INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **48**:1023–1040
Published online 7 April 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.975

A simple algebraic interface capturing scheme using hyperbolic tangent function

F. Xiao[1,2,*,†], Y. Honma[1] and T. Kono[1]

[1]*Department of Energy Sciences, Tokyo Institute of Technology, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan*
[2]*State key Laboratory of Severe Weather, Chinese Academy of Meteorological Sciences, 46 Zhongguancun South Street, Beijing 100081, China*

SUMMARY

This paper presents a simple and practical scheme for capturing moving interfaces or free boundaries in multi-fluid simulations. The scheme, which is called THINC (tangent of hyperbola for interface capturing), makes use of the hyperbolic tangent function to compute the numerical flux for the fluid fraction function, and gives a conservative, oscillation-less and smearing-less solution to the fluid fraction function even for the extremely distorted interfaces of arbitrary complexity. The numerical results from the THINC scheme possess adequate quality for practical applications, which make the extra geometric reconstruction, such as those in most of the volume of fluid (VOF) methods unnecessary. Thus the scheme is quite simple. The numerical tests show that the THINC scheme has competitive accuracy compared to most exiting methods. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: computational fluid dynamics; moving interface; interface capturing; advection scheme; multi-phase flow

1. INTRODUCTION

Computation of moving interface or free boundary is of great importance in the direct simulations of multi-phase fluid dynamics, where the interface separating different fluids and moving with the flow field needs to be explicitly computed. In contrast to the so-called front tracking methods where Lagrangian particles are used [1, 2], the front capturing methods, such as the volume of fluid (VOF) and the level set methods use time and space-dependent indication functions to identify the interfaces.

VOF has been used to name a wide spectrum of methods that employ an indication function to define the volume fraction of a certain fluid for each mesh volume. The VOF-type methods have so far got a popularity in multi-phase fluid simulations because of their conservative

*Correspondence to: F. Xiao, Department of Energy Sciences, Tokyo Institute of Technology, 4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan.
†E-mail: xiao@es.titech.ac.jp

Received 20 December 2004
Revised 4 March 2005
Accepted 5 March 2005

Copyright © 2005 John Wiley & Sons, Ltd.

nature. The key in a VOF scheme is how to compute the numerical fluxes to update the function of fluid fraction. In the original VOF method of Hirt and Nichols [3], a donor–acceptor formulation is used with some flux-limiting manipulations to assure the boundedness of the numerical solution. Other successive researches along this direction are found in References [4, 5]. These methods do not require explicitly the geometrical reconstructions in the solution procedure, thus are referred to as algebraic-type methods. Another category of the VOF methods, the so-called geometrical type, make them different from the algebraic ones by adding an extra step after the advection to explicitly identify the geometry of the moving interface. Principally, a conventional geometrical VOF algorithm consists of two steps, i.e. (1) carrying forward the interface according to fluid velocity (advection) and (2) identifying the interface at new position (geometrical reconstruction). The early geometrical VOF scheme is known as SLIC (simple line interface calculation) [6] where the reconstruction is conducted by a straight line parallel to coordinate axis. More sophisticated methods, known as PLIC (piecewise linear interface calculation), have been proposed to use a linear approximation with the orientation of the interface considered [7–11]. A PLIC scheme is proved to be more accurate than a SLIC one, but a significant increase in the algorithmic complexity at the geometrical reconstruction step seems to be unavoidable.

Considering that the VOF function is nothing more than a step-like function that moves with the local flow velocity, it is quite natural for one to think of using a high-resolution advection scheme to compute the VOF function. Some efforts have been made in this direction [12–14]. In these schemes, numerical diffusions and oscillations across the interface jump are effectively suppressed and no geometrical reconstruction is involved. Without the geometrical reconstruction, these methods may also be more appropriately called ‘algebraic VOF’ or algebraic interface capturing methods. Numerical results show that the algebraic interface capturing methods using high resolution advection schemes are more accurate than the SLIC method and the original VOF method, but inferior to most of the PLIC methods in the presence of complex flow fields. Nevertheless, because an algebraic interface capturing method is much simpler and more computationally efficient than a PLIC VOF scheme, new schemes of this category should be worthy of further investigation.

The desired numerical solution of the VOF function (or more properly, the density function in the context of an algebraic interface capturing scheme) should at least be conservative, free from spurious oscillation and numerically diffusion less. It is obvious that the advection scheme used to transport the fluid fraction function or VOF function substantially affects the numerical solution of the interface. With a well-designed flux limiter or slope limiter, a conservative Eulerian high-resolution scheme is able to effectively prevent the numerical oscillation. However, any scheme of this type does smear the initial jump in the density function due to the inherent numerical diffusions. Most of the algebraic methods are devised by incorporating an artificial compression or anti-diffusion manipulation to the high-resolution schemes [12–15].

On the other hand, researches that make use of special analytical functions to improve the numerical solutions of difference equations are also found in the literature. The exponential function is the mostly used one in solving advection–diffusion equation since the solution to the 1D advection–diffusion equation is in the form of the exponential function. In regions where the solution has an exponential distribution, the finite difference approximation which is equivalent to the polynomial fitting cannot give sufficient accuracy. Fitting the solution with an exponential and deriving the discretized equation by the constraint conditions imposed at

Copyright © 2005 John Wiley & Sons, Ltd.

grid nodes appear to be effective in getting more accurate solutions to the problems involving advection–diffusion equation. The applications of the so-called exponentially fitted method are found in the boundary layer problems of viscous fluid [16, 17] and the transport problem in silicon device [18]. A recent review with the new progress of the exponentially fitted method is reported in Reference [19]. In the interface capturing computations, a VOF function has step-like distributions across the moving interface, which tends to cause significant numerical errors if conventional finite difference or finite volume methods, as the most Eulerian schemes mentioned above, are used. So, fitting the step-like solution with some particular functions may be used as well to improve the numerical representation of the VOF function. In our previous work [20], a tangent transformation is incorporated to a high-order advection scheme [21] to improve the step jump in the density function. The resulting scheme, however, is not numerically conservative.

In this paper, we develop a new interface capturing scheme, namely THINC (tangent of hyperbola for interface capturing), by using the hyperbolic tangent function. The step-like nature of the hyperbolic tangent function makes it a suitable interpolant for the flux computation of a VOF function, and is effective in eliminating numerical diffusion and oscillation. The hyperbolic tangent interpolation function is constructed based only on the value of fluid fraction which is between 0 and 1. The jump and slope of the hyperbolic tangent function, which play a key role in getting a geometrically faithful solution, are automatically determined in the THINC scheme depending on the local solution. Our numerical experiments show that the THINC scheme possesses adequate accuracy in both pure advection and fluid application, and has a more robust performance in keeping the compact thickness of largely distorted interface compared to other methods using high-resolution schemes [12–15]. Without geometrical reconstruction, the THINC scheme is quite simple and computationally efficient.

In Section 2, the algebraic interface capturing scheme THINC is presented. The numerical experiments of pure advection of the moving interface are given in Section 3 with comparisons to other existing numerical methods. Some validations and applications of the THINC scheme to multi-fluid simulations are presented in Section 4, and the paper ends with a short summary in Section 5.

2. THE THINC SCHEME

The VOF function evolves according to the following advection equation:

$$\frac{\partial f}{\partial t} + \nabla \cdot (\mathbf{u} f) - f \nabla \cdot \mathbf{u} = 0 \tag{1}$$

where $\mathbf{u}$ is the velocity field. The VOF function f has a value between 0 and 1.

