline{u}_{i,j+1} + d_9\underline{u}_{i+1,j+1}\\ &+ d_{10}\underline{u}_{i,j-2} + d_{11}\underline{u}_{i,j+2} + d_{12}\underline{u}_{i-2,j}\big) \qquad (13)\end{aligned}$$

Substitution of the Taylor series expansions for $\underline{u}_{i\pm1,j}$, $\underline{u}_{i-2,j}$, $\underline{u}_{i,j\pm1}$, $\underline{u}_{i,j\pm2}$, $\underline{u}_{i\pm1,j\pm1}$ into the above equation, we are led to derive the resulting modified equation for $\underline{u}_x$. The derivation is followed by eliminating eleven leading error terms to yield a system of eleven algebraic equations. One more equation has to be derived so as to be able to uniquely determine $d_1 \sim d_{12}$ shown in Eq. (13). By solving all the equations with the dispersion-relation-preserving equation [20], we can obtain the following coefficients $d_1 = d_3 = d_7 = d_9 = 0$, $d_2 = d_8 = \frac{1}{9}\frac{\pi(3\pi-10)}{(3\pi-8)}$, $d_4 = -1$, $d_6 = \frac{1}{3}$, $d_5 = \frac{1}{6}\frac{3\pi^2-19\pi+24}{(3\pi-8)}$, $d_{10} = a_{11} = -\frac{1}{36}\frac{\pi(3\pi-10)}{(3\pi-8)}$, and $d_{12} = \frac{1}{6}$. $\underline{u}_x$ is also shown to have a spatial accuracy order of three by the resulting modified equation: $\underline{u}_x \simeq \frac{h^3}{12}\underline{u}_{xxxx} + \frac{h^3}{18}\frac{\pi(3\pi-10)}{(\pi^2-6\pi+8)}\underline{u}_{yyyy} - \frac{h^4}{30}\underline{u}_{xxxxx} + \frac{h^5}{72}\underline{u}_{xxxxxx} + \cdots +$ HOT. Noted that the fifth order dispersion-relation-preserving dual-compact scheme is used for approximating the advection terms shown in the conservative level set equation. For the details of derivations, the reader can refer to [21] and [22].

C. Velocity-pressure coupling

When solving the incompressible flow equation with primitive variable form, special care must be taken for the coupling between velocity and pressure. When use the non-staggered grids, simply use the standard central difference for approximating pressure gradient will lead to a unphyscial distribution for the pressure field, known as the odd-even decoupling [23]. While the the odd-even decoupling problem can be eliminated on the staggered grid [23], the resulting programming complexity is still a main task. In the present study, a semi-staggered grid is used for coupling the velocity and pressure [24]. The velocity vectors are stored at the cell edge, and the pressure and other scalar fields are stored at the cell center, as shown in Fig. 2. For this grid system, the programming is much simpler than staggered grid system, and the coupling between velocity and pressure can be easily achieved if one employs a pressure interpolation from cell center to cell edge.

IV. Numerical results

A. Dam-break problem

The dam break problem has been frequently employed to validate the code for predicting free surface hydrodynamics. In the current calculation, the fluid properties is the same as Martin and Moyce [25]. The initially prescribed width(L) and height(2L) of the water column are 0.146 m and 0.292 m, respectively. The liquid density ρ_L = 10^3 kg/m^3, viscosity μ_L = 0.5 Pa s, background gas density ρ_G = 1.0 kg/m^3, viscosity μ_G = 0.5×10^3 Pa s, gravity g = 9.8 m/s^2 and the surface tension coefficient σ = 0.0755 N/m. By choosing the reference velocity as $\sqrt{g(2L)}$, it will lead to $Re = 493.954$, $Fr = 1.414$, $We = 5533.690$, $\rho_{GL} = 0.001$ and $\mu_{GL} = 0.001$. 40×40 and 80×80 meshes are used for the calculations. Good agreements with the experimental results given in [25] and numerical results given in [10] are clearly demonstrated in Fig. 3 for the predicted front location.

