Parasitic current generation in Combined Level Set and Volume of Fluid immiscible fluid simulations

D. J. E. Harvie[1] M. Rudman[2] M. R. Davidson[3]

(Received 31 August 2006; revised 8 January 2008)

Abstract

The Continuum Surface Force technique is a popular tool used to implement surface tension forces in Eulerian based Computational Fluid Dynamics codes. Under this technique, inaccuracies present in calculating the surface tension force can manifest themselves as 'parasitic' currents. We detail a new Combined Level Set and Volume of Fluid implementation of the Continuum Surface Force method. We then develop a correlation for the magnitude of parasitic currents that are generated under this new method, as a function of both the numerical and physical parameters employed in a simulation. A set of numerical experiments validate this correlation and show that, importantly, the magnitude of currents produced by the method decreases as the computational cell size reduced.

See http://anziamj.austms.org.au/ojs/index.php/ANZIAMJ/article/view/130 for this article, © Austral. Mathematical Soc. 2008. Published January 11, 2008. ISSN 1446-8735

Contents

1 Introduction

The Continuum Surface Force (CSF) technique is a popular tool used in Eulerian based Computational Fluid Dynamics (CFD) algorithms to model surface tension forces [1]. Under the CSF method, the stress jump that occurs across an interface between immiscible fluids is replaced by a volume force which acts on both fluids within a small region surrounding the interface. The magnitude and direction of the volume force is chosen so that in the limit of an infinitely small surrounding region, the total applied force approximates the stress jump that occurs across a real interface.

The principle advantage of the CSF method is that by reformulating the surface force as a volume force, it allows surface tension effects to be implemented in Eulerian CFD codes with relative ease. For this reason, it has found widespread application in Volume of Fluid (VOF) [3, 6] and Level Set (LS) [10, 7] based algorithms. The main downfall of the technique to date is that inaccuracies inherent in calculating the volume force mean that it is not possible to use the technique on all physically relevant systems. During a simulation, inaccuracies in the CSF force manifest themselves as errors in the calculated pressure and velocity fields. Errors in the velocity field appear as spurious currents, termed 'parasitic' currents [4].

Previously [2], we developed a correlation for the magnitude of these currents as a function of the numerical and physical parameters used in a simulation. This analysis was based on a CSF method implemented within a VOF CFD algorithm [5]. It was found, as has been widely recognised for all CSF VOF implementations, that the magnitude of generated parasitic currents does not decrease as the cell size is reduced. Level Set implementations of the CSF method are known to be more accurate in modelling surface tension forces. However, the principal disadvantage of the LS method is that mass conservation is not assured. A number of authors have recently presented Combined Level Set and Volume of Fluid (CLSVOF) methods which reportedly share the advantages of both the LS and VOF methods—accurate surface tension calculation combined with rigorous mass conservation [9, 8, 11].

In this article we detail a preliminary CLSVOF CSF technique which we have developed. We then develop a correlation for the maximum parasitic current magnitudes produced by this technique that would develop during a simulation as a function of the physical and numerical parameters used in the simulation. For simplicity we consider only systems where the density and viscosity of the phases are equal. The validity of the proposed correlation is tested by comparing predicted velocity magnitudes against those obtained from a series of numerical experiments. We show that, importantly, under the CLSVOF method the parasitic current magnitude decreases as the computational cell size is reduced.

2 The Combined Level Set and Volume of Fluid method

2.1 Multiphase flow solver

The multiphase CFD algorithm in which the CLSVOF method has been implemented is due to Rudman [5]. The algorithm solves the following volume averaged continuity, volume fraction and momentum equations on a structured, staggered mesh:

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0\,, \tag{1}$$

$$\frac{\partial \phi}{\partial t} + \boldsymbol{\nabla} \cdot \phi \boldsymbol{u} = 0\,, \tag{2}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{u}\boldsymbol{u} = -\boldsymbol{\nabla} p + \frac{1}{\mathrm{We}} \kappa_{\mathrm{v}} \boldsymbol{n}_{\mathrm{v}} + \frac{1}{\mathrm{Re}} \boldsymbol{\nabla}^2 \boldsymbol{u}\,. \tag{3}$$

In these equations ϕ is the volume averaged volume fraction of a chosen phase (termed the 'disperse' phase), and $\boldsymbol{u}$ is the volume averaged fluid velocity. The second term on the right of equation (3) is the force due to surface tension, represented as a volume force via the CSF technique. The variables in this term are κ_{v}, the local curvature of the interface, and $\boldsymbol{n}_{\mathrm{v}}$, a vector normal to the interface which is non-zero only in a small region surrounding each interface.