We want our advection scheme to meet at least the three requirements: (1) exactly conserves the advected quantity, (2) effectively eliminates the numerical smearing (diffusion) and (3) does not produce spurious oscillation around the step-jump of the VOF function. The straightforward way to achieve these properties is to employ a formulation of flux form and use artificial compression or anti-diffusion which may be inherently built in the scheme through the interpolation reconstruction for flux computation [13, 14] or by extra treatments

Copyright © 2005 John Wiley & Sons, Ltd.

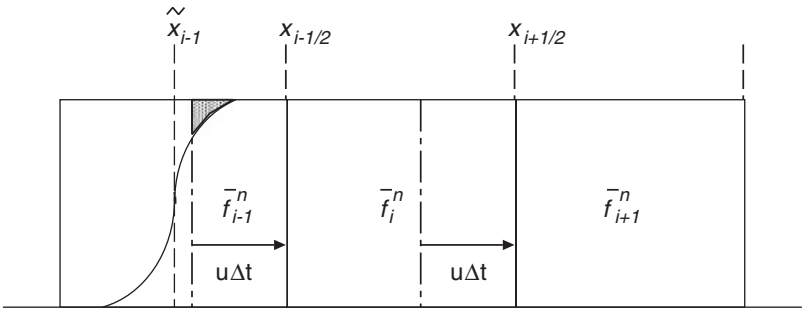

Figure 1. The one-dimensional VOF function. The interface is identified by the step-jump falling in a cell where the VOF function $\bar{f}_{i-1}$ has a value between 0 and 1. $\tilde{x}_{i-1}$ indicates the position of the jump. The hyperbolic tangent function is used to compute the numerical fluxes. The shaded area indicates the numerical diffusion and is determined by parameter β.

separating from the advection [15, 12]. However, for a step-like function with values between 0 and 1, the construction of the advection scheme can be significantly simplified by use of some special interpolation functions. The THINC scheme presented below makes use of the hyperbolic tangent function to evaluate the numerical flux.

The basic 1D THINC scheme is devised for the advection equation of the VOF function

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}(uf) - f\frac{\partial u}{\partial x} = 0 \tag{2}$$

where t refers to the time, x the spatial coordinate, u the advection speed and f the transported quantity.

The VOF function has its solution bounded by 0 and 1. A moving interface in one dimension is represented by a jump in the VOF function as shown in Figure 1. The Lagrangian invariant solution of (2) states that a jump in the initial condition remains unchanged along the trajectory. However, it is known that any straightforward use of finite difference method or finite volume method will produce significant numerical errors, such as the numerical diffusion or the spurious oscillation because of the poor accuracy of a polynomial-based fitting to a discontinuous jump. Analogous to the exponential fitted method used for the advection–diffusion equation, we use the hyperbolic tangent function to fit the jump in the VOF function and to get less diffused and more accurate numerical solutions.

Let $\bar{f}_i^n$ denote the cell-averaged value of the numerical solution to (2), which is defined over $[x_{i-(1/2)}, x_{i+(1/2)}]$ at the nth time step $(t = t^n)$ by

$$\bar{f}_i^n = \frac{1}{\Delta x_i}\int_{x_{i-(1/2)}}^{x_{i+(1/2)}} f(x, t^n)\,\mathrm{d}x \tag{3}$$

where $\Delta x_i = x_{i+(1/2)} - x_{i-(1/2)}$.

It is obvious that the hyperbolic tangent is the simplest continuous function that well fits a step-jump distribution. We use the piecewise modified hyperbolic tangent function as

$$F_i(x) = \frac{\alpha}{2}\left(1 + \gamma\,\tanh\left(\beta\left(\frac{x - x_{i-(1/2)}}{\Delta x_i} - \tilde{x}_i\right)\right)\right) \tag{4}$$

Copyright © 2005 John Wiley & Sons, Ltd.

The parameters α, β and γ are of substantial importance in determining the quality of the numerical solution, and will be discussed later.

Given α, β and γ, the only unknown left in (4) is the middle point of the transition jump in the hyperbolic tangent function $\tilde{x}_i$, which is computed from the cell-integrated average $\bar{f}_i^n$ as follows:

$$\frac{1}{\Delta x_i}\int_{x_{i-(1/2)}}^{x_{i+(1/2)}} F_i(x)\,\mathrm{d}x = \bar{f}_i^n \tag{5}$$

After the piecewise interpolation functions $F_i(x)$ have been computed for all mesh cells, the VOF function f is updated by the formulation of flux form as

$$\bar{f}_i^{n+1} = \bar{f}_i^n - (g_{i+(1/2)} - g_{i-(1/2)})/\Delta x_i + \bar{f}_i^n(u_{i+(1/2)} - u_{i-(1/2)})/\Delta x_i \tag{6}$$

where $g_{i+(1/2)}$ denotes the flux across boundary $x = x_{i+(1/2)}$ during $t^{n+1} - t^n$ and is computed as

$$g_{i+(1/2)} = \begin{cases} -\displaystyle\int_{t^n}^{t^{n+1}} F_i(x_{i+(1/2)} - u_{i+(1/2)}(t - t^n))\,\mathrm{d}t & \text{if } u_{i+(1/2)} \geqslant 0 \\ \displaystyle\int_{t^n}^{t^{n+1}} F_{i+1}(x_{i+(1/2)} - u_{i+(1/2)}(t - t^n))\,\mathrm{d}t & \text{otherwise} \end{cases} \tag{7}$$

The way to determine parameters α, β and γ is as follows.

Parameter γ depends on the slope orientation of the jump, and is determined as

$$\gamma = \begin{cases} 1 & \text{if } \bar{f}_{i+1}^n \geqslant \bar{f}_{i-1}^n \\ -1 & \text{otherwise} \end{cases} \tag{8}$$

It easy to show that (8) guarantees the reconstruction (4) to have a monotonicity consistent with the spatial distribution of the VOF function.

In order to have the interpolation function $F_i(x)$ bounded between $\bar{f}_{i-1}^n$ and $\bar{f}_{i+1}^n$, parameter α is determined as

$$\alpha = \begin{cases} \bar{f}_{i+1}^n & \text{if } \bar{f}_{i+1}^n \geqslant \bar{f}_{i-1}^n \\ \bar{f}_{i-1}^n & \text{otherwise} \end{cases} \tag{9}$$

From (9), it is also obvious that the jump in the interpolation function for cell i is determined according to the larger value of the VOF function in the neighboring cells. Hence, an inherent anti-diffusion effect is included.

Parameter β determines the steepness of the jump in the interpolation function. To show how β affects the numerical diffusion, we consider the case of a single jump located in cell $i-1$ shown in Figure 1. Without losing generality, we assume a constant advection velocity $(u > 0)$ and $\bar{f}_{i+1}^n = \bar{f}_i^n > \bar{f}_{i-1}^n$. For the case of $u\Delta t < x_{i-(1/2)} - \tilde{x}_{i-1}$, the numerical fluxes at the two ends of cell i should be identical and equal to

$$g_{i-(1/2)} = g_{i+(1/2)} = \bar{f}_i^n \times (u\Delta t) \tag{10}$$

if the jump is perfect and without any smearing. Thus, $\bar{f}_i^{n+1}$ does not suffer any numerical dissipation and should remain exactly the same as $\bar{f}_i^n$ according to (6). Consequently, the

Copyright © 2005 John Wiley & Sons, Ltd.