B. Rayleigh-Taylor instability problem

Flow instability of the Rayleigh-Taylor type is associated with the penetration of a heavy fluid into a light fluid in the direction of gravity. The interface is given by $y(x) = (2D + 0.1D\ cos(2\pi x/D))$ in the rectangular domain $[0,D]\times[0,4D]$. The Reynolds number Re under investigation is 3000. The density difference is represented by the Atwood number $At = (\rho_L - \rho_G/\rho_L + \rho_G) = 0.5$ and the viscosity ratio is 1. Surface tension force is ignored for this problem. The predicted interface profiles with 100×400 meshes are given in Fig 4(a)-(d). We also compare the top of the rising fluid and the bottom of the falling fluid with the solutions of Guermond et al. [26] and Ding et al. [27]. From the Fig. 4(e), The present method is justified by the good agreements between our solutions and previous studies obtained by [26] and [27].

In order to validate the mass conservations, the ratio of the total mass against time for the above two problems are plotted in Fig. 5. It can can be seen that the total masses are exactly conserved for all the investigated problems.

C. Bubble merging problem

The bubble merging problems with coaxial coalescence is considered here. There are two bubble with radius R in the cubic domain $[0,4R]\times[0,4R]\times[0,8R]$. The upper bubble is at (2R,2R,2.5R) and the lower bubble is at (2R,2R,1R). The Eotvos number ($Eo = \frac{\rho_r g L_r^2}{\sigma}$) is 16, and the Morten number ($Mo = \frac{g\mu_r^4}{\rho_r\sigma^3}$) is 2×10^{-4}. This will lead that Reynolds number is 67.27, Weber number is 16, and the Froude number is 1. The ratios for density and viscosity are $\rho_{GL} = 0.001$ and $\mu_{GL} = 0.01$, respectively. The time-history solutions obtained by $80\times80\times160$ meshes for different physical time are plotted in Fig. 6. The agreement with the experimental observations by Brereton and Korotney [29] can be shown for the present solutions.

V. Parallelization

For the two-phase flow problem, it is essential to get the high quality solution. For example, if the gird size is not fine enough, we can not see the real topology change of the bubble or droplet. However, it will cost a lot of computing time and resources. In order to accelerate the calculations, we use the SMP parallelization and also implement GPU acceleration in this study. We evaluate the performance by solving the 3D coaxial coalescence bubble problem which

have validated in Section IV-C. For a sequential program, high computational cost is an big issue for two-phase flow because it is more difficult to apply fast-solvers for solving the pressure Poisson equation. From our numerical experiments, the Poisson solver will cost almost 80% of total elapsed time for the bubble merging problem with meshes $40 \times 40 \times 80$. With the increasing of mesh size, most of time will spend on the Poisson solver, shown in Fig 7. That is, to speed up the Poisson solver will be the main issue of performance. In present study, we use OpenMP directives for the SMP parallelization and choose the Parallel CDR (*pCDR*) library, which was a set of code for solving a convection–diffusion–reaction scalar transport equation using GPU cards as an accelerator, to solve the pressure Poisson's equation. However, we still have to modified some kernel functions of the Red-black SOR solver to fit to our problem. A single computing node with two Intel Xeon x5472 CPUs and one NVDIA Tesla C1060 GPU card is used for the present study. Details of hardware were shown in Table I.

From Fig. 8 (a), there is 6.2x speed and total speed up ratio will achieve 3.1x on PPE solver by using 8 CPU cores with meshes $40 \times 40 \times 80$. When GPU was involved in calculation, we can only get 1.5x faster because the latency of data transfer between host and device can not be ignored.

Fig. 8 (b) shows that throwing in more CPU cores is not necessarily the optimal approach. The speed up ratio did not grow linearly with increasing CPU cores. However, when we increase the meshes to $80 \times 80 \times 160$, more iterations is needed for PPE solver and the latency can be hide ignored. Finally, we enlarge the meshes to $160 \times 160 \times 320$. The results are shown in Table II. It is about 7.76x faster for the present two phase flow problem. From the above tests, it is concluded that with one GPU card involved in calculation, high accuracy results can be obtained with much reduced computational time.