Equations (1–3) are employed in nondimensional forms. Velocity is scaled with $\bar{u}$, length with $\bar{x}$ and time with $\bar{x}/\bar{u}$. The nondimensional numbers appearing in equation (3) are the Reynolds and Weber numbers. These numbers, and the related capillary number, are defined as

$$\mathrm{Re} = \frac{\bar{\rho}\bar{u}\bar{x}}{\bar{\mu}}\,, \quad \mathrm{We} = \frac{\bar{\rho}\bar{v}^2\bar{x}}{\bar{\sigma}}\,, \quad \text{and} \quad \mathrm{Ca} = \frac{\mathrm{We}}{\mathrm{Re}} = \frac{\bar{v}\bar{\mu}}{\bar{\sigma}}. \tag{4}$$

These numbers represent the ratio between inertial and viscous, inertial and surface tension, and viscous and surface tension forces acting in the system, respectively. In these definitions $\bar{\rho}$ and $\bar{\mu}$ are the density and viscosity of either phase, respectively, and $\bar{\sigma}$ is the surface tension coefficient acting at the interface between phases (assumed constant).

2.2 Constructing the Level Set function

The CLSVOF algorithm detailed here differs from a conventional VOF algorithm only in the way in which the CSF force is calculated. Specifically, once the volume fraction field, ϕ, is updated, a signed distance function, s, is calculated using the positions of the reconstructed VOF interfaces in each mass cell as internal boundary conditions.

The algorithm used to calculate s is constructed from three basic routines:

1. *Initialisation in* VOF *interface cells:* The distance function is calculated in interface cells as the distance from the centre of that cell to a linear interface segment, positioned within the cell using VOF techniques. Under the VOF method, interface cells are defined as those which have a volume fraction in the range $\phi_{\text{cut}} < \phi < 1 - \phi_{\text{cut}}$. In our study $\phi_{\text{cut}} = 1 \times 10^{-7}$. An interface (a straight line in 2D) is positioned in these cells so that the cell volume is truncated into two regions having volumes in the ratio $\phi : 1 - \phi$. These interfaces are represented by equations of the form $\boldsymbol{n} \cdot \boldsymbol{x} = d$ where $\boldsymbol{n}$ is a unit normal. For our CLSVOF method the normals used to perform these reconstructions are based on $\boldsymbol{n}_{\text{ls}}$ from the previous timestep.

   Once the interfaces have been reconstructed the distance from the centre of each interface cell, $\boldsymbol{x}_{\text{c}}$, to the interface within that cell is calculated using

$$s = \boldsymbol{n} \cdot \boldsymbol{x}_{\text{c}} - d\,. \qquad (5)$$

As $\boldsymbol{x}_\mathrm{c}$ is positioned at the geometric centre of each mass cell, s values calculated using equation (5) are positive when $\phi > 1/2$, negative when $\phi < 1/2$ and zero when $\phi = 1/2$.

2. *Expansion*: To calculate s in cells which do not contain a VOF interface, we solve the equation
$$|\nabla s| = 1\,. \tag{6}$$
This equation is solved in an iterative manner by sweeping outwards from interface cells in the direction of increasing $|s|$. Locally, equation (6) is solved implicitly using first order approximations for the spatial derivatives, differenced in the direction of decreasing $|s|$. As neighbouring s values used in this differencing are more often than not already updated values, this results in a numerical scheme that is able to calculate s at large distances away from the interface for only a small computational cost. Similar schemes are employed to reinitialise the distance function in some LS methods [7].

3. *Iterative improvement*: In VOF interface cells where s is calculated using the reconstructed interface segments, s does not precisely satisfy $|\nabla s| = 1$. For this reason a pseudo transient improvement procedure is applied to s to ensure that the distance function varies smoothly and monotonically between phases. Specifically, each local s value is improved using
$$s^\star = \frac{1}{2}(1-w)\Delta x\,\mathrm{sign}(s)[1-|\nabla s|] + s\,, \tag{7}$$
where $s^\star$ is the next (improved) estimate of s. The derivatives used in $|\nabla s|$ are differenced in the same way as in the expansion routine. The weighting function
$$w = \max\left(1 - \frac{|s|}{\sqrt{2}\Delta x}, 0\right)\,. \tag{8}$$

> This weighting function ensures that changes to the s values specified by equation (7) are concentrated away from regions where the magnitude of s is small, that is, away from regions where the initial estimation of s is likely to be the most accurate. Equations similar to equation (7), but without the VOF weighting function, have been used in LS reinitialisation schemes [10, 7].

The above routines are applied in a sequential manner to calculate s. First, s is initialised in cells which contain a VOF interface. The s field is then expanded over a small region surrounding this interface. This expansion is performed so that cells next to VOF interface cells contain s values, allowing derivatives of s to be calculated in the VOF interface cells. Next, the improvement procedure is applied to the s values in a small region surrounding the interface. Finally, the expansion procedure is applied to all cells that are not VOF interface cells in which the s function is required for the CSF calculation.