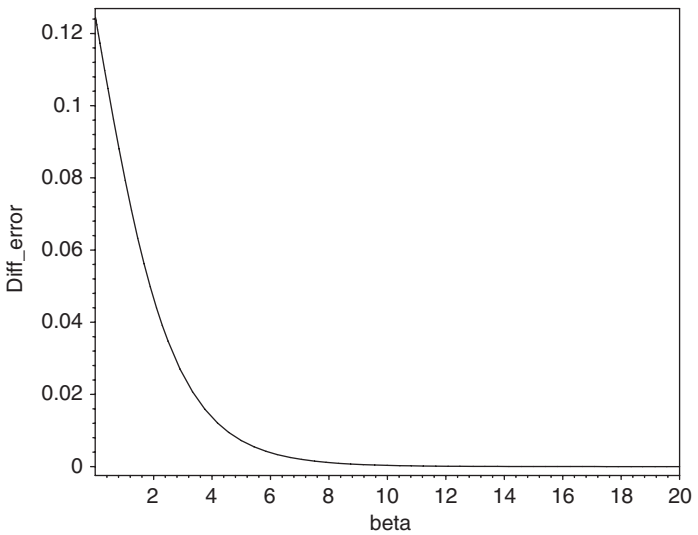

Figure 2. The relation between the numerical dissipation and parameter β. As β increases, the jump in the hyperbolic tangent function becomes steeper and the numerical diffusion is reduced.

numerical diffusion to $\bar{f}_i^{n+1}$ because of using the reconstruction (4) can be measured by

$$\mathrm{Diff}_{\mathrm{error}} \equiv \frac{|\bar{f}_i^n \times (u\Delta t) - \int_{x_{i-(1/2)}-u\Delta t}^{x_{i-(1/2)}} F_{i-1}(x)\,\mathrm{d}x|}{|\bar{f}_i^n|} \tag{11}$$

Figure 2 shows the dependency of the numerical diffusion on parameter β. It is obvious that a larger β leads to a steeper jump in the interpolation reconstruction and thus a less numerical dissipation. However, our numerical experiments show that a large β gives a steep interface jump but tends to wrinkle an interface which is parallel to the velocity direction, thus makes the scheme more similar to a 'downwind' donor–acceptor scheme. It is also found that the THINC scheme is able to provide a steep but less wrinkled interface with a proper choice of β. We used an ad hoc value of $\beta = 3.5$ in the all numerical runs in this paper.

As observed above, the THINC algorithm, with the numerical fluxes evaluated algebraically, is extremely simple. From the hyperbolic tangent function (4), we know that the interpolation function used in the THINC scheme is free from oscillation. The numerical diffusion, which usually stems from the interpolation reconstruction in an Eulerian high-resolution scheme, can be effectively eliminated by properly choosing α and β.

The THINC method has some features similar to the existing schemes which are based partly on the donor–acceptor formulation, such as the original VOF [3] and the SURFER [4] methods. In the latter, the numerical fluxes are computed by a piecewise constant approximation of the fluid fraction function. The integrated average value for each cell is calculated by either the smearing 'upwind' scheme or the steepening 'downwind' scheme. As discussed in Reference [4], a 'downwind' scheme can effectively keep the interface jump from being smeared, but tends to wrinkle the moving boundary that is nearly parallel to the flow di-

Copyright © 2005 John Wiley & Sons, Ltd.

rection. A remedy to this can be a switching between a 'upwind' scheme and a 'downwind' scheme according to how the interface is perpendicular to the flow. The step-like feature of the hyperbolic tangent function leads to a somewhat piecewisely constant approximation. However, the THINC scheme proposed in this paper is substantially different from the existing schemes in the following aspects: (1) using the hyperbolic tangent, the THINC scheme automatically preserves the boundedness of the VOF function, thus does not need any extra manipulation required in the donor–acceptor formulation to eliminate the over-flow and under-flow in the numerical solution, (2) as shown later, the THINC method gives adequate numerical results even when there is no upwind–downwind switching involved, (3) shown in the numerical tests given next, the THINC scheme does not produce flotsams that always associate with other donor–acceptor formulations.

A 1D advection of a square pulse is computed. The numerical result after 5000 steps of updating with a CFL number of 0.25 is plotted in Figure 3. The step-like jump in the solution is 30 constantly preserved and well resolved.

In the current THINC scheme, multi-dimensional computation is conducted by operator splitting. The one-dimensional building block discussed above is separately used in each coordinate direction. This makes the THINC scheme straightforward and simple in multi-dimensional implementation. Being of an algebraic type, the THINC method does not involve any geometrical reconstruction in multi-dimensional computation, but appears to be able to give adequately accurate numerical results as shown next.

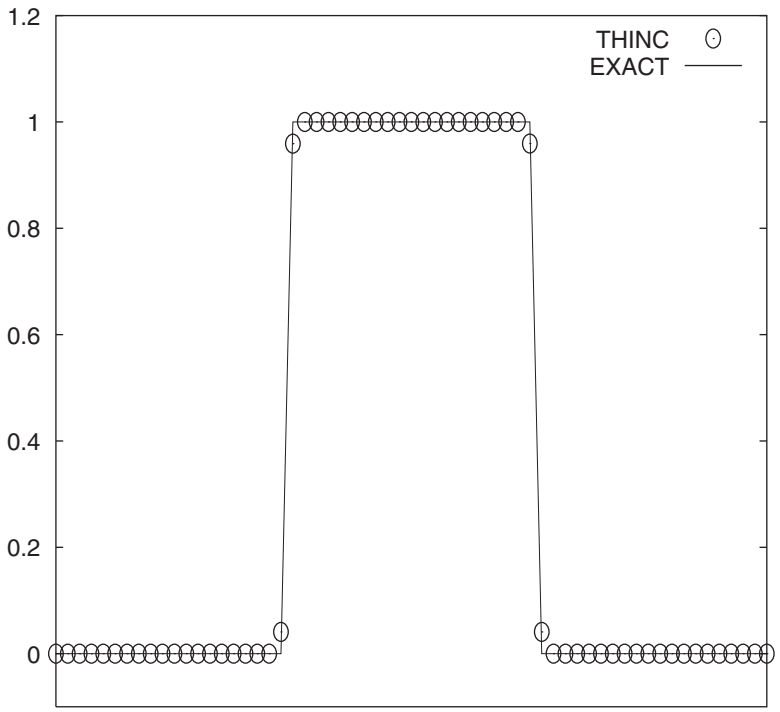

Figure 3. A 1D transport of a square wave after 5000 steps.

Copyright © 2005 John Wiley & Sons, Ltd.

3. ADVECTION TESTS

To see how the THINC scheme performs in tracking interfaces with different orientations against the flow direction, we conducted the numerical test shown as Figure 4 in [4]. An initial square shown in Figure 4 (top left) of size 0.125×0.125 is centred on a 1×1 computational domain partitioned by a 64×64 grid. Three velocity fields used in the advection tests are, respectively, defined as $(u, v) = (-1/10, -1); (-1/2, -1); (-1, -1)$. The advection computations were carried forward for 160 steps with a CFL number of 0.15. The numerical results are given in the rest of the panels of Figure 4. The square is accurately transported with all