VI. Concluding remarks

A one-step conservative level set method for modelling the incompressible two-phase flow have been proposed. The mass conservation can be exactly satisfied by the proposed global mass correction method. Both of the proposed dispersion-relation-preserving dual-compact upwind advection scheme and multi-dimensional dispersion-relation-preserving upwind scheme have shown to be robust for the present two-phase solver. Also, benchmark problems with/without consideration of surface tension have been numerically investigated with the SMP parallelization. All the predicted results have been shown to compare fairly well with the benchmark, experimental and other numerical results.

By using 8 CPU cores and one GPU card, we can achieve 7.76x faster than the sequential program on a single computing node.

Acknowledgments

This first author thanks the National Center for High Performance Computing (NCHC) in HsinChu, Taiwan for the computing facilities and financial supports.

References

[1] Anderson C. R., A vortex method for flows with slight density variations, *J. Comput. Phys.*, 61 (1985) 417-444.
[2] Boultone-Stone J. M., Blake J. R., Gas bubbles bursting at a free surface, *J. Fluid Mech.*, 254 (1993) 437-466.
[3] Hirt C. W., Nichols B. D., Volume of fluid method (VOF) for the dynamics of free boundaries, *J. Comput. Phys.*, 39 (1981) 201-225.
[4] Unverdi S., Tryggvason G., A front-tracking method for viscous, incompressible, multi-fluid flows, *J. Comput. Phys.*, 100 (1992) 25-37.
[5] Sussman M., Smereka P., Axisymmetric free boundary problems, *J. Fluid Mech.*, 341 (1997) 269-294.
[6] Sethian J. A., Smereka P., Level set methods for fluid interfaces, *Annu. Rev. Fluid Mech.*, 35 (2003) 341-372.
[7] Sussman M., Puckett E. G., A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows, *J. Comput. Phys.*, 162 (2000) 301-337.
[8] Xiao F., Honma Y., Kono T., A simple algebraic interface capturing scheme using hyperbolic tangent function, *Int. J. Numer. Methods Fluid*, 48 (2005) 1023-1040.
[9] Yokoi K., Efficient implementation of THINC scheme: A simple and practical smoothed VOF algorithm, *J. Comput. Phys.* 226 (2007) 1985-2002.
[10] Sun D. L., Tao W. Q.. A coupled volume-of-fluid and level set (VOSET) method for computing incompressible two-phase flows, *Int. J. Heat Mass Tran.* 53 (2010) 645-655.
[11] Olsson E., Kreiss G., A conservative level set method for two phase flow, *J. Comput. Phys.*, 210 (2005) 225-246.
[12] Krüger J., Westermann R., Linear algebra operators for GPU implementation of numerical algorithms, ACM Trans. Graphics 22 (3) (2003), pp. 908-916.
[13] Goodnight N., Woolley C., Lewin G., Luebke D., Humphreys G., A multigrid solver for boundary value problems using programmable graphics hardware, Graphics Hardware (2003), pp. 1-11.
[14] Harris M. J., Fast fluid dynamics simulation on the GPU, GPU Gems (2004), pp. 637-665 (Chapter 38).
[15] Hagen T.R., Lie K.A., Natvig J.R., Solving the Euler equations on graphics processing units, Comput. Sci. ICCS 3994 (2006), pp. 220-227.
[16] Brandvik T., Pullan G., Acceleration of a two-dimensional Euler flow solver using commodity graphics hardware, Proc. Inst. Mech. Engineers, Pt C: J. Mech. Engrg. Sci. 221 (12) (2007), pp. 1745-1748.
[17] Corrigan A., Camelli F., Löhner R., Wallin J., Running unstructured grid based CFD solvers on modern graphics hardware, AIAA Paper 2009-4001, 19th AIAA Computational Fluid Dynamics, June 2009.
[18] Sheu T. W. H., Yu C. H., Chiu P. H., Development of a dispersively accurate conservative level set scheme for capturing interface in two-phase flows, *J. Comput. Phys.*, 228 (2009) 661-686.
[19] Brackbill J. U., Kothe D. B., Zemach C., A continuum method for modeling surface tension, *J. Comput. Phys.*, 100 (1992) 335-354.
[20] Tam C. K. W., Webb J. C., Dispersion-relation-preserving finite difference schemes for computational acoustics, *J. Comput. Phys.*, 107 (1993) 262-281.
[21] Chiu P. H., Sheu T. W.H., Lin R.K., Development of a dispersion-relation-preserving upwinding scheme for incompressible Navier-Stokes equations on non-staggered grids, *Numer. Heat Transf., B Fundam*, 48 (2005) 543-569.
[22] Chiu P. H., Sheu T. W. H., On the development of a dispersion-relation-preserving dual-compact upwind scheme for convection-diffusion equation, *J. Comput. Phys.*, 228 (2009) 3640-3655.
[23] M. Perić, R. Kessler and G. Scheuerer, Comparison of finite-volume numerical methods with staggered and colocated grids, *Comput. Fluids*, 16 (1988) 389-403.
[24] Golub G. H., Huang L. C., Simon H., Tang W. P., A fast Poisson solver for the finite difference solution of the incompressible Navier-Stokes equation, *SIAM J. Sci. Comput.*, 19 (1998) 1606-1624.
[25] Martin J. C., Moyce W. J., An experimental study of the collapse of fluid columns on a rigid horizontal plane, *Philos. Trans. Roy. Soc. Lond.: Ser. A*, 244 (1952), 312-324.
[26] Guermond J. L., Quartapelle L., A projection FEM for variable density incompressible flows, *J. Comput. Phys.*, 165 (2000) 167-188.
[27] Ding H., Spelt P. D. M., Shu C.. Diffuse interface model for incompressible two-phase flows with large density ratios, *J. Comput. Phys.* 226 (2007) 2078-2095.
[28] Kuo C. H., Hsieh C. W., Lin R. K., and Sheu W. H. , Solving Burgers's Equation Using Multithreading and GPU, (2010) LNCS 6082, pp. 297-307.

[29] Brereton G., Korotney D., Coaxial and oblique coalescence of two rising bubbles, In: Sahin, I., Tryggvason, G. (Eds.), Dynamics of Bubbles and Vortices Near a Free Surface, AMD-vol. 119, 1991, ASME, New York.

TABLE I
DETAILS OF COMPUTER HARDWARE USED TO RUN THE SIMULATIONS.

Hardware	Details	
CPU	Intel Xeon Processor X5472	
	Frequency of processor cores	3.0 GHz
	L2 cache size	12 MB
	# of Processor Cores	4
GPU	NVIDIA Tesla C1060	
	Frequency of processor cores	1.44 GHz
	RAM	4GB DDR3
	# of Streaming Processor Cores	240

TABLE II
TOTAL ELAPSED TIME AND SPEED UP IN THE PRESENT STUDY CASE.

Mesh size	Sequential	8 cores+GPU	Speedup
$160 \times 320 \times 160$	6.89 hr	0.89 hr	7.76

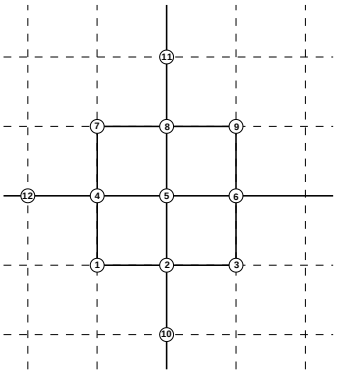
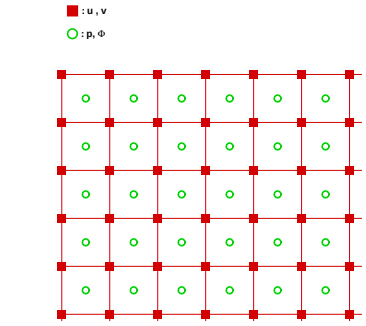
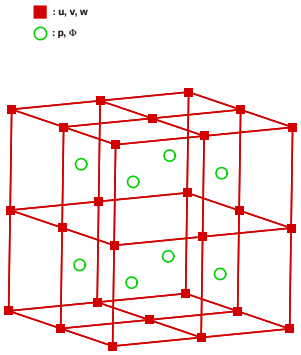


Fig. 1. Schematic of the stencil points invoked in the proposed two-dimensional DRP convection scheme.