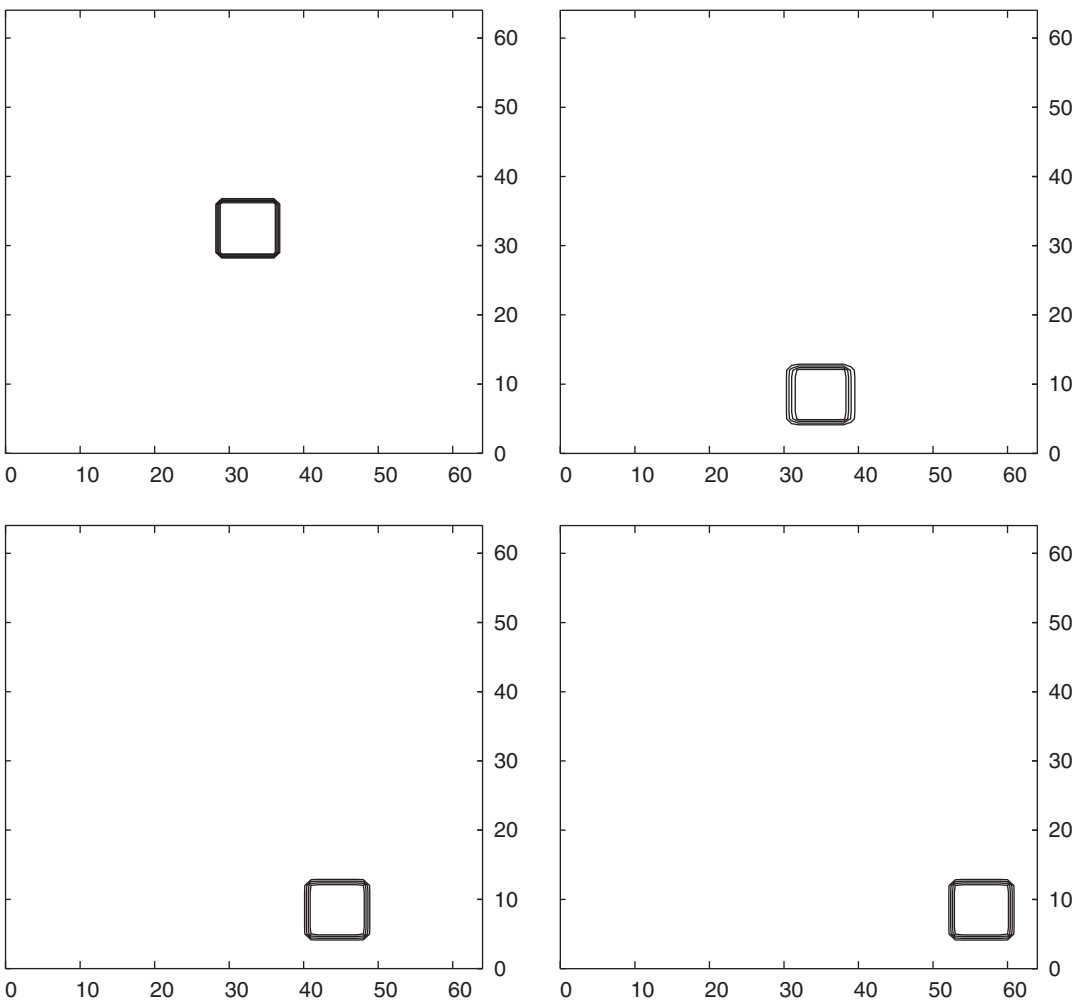

Figure 4. Translation of a 2D square initially centered in the computational domain (top-left) with different velocity fields: $(u, v) = (-1/10, -1)$ (top-right), $(u, v) = (-1/2, -1)$ (bottom-left) and $(u, v) = (-1, -1)$ (bottom-right). Plotted are the contours of 0.05, 0.4, 0.6 and 0.95 as in Reference [4].

Copyright © 2005 John Wiley & Sons, Ltd.

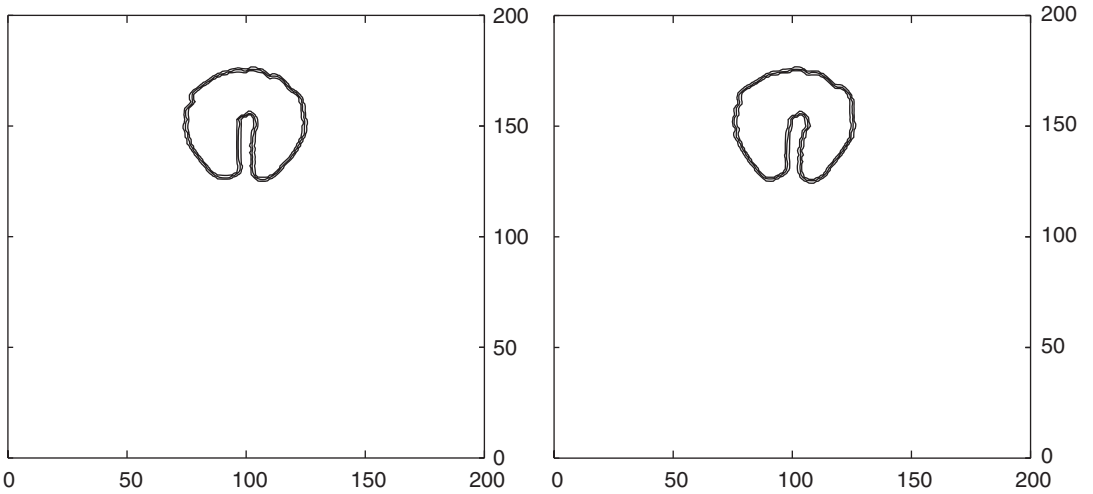

Figure 5. Zalesak's test after one (left) and two (right) revolutions of rotation. Plotted are the contours of 0.05, 0.5 and 0.95.

velocity fields. The numerical solutions are free from numerical diffusion and spurious oscillation, and are geometrically faithful. The distortions and the flotsams observed in Reference [4] are not found here.

The solid rotation test of a notched cylinder of Rudman [13] is computed. The results after one and two revolutions are shown in Figure 5. Some distortions in the shape of the cylinder are observed, but the compactness of the interface jump is preserved. Compared to the one-revolution output, the two-revolution solution is well resolved as well without remarkable degradation.

A practical interface capturing method for multi-fluid simulations should be tested and verified with not only the translational and rotational velocity field but also with highly deforming fluid flow. We tested THINC scheme with the 2D velocity field [13] defined as

$$\mathbf{u} = (\sin(x)\cos(y), -\cos(x)\sin(y)) \tag{12}$$

The computational domain $[0,\pi]\times[0,\pi]$ is equally divided into a 100×100 mesh. The initial volume fraction distribution is a circle centred at $(\pi/2,(\pi+1)/5)$ with a radius of $\pi/5$. As in Reference [13], the initial volume fraction is predicted with the velocity field (12) for N steps and then transported back with a reversed velocities for another N steps. As expected, the flow field defined above leads to the stretching and spiraling of the initial shape, which is then characterized by a thin film tail that becomes unresolvable by the finite resolution of the fixed mesh when N is large. To compare with the results in Reference [13], we plotted the results for $N=1000$ and $N=2000$ in Figures 6 and 7.

The compact thickness of the interface jump is effectively preserved. The distorted spiral shapes and the reversed circles were reproduced with a reasonable accuracy. From the bird's eye view of Figures 6 and 7 for $N=1000$ and $N=2000$, it is observed that the THINC scheme maintains satisfactorily the step-like distributions in the restored solutions even for a highly deformed interface. The L_1 numerical errors defined in Reference [13] are given in

Copyright © 2005 John Wiley & Sons, Ltd.

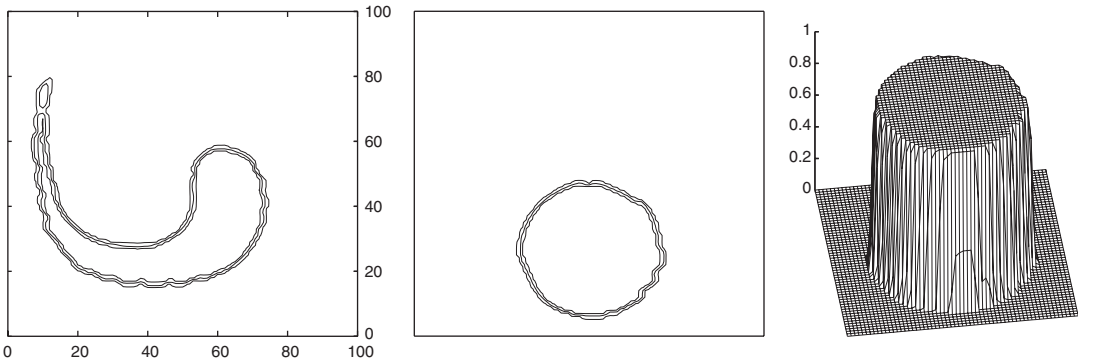

Figure 6. Single-vortex shearing flow test on a 100×100 mesh with $N = 1000$. The velocity field reverses at $N = 1000$, and restores the configuration back to its initial state after 2000 steps. Displayed are the contours of 0.05, 0.5 and 0.95 of the numerical results at 1000 steps (left) and 2000 steps (middle), and a bird's eye view of the result at 2000 steps.