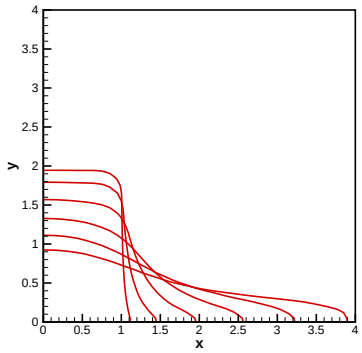


(a)

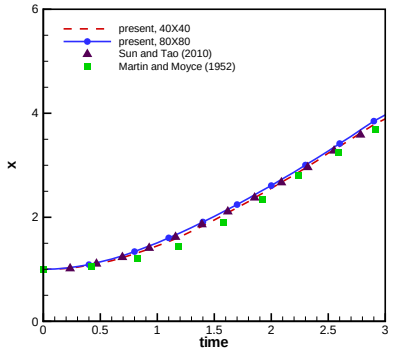


(b)

Fig. 2. Schematic for the present semi-stagger grid system. (a) two-dimensional; (b) three-dimensional.

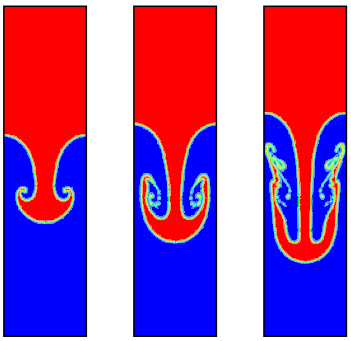


(a)

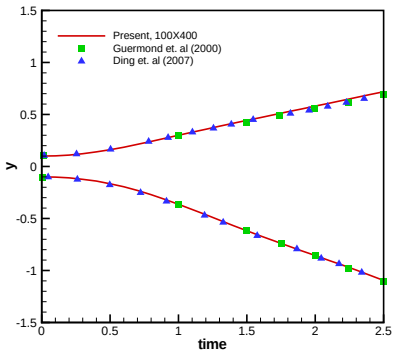


(b)

Fig. 3. Calculated results for the dam-break problem. (a)the time-history front profile; (b)Comparisons of the predicted front locations with the experimental data [25] and the numerical results [10].

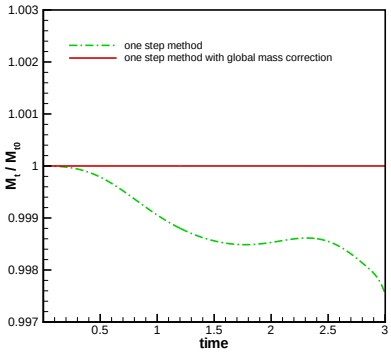


(a)

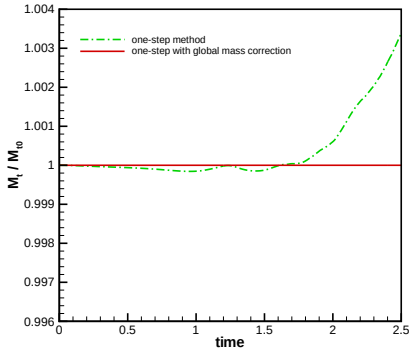


(b)

Fig. 4. The results for the Rayleigh-Taylor problem at the different time. (a) the interface profiles; (e) comparisons of the top of the rising fluid and the bottom of the falling fluid with the present results and the previous results.