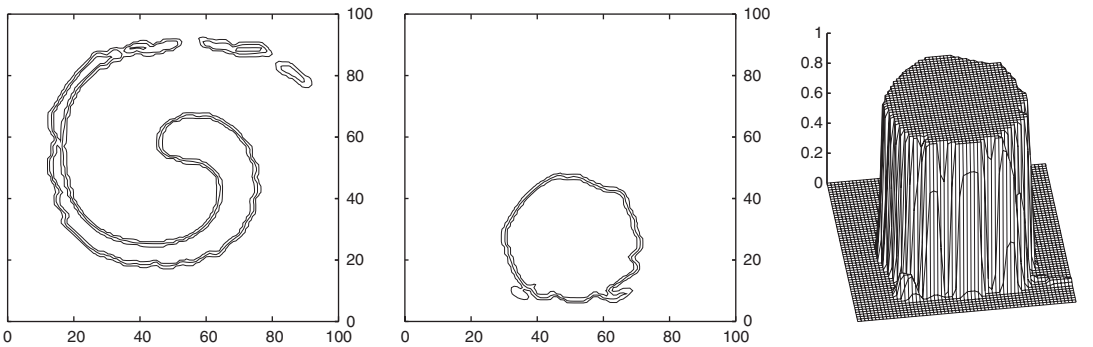

Figure 7. Single-vortex shearing flow test on a 100×100 mesh with $N = 1000$. The velocity field reverses at $N = 2000$, and restores the configuration back to its initial state after 4000 steps. Displayed are the contours of 0.05, 0.5 and 0.95 of the numerical results at 2000 steps (left) and 4000 steps (middle), and a bird's eye view of the result at 4000 steps.

Table I for different numbers of stepping N. Compared with Table 3 in Reference [13], we find that the numerical error is less dependent on the stepping number N in the results of THINC scheme. For $N \geqslant 1000$, the THINC scheme is obviously superior to the SLIC [6], the original VOF [3] and the FCT-VOF [13]. Compared to other algebraic type methods based on high resolution schemes, the THINC method is more effective in maintaining the sharp step-jump in complex flows.

As seen in (6), THINC scheme is of a flux-based form. Thus, the VOF function $\bar{f}$ is conserved in a divergence-free velocity field. We evaluated the conservativeness of the scheme by examining the change in the total mass of $\bar{f}$ as

$$\text{Mass}_{\text{error}} = \frac{\sum \bar{f}_{ij}^{n} - \sum \bar{f}_{ij}^{0}}{\sum \bar{f}_{ij}^{0}} \tag{13}$$

Copyright © 2005 John Wiley & Sons, Ltd.

Table I. L_1 errors for shearing flow test.

N	250	500	1000	2000
Error	3.21×10^{-2}	3.53×10^{-2}	3.84×10^{-2}	6.81×10^{-2}

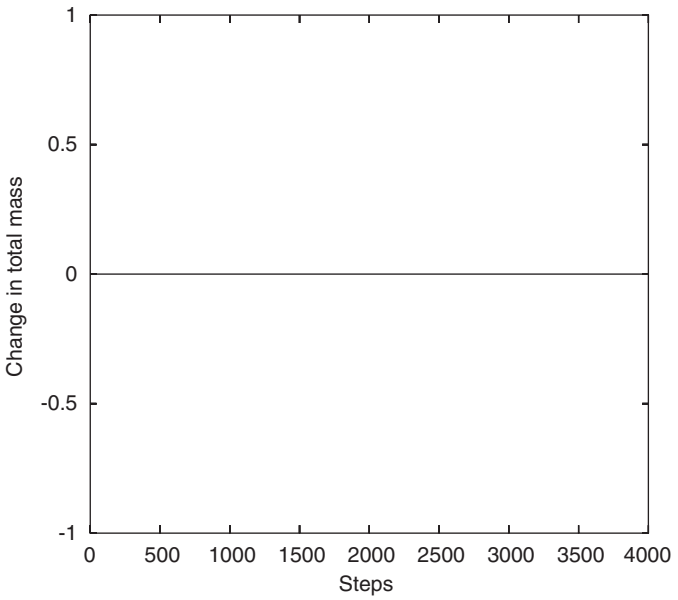

Figure 8. Relative change of the total mass of the VOF function in the single-vortex shearing flow test with $N = 2000$. The relative variation in the total mass of the VOF function, as indicated by the solid line, is computed by $\text{Mass}_{\text{error}} = (\sum \tilde{f}_{ij}^{n} - \sum \tilde{f}_{ij}^{0}) / \sum \tilde{f}_{ij}^{0}$, with $\sum \tilde{f}_{ij}^{0}$ and $\sum \tilde{f}_{ij}^{n}$ being the total mass of the initial VOF function and that of the nth time step, respectively.

where $\sum \tilde{f}_{ij}^{0}$ and $\sum \tilde{f}_{ij}^{n}$ are the total mass of the initial VOF function and that of the nth time step, respectively.

The mass change defined by (13) for the case of $N = 2000$ is shown in Figure 8. The total mass has been exactly conserved even the interface experienced very heavy distortion in this numerical test.

Another more stringent test is to capture an interface transported by a velocity field defined in a stream function by

$$\Psi = \frac{1}{4\pi} \sin\left(4\pi\left(x + \frac{1}{2}\right)\right) \cos\left(4\pi\left(x + \frac{1}{2}\right)\right) \cos(\pi t/T) \tag{14}$$

This velocity field is characterized by multi-vortex array and was used by Rider and Kothe [22] for testing interface tracking methods. As can be seen in Figure 3 of Reference [22], the initial sphere was largely deformed into several segments which appear topologically diverse, such as spiral, thin film and thin bridge. The thin segments usually appear not to be resolvable for most Eulerian interface tracking methods.

Copyright © 2005 John Wiley & Sons, Ltd.