(a)



(b)

Fig. 5. The predicted total mass ratios against time for the investigated problems. (a) the dam-break problem; (b) Rayleigh-Taylor instability problem. Note that $M = \int_\Omega \Phi d\Omega$ means the total mass and $M_0 = \int_\Omega \Phi|_{t=0} d\Omega$ means the total mass at $t = 0$.

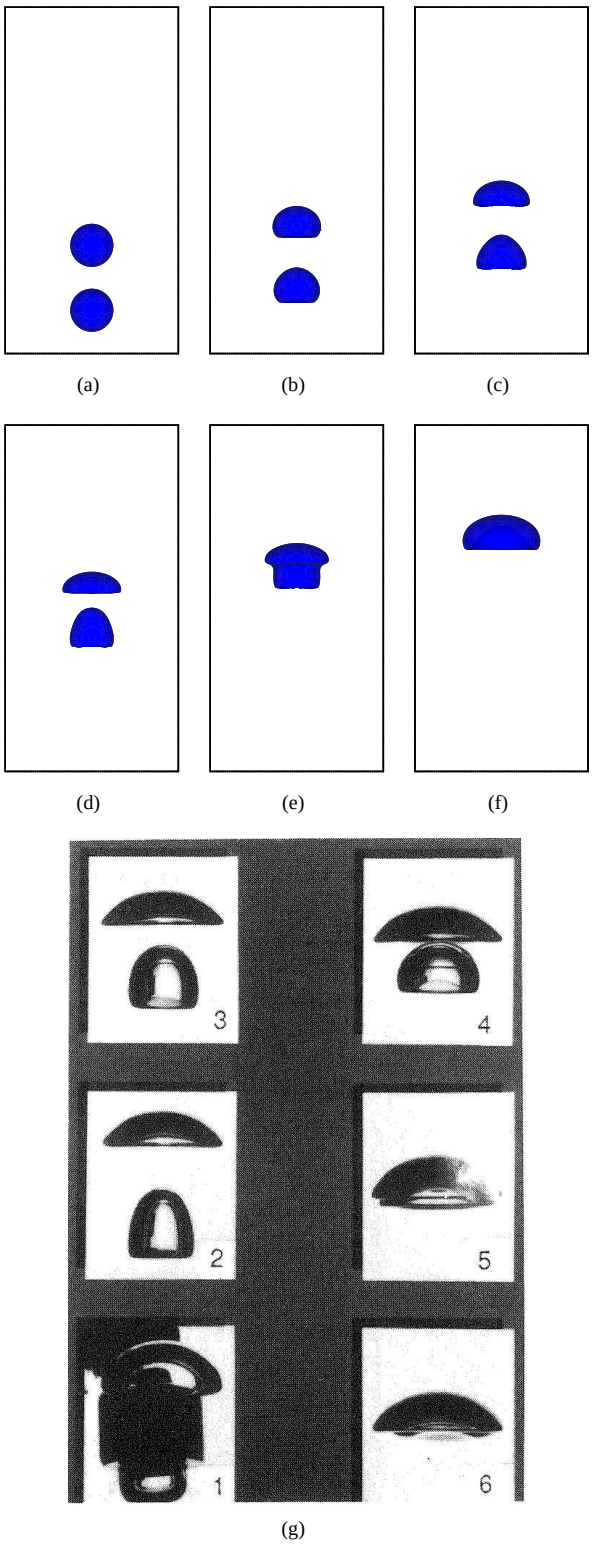


Fig. 6. Comparisons of the bubble shapes for the present method and experimental results (time difference between subsequent photographs is 0.03 s). (a) t = 0.0s; (b) t = 0.03s; (c) t = 0.06s; (d) t = 0.09s; (e) t = 0.12s; (f) t = 0.15s; (g) experimental results by Brereton and Korotney [29].