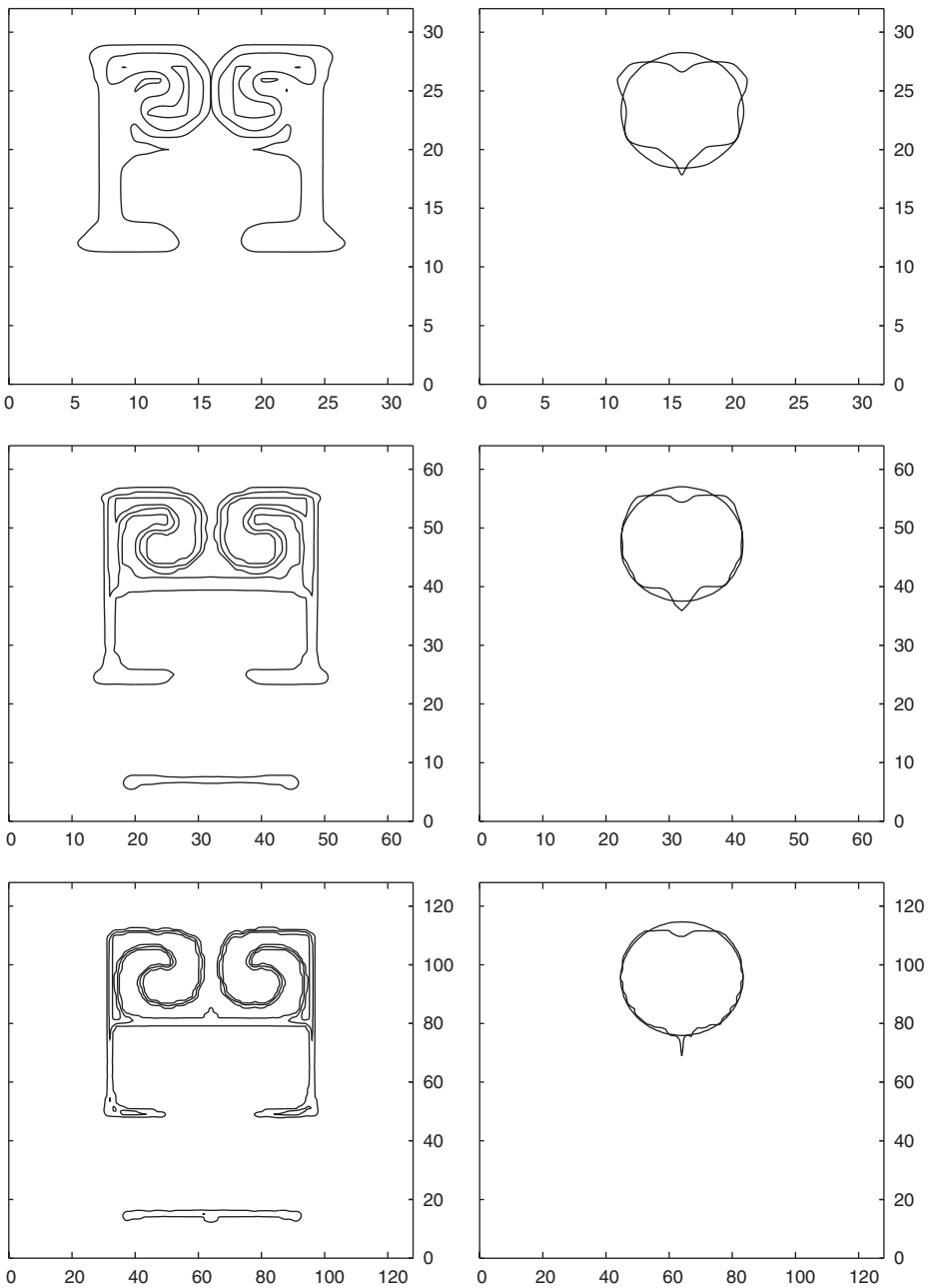

Figure 9. Multi-vortex deformational flow test on a 32×32 (top), 64×64 (middle) and 128×128 (bottom) meshes with $T = 2$. The velocity field reverses at $t = T/2$, and restores the configuration back to its initial state at $t = T$. Displayed are the contours of 0.05, 0.5 and 0.95 of the numerical results at $t = 1$ (left) and the final interfaces at $t = 2$ against the exact solution (right).

Copyright © 2005 John Wiley & Sons, Ltd.

Table II. L_1 errors and convergence rates of THINC on the multi-vortex deformation problem.

32×32	Order	64×64	Order	128×128
2.60×10^{-2}	0.913	1.38×10^{-2}	0.576	9.26×10^{-3}

A 2D cylinder as defined in Reference [22] was transported over a 1×1 square with the velocity field (14). The time required to move the distribution back to its initial state is $T = 2$. A periodical lateral boundary condition is imposed in this computation. Figure 9 shows the numerical results on 32×32, 64×64 and 128×128 meshes, respectively. Again, the steepness of the jump is effectively preserved, and the thickness of the transition layer remains compact. The symmetry of the solution is well retained. The right side panels in Figure 9 show the final interface against the exact solution. The main part of the body has been satisfactorily restored, even with coarse computational grids.

The L_1 errors, same as that used in Reference [22], are given in Table II. Compared with Table 4 in Reference [22], the THINC shows a capability competitive to other existing methods in capturing interfaces in complex flow. The convergence rates for this particular benchmark test, however, one below one.

4. MULTI-FLUID SIMULATIONS

The THINC scheme has been also validated in realistic simulations of multi-fluid dynamics.

We consider the following governing equations for incompressible multi-fluid flows in volume Ω with its surface denoted by Γ:

$$\int_\Gamma \mathbf{u} \cdot \mathbf{n} \, \mathrm{d}S = 0 \tag{15}$$

$$\frac{\partial}{\partial t} \int_\Omega \mathbf{u} \, \mathrm{d}V + \int_\Gamma \mathbf{u}(\mathbf{u} \cdot \mathbf{n}) \, \mathrm{d}S = -\frac{1}{\rho} \int_\Gamma p\mathbf{n} \, \mathrm{d}S + \frac{1}{\rho} \int_\Gamma 2\mu \mathbf{D} \cdot \mathbf{n} \, \mathrm{d}S + \frac{1}{\rho} \int_\Gamma \sigma\kappa\delta(\mathbf{X}_I)\mathbf{n} \, \mathrm{d}S + \mathbf{g} \tag{16}$$

where $\mathbf{u} = (u, v, w)$ is the velocity vector, $\mathbf{n}$ the outgoing normal, ρ the density, p the pressure, μ the viscosity, $\mathbf{D}$ the velocity strain tensor, σ the coefficient of surface tension, κ curvature, and $\mathbf{X}_I$ the point on the free interface and $\mathbf{g}$ the gravity force.

We have recently developed a numerical solver for fluid dynamics, namely VSIAM3 [23, 24]. The VSIAM3 is a finite volume method that makes simultaneously use of the volume integrated average and the surface integrated average as the dependent variables. The VSIAM3 is devised for general fluid dynamics, and is found to be robust and accurate for multi-phase fluid simulations from our previous applications.

We incorporated the THINC to the VSIAM3 code. The air/water interface is identified by the 0.5 contour of the VOF function $f(x, y, z, t)$, which is governed by (1) and computed by the THINC scheme. The density and viscosity coefficient over the whole computational domain are defined by

$$p(x, y, z, t) = \bar{f}(x, y, z, t)\rho_\mathrm{l} + [1 - \bar{f}(x, y, z, t)]\rho_\mathrm{a} \tag{17}$$

Copyright © 2005 John Wiley & Sons, Ltd.

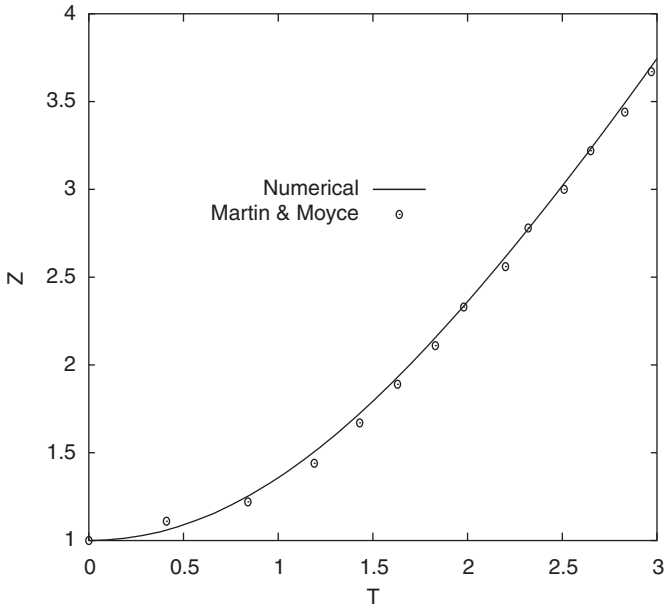

Figure 10. The displacement of the water front. Same as Reference [25], the normalized distance is defined by $Z = z/a$ with z being the real distance of the surge front and a the width of the initial water column. The time is normalized as $T = \sqrt{2g/at}$, where t is the real time and g the gravitational acceleration.

and

$$\mu(x, y, z, t) = \bar{f}(x, y, x, t)\mu_{\mathrm{l}} + [1 - \bar{f}(x, y, z, t)]\mu_{\mathrm{a}} \tag{18}$$

where the subscripts 'a' and 'l' denote air and liquid, respectively. The surface tension is computed with the continuum surface force formulation in Reference [25]. In the calculation of curvature, a level set function is created with the 0.5 contour of $\bar{f}$ being the interface.