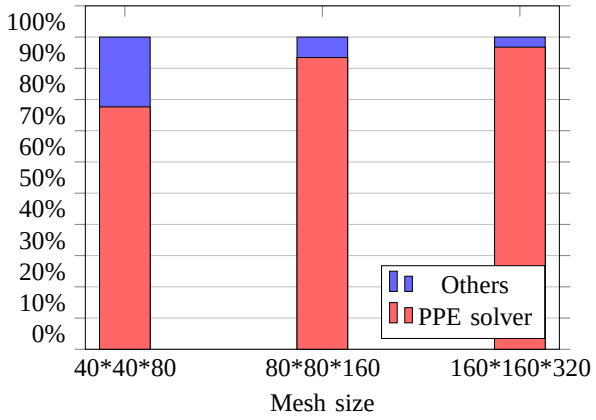


Fig. 7. PPE solver time spending as a percentage of total elapsed time.

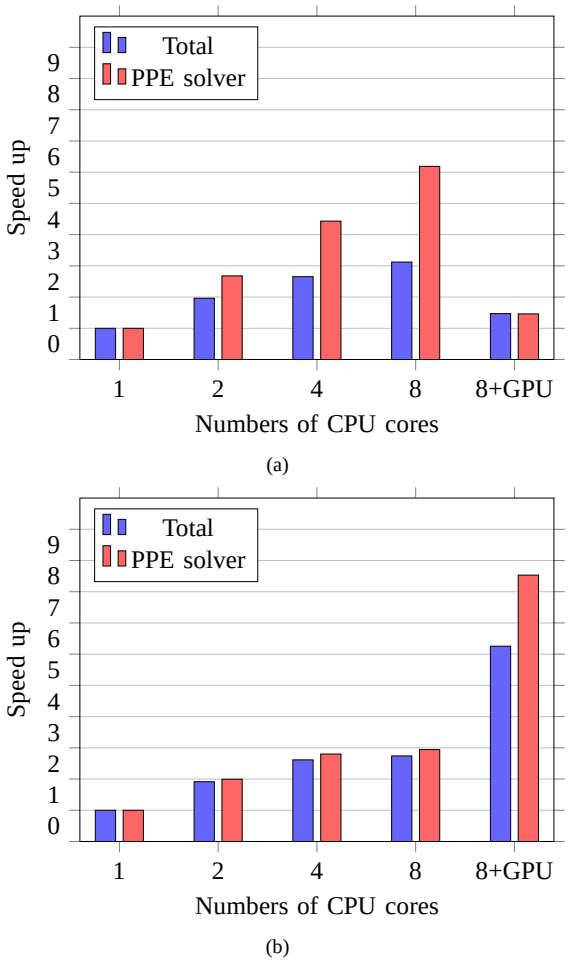


Fig. 8. Speed up ratio for the present study with different mesh size. (a)$40 \times 40 \times 80$; (b)$80 \times 80 \times 160$. Note that '8+GPU' means that we solve the problem with 8 CPU cores and one GPU card.

Sheng-Hsiu Kuo Sheng-Hsiu Kuo received his Master Degree in Engineering Science and Ocean Engineering from National Taiwan University in 2008. He is presently a Assistant Research Scientist in National Center for High-Performance, Taiwan. His research interest includes Computational Fluid Dynamics, High Performance Computing and Parallel Programming.

Pao-Hsiung Chiu Pao-Hsiung Chiu is a associate researcher of Nuclear Engineering Division, Institute of Nuclear Energy Research, Longtan, Taiwan, ROC. He obtained his PhD from the National Taiwan University, Taiwan, ROC at 2009. His domain of interest is: computational fluid dynamics, two-phase flows and applied mathematics.

Reui-Kuo Lin Reui-Kuo Lin is a associate researcher of Taiwan Typhoon and Flood Research Institute, Taipei, Taiwan, ROC. He obtained his PhD from the National Taiwan University, Taiwan, ROC at 2005. His domain of interest is: computational fluid dynamics, and applied mathematics.

Yan-Ting Lin Yan-Ting Lin is a associate researcher of Nuclear Engineering Division, Institute of Nuclear Energy Research, Longtan, Taiwan, ROC. He obtained his PhD from the National Cheng Kung University, Taiwan, ROC at 2009. His domain of interest is: heat transfer and computational fluid dynamics.