In the real hydrodynamic simulations reported in this paper, the gravitational acceleration is $g = 9.8\,\mathrm{m/s^2}$. The densities for air and liquid are specified as $\rho_{\mathrm{a}} = 1.1763\,\mathrm{kg/m^3}$ and $\rho_{\mathrm{l}} = 996.62\,\mathrm{kg/m^3}$, respectively. The viscosity coefficients are $\mu_{\mathrm{a}} = 18.62 \times 10^{-6}\,\mathrm{Pa\,s}$ and $\mu_{\mathrm{l}} = 854.4 \times 10^{-6}\,\mathrm{Pa\,s}$.

In order to validate the THINC scheme presented in this paper, we computed the dam-breaking problem, which was experimentally studied by Martin and Moyce [26] and then widely used as a benchmark test for numerical models. Our computational domain is a $36\,\mathrm{in} \times 36\,\mathrm{in}$ 2D square. The initial rectangular water column has a width of $a = 2\frac{1}{4}$ inches and a height of $2a$. A mesh of 40×40 with uniform spacing is used. We validated the computational model by examining the displacement of the water front along the lower surface. The scaled displacement is plotted in Figure 10. The experimental data is obtained from Table II in Reference [26]. The computational result of the THINC scheme agrees well with the experiment. In order to examine the numerical smearing across the moving interface, we conducted another run with a 75×80 grid and plotted the 0.05, 0.5 and 0.95 contours of $\bar{f}$ in Figure 11.

Copyright © 2005 John Wiley & Sons, Ltd.

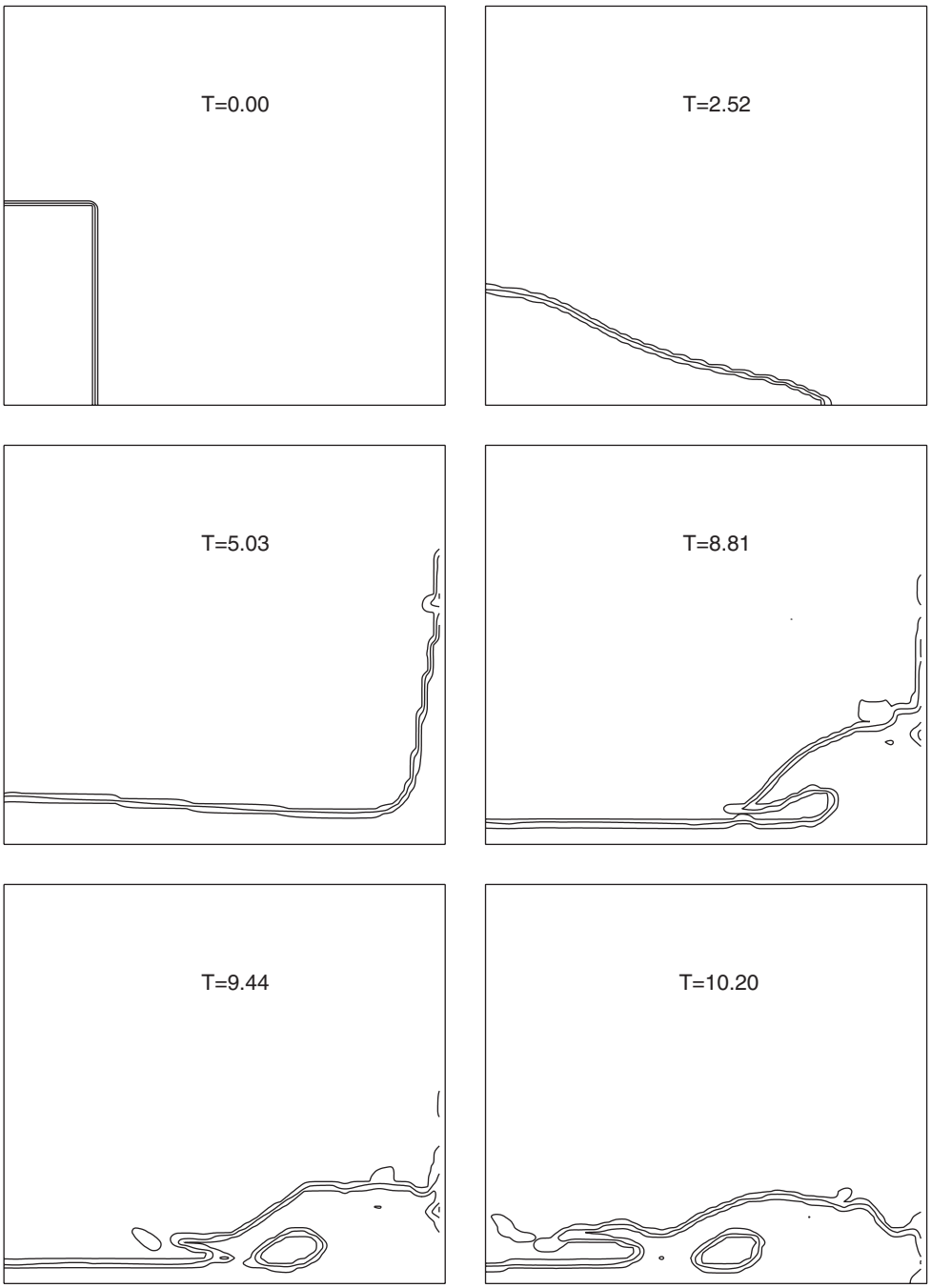

Figure 11. The VOF function of the density current generated by the release of a rectangular water dam. Displayed are the contours of 0.05, 0.5 and 0.95.

Copyright © 2005 John Wiley & Sons, Ltd.

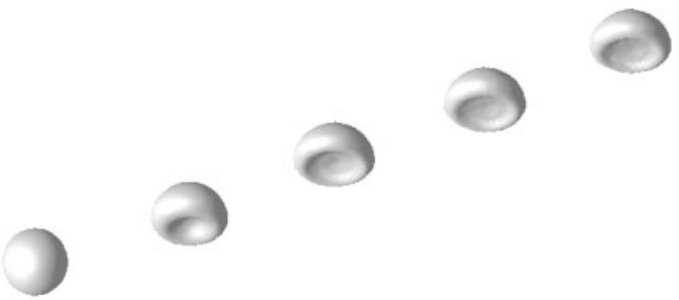

Figure 12. The snapshots of the rising bubble at different instants. Time increases from left to right.

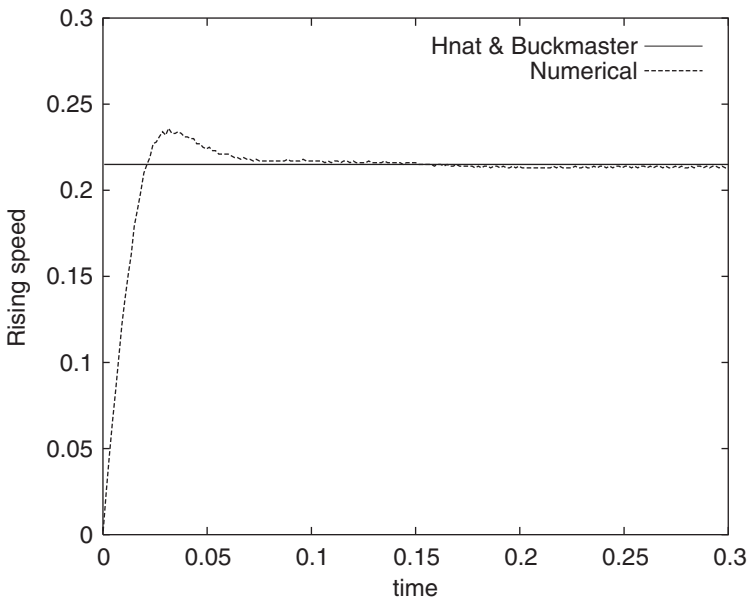

Figure 13. The rising speed of the bubble plotted against the experiment results in Reference [27].

As expected, the THINC method effectively prevents the smearing of the interface and the thickness of the transition layer of the interface was kept compact even after the interface has been heavily deformed with significant topological changes. The air bubbles trapped in the water after $T = 8.81$ has been reproduced with a well-defined interface.

The dynamics of a single bubble rising in a viscous liquid was simulated to verify the whole computational model. A systematic study on the behaviors of single bubble rising in a viscous fluid is found in Reference [27]. The experiment observations of several steady bubbles provide a good test bed for numerical models. Gueyffier *et al.* [28] and Sussman and Smereka [29] conducted axisymmetric simulations with fine grids. In the present study, we have carried out a 3D simulation for bubble A in Reference [27] with a relatively low resolution in a Cartesian mesh ($45^2 \times 160$). As shown in Figure 12, after the deformation

Copyright © 2005 John Wiley & Sons, Ltd.

from an initial spherical bubble in the early stage, the bubble reaches a stable shape and rises at a constant speed under the balance of surface tension force, inertial force and the viscous force. Figure 13 displays the bulk speed of the rising bubble. It is found that the THINC scheme gives an excellent agreement with the experimental result.

5. CONCLUSIONS

A simple and practical numerical method for capturing free boundary is devised using the hyperbolic tangent function. The resulting method, THINC, conserves the mass of transported quantity, and effectively eliminates the numerical diffusion and spurious oscillation from the moving interface. Our numerical tests show that the present method gives satisfactory results for even heavily deforming velocity field. Without geometrical surface reconstruction, the THINC scheme appears very computationally efficient and the implementation in 3D is straightforward. Numerical tests of both pure advection and multi-fluid computation reveal that the THINC method is promising and applicable to a wider spectrum of interfacial flows in real applications.

ACKNOWLEDGEMENTS

We gratefully acknowledge the anonymous reviewers for their constructive comments on the original manuscript, especially, those concerning the related works on the exponentially fitted technique.

REFERENCES

1. Unverdi SO, Tryggvason G. A front-tracking method for viscous, incompressible multi-fluid flow. *Journal of Computational Physics* 1992; **100**:25–37.
2. Glimm J, Graham MJ, Grove J, Li XL, Smith TM, Tan D, Tangerman F, Zhang Q. Front tracking in two and three dimensions. *Journal of Computational Mathematics* 1998; **7**:1–12.
3. Hirt CW, Nichols BD. Volume of fluid (VOF) methods for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
4. Lafaurie B, Nardone C, Scardovell R, Zaleski S, Zanetti G. Modeling merging and fragmentation in multiphase flows with SURFER. *Journal of Computational Physics* 1994; **113**:134–147.
5. Ubink C, Issa RI. A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics* 1999; **153**:26–50.
6. Noh WF, Woodward P. SLIC (simple line interface method). *Lecture Notes in Physics* 1976; **24**:330–340.
7. Youngs DL. Time-dependent multi-material flow with large fluid distortion. In *Numerical Methods for Fluid Dynamics*, Morton KW, Baines MJ (eds), vol. 24. Academic Press: New York, 1982; 273–285.
8. Ashgriz A, Poo JP. FLAIR: flux line-segment model for advection and interface reconstruction. *Journal of Computational Physics* 1991; **93**:449–468.
9. Puckett EG, Almgren AS, Bell JB, Marcus DL, Rider WJ. A high-order projection method for tracking fluid interface in variable density incompressible flows. *Journal of Computational Physics* 1997; **130**:269–282.
10. Rider WJ, Kothe DB. Reconstructing volume tracking. *Journal of Computational Physics* 1998; **141**:112–152.
11. Harvie DJE, Fletcher DF. A new volume of fluid advection algorithm: the stream scheme. *Journal of Computational Physics* 2000; **162**:1–32.
12. Pan D, Chang CH. The capturing of free surfaces in incompressible multi-fluid flows. *International Journal for Numerical Methods in Fluids* 2000; **30**:203–222.
13. Rudman M. Volume-tracking methods for interfacial flow calculations. *International Journal for Numerical Methods in Fluids* 1997; **24**:671–691.
14. Xiao F, Ikebata A. An efficient method for capturing free boundaries in multi-fluid simulations. *International Journal for Numerical Methods in Fluids* 2003; **42**:187–210.
15. Yang H. An artificial compression method for ENO schemes: the slope modification method. *Journal of Computational Physics* 1990; **89**:125–160.

Copyright © 2005 John Wiley & Sons, Ltd.

16. Allen DN de G, Southwell RV. Relaxation methods applied to determine the motion, in two dimensions, of a viscous fluid past a fixed cylinder. *Quarterly Journal of Mechanics and Applied Mathematics* 1955; **8**: 129–145.
17. El-Mistikawy TM, Werle MJ. Numerical method for boundary layers with blowing-the exponential box scheme. *AIAA Journal* 1978; **16**:749–751.
18. Schrfetter DL, Gummel HK. Large-signal analysis of a silicon read diode oscillator. *IEEE Transactions on Electron Devices* 1969; **ED-16**:64–77.
19. Ramos JI. On diffusive methods and exponential fitted techniques. *Applied Mathematics and Computation* 1999; **103**:69–96.
20. Yabe T, Xiao F. Description of complex and sharp interface during shock wave interaction with liquid drop. *Journal of the Physical Society of Japan* 1993; **62**:2537–2540.
21. Yabe T, Aoki T. A universal solver for hyperbolic-equations by cubic-polynomial interpolation. 1. One-dimensional solver. *Computer Physics Communications* 1991; **66**:219–232.
22. Rider WJ, Kothe DB. Stretching and tearing interface tracking methods. *Los Alamos National Laboratory Report*, *LA-UR-95-1145* (*AIAA-95-1717*), 1995.
23. Xiao F. Unified formulation for compressible and incompressible flows by using multi-integrated moments I: one-dimensional inviscid compressible flow. *Journal of Computational Physics* 2004; **195**:629–654.
24. Xiao F, Ikebata A, Hasegawa T. Numerical simulations of free-interface fluids by a multi-integrated moment method. *Computers and Structures* 2005; **83**:409–423.
25. Brackbill JU, Kothe DB, Zemach C. A continuum method for modeling surface tension. *Journal of Computational Physics* 1992; **100**:335–354.
26. Martin JC, Moyce WJ. An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Philosophical Transactions of the Royal Society of London Series A* 1952; **244**:312–324.
27. Hnat JG, Buckmaster JD. Spherical cap bubbles and skirt formation. *Physics of Fluids* 1976; **19**:182–194.
28. Gueyffier D, Li J, Nadim A, Scardovelli R, Zaleski S. Volume of fluid interface tracking with smoothed surface stress methods for three-dimensional flows. *Journal of Computational Physics* 1999; **152**:423–456.
29. Sussman M, Smereka P. Axisymmetric free boundary problems. *Journal of Fluid Mechanics* 1997; **341**: 269–294.

Copyright © 2005 John Wiley & Sons, Ltd.