Experimentation showed that the highest accuracy and stability of this scheme was achieved by iterating through the expansion and improvement steps a number of times. If the number of iterations performed on these routines was too low, then the 'errors' in s near the interface produced fluctuations in the curvature which tended to increase the magnitude of the generated parasitic currents. If the number of iterations performed on these routines was too high, then the link between s and the VOF data at outer regions of the VOF interface could be destroyed, promoting instability of the interface. For the results presented in Section 4, we used 20 iterations through the these two routines.

2.3 Implementing the continuum surface force

Once the distance function is known, normals are calculated as the gradient of s, and then these are then differentiated as in the VOF algorithm to find the curvature κ_v,

$$\boldsymbol{n}_\mathrm{ls} = \boldsymbol{\nabla} s \quad \text{and} \quad \kappa_\mathrm{v} = -\boldsymbol{\nabla} \cdot \frac{\boldsymbol{n}_\mathrm{ls}}{|\boldsymbol{n}_\mathrm{ls}|} \,. \tag{9}$$

The normals based on the distance function $\boldsymbol{n}_\mathrm{ls}$ require normalisation when calculating the curvature as the distance function that we construct from the VOF data does not satisfy $|\boldsymbol{\nabla} s| = 1$ precisely. The normals required in the CSF term are also based on the distance function. Recognising that an effective continuously varying volume fraction field can be defined from the signed distance function as

$$\phi(s) = \frac{1}{2}\left[\sin\left(\frac{s\pi}{2h\Delta x}\right) + 1\right], \tag{10}$$

the interface normals are computed using

$$\boldsymbol{n}_\mathrm{v} = \frac{\partial\phi}{\partial s}(s)\boldsymbol{n}_\mathrm{ls} \,. \tag{11}$$

In equation (10) Δx is the dimension of the computational cells (square in our study), and h is the radius of the convolution kernel as a multiple of Δx. The differentiation techniques used to compute the normals and curvature under the CLSVOF method are the same as those used in the Rudman [5] VOF algorithm, except that in our new method the calculations are performed on a mesh that is the same density as the momentum cell mesh, rather than one which is twice as fine.

3 Parasitic current correlation

As in the previous study concerned with the VOF method [2], to develop a correlation for the magnitude of parasitic currents that develop under the CLSVOF method we perform an order of magnitude analysis on momentum equation (3) with the CSF term included. We employ the conservative assumption that the erroneous rotational component of the surface tension force that drives the currents is balanced by only one of the velocity dependent terms in the momentum equation; the specific term being the one which balances the erroneous force with the smallest magnitude of generated currents.

3.1 Viscous limited currents

If we assume that the erroneous surface tension force is balanced by only the viscous term from equation (3), then in an order of magnitude sense we find

$$\mathcal{O}\left(\frac{1}{\mathrm{We}}\mathcal{E}\left(\kappa_{\mathrm{v}}\boldsymbol{n}_{\mathrm{v}}\right)\right)=\mathcal{O}\left(\frac{1}{\mathrm{Re}_{\mathrm{c}}}\boldsymbol{\nabla}^{2}\boldsymbol{u}_{\mathrm{p}}\right). \qquad (12)$$

Here $\mathcal{E}\left(\kappa_{\mathrm{v}}\boldsymbol{n}_{\mathrm{v}}\right)$ represents the rotational error in the numerically calculated surface tension force, and $\boldsymbol{u}_{\mathrm{P}}$ represents the parasitic velocity field.

To calculate an estimate for $\mathcal{E}\left(\kappa_{\mathrm{v}}\boldsymbol{n}_{\mathrm{v}}\right)$, we first note that within VOF interface cells, the interface segments that are reconstructed are linear and bounded by the cell boundaries. Thus, a point on an interface segment may be separated from the real interface by a distance which scales with the cell dimension. In interface cells, s is initialised as the minimum distance from the cell centre to a point on the reconstructed interface segment. As this point may be separated from the real interface by a distance which scales with the cell size, we find that in general, $\mathcal{E}\left(s\right)\propto\Delta x$. This result has been confirmed by numerical experiment.

The normals $\boldsymbol{n}_{\mathrm{v}}$ used in the CLSVOF calculation are calculated by differencing s using high order B-spline kernels as per equations (9) and (11). If s was a smoothly varying differentiable function, then we would expect the error in calculating a derivative of s using these B-spline convolving kernels to be proportional to Δx^{n}, where $n\geq 2$. However, the numerically calculated s

is not smoothly varying as it contains errors of magnitude $\mathcal{E}\left(s\right) \propto \Delta x$. When these errors are differenced using the convolving kernels, their magnitude is multiplied by the magnitude of the kernels, that is, by terms of $\mathcal{O}\left(1/(h\Delta x)\right)$. Thus, we find the normals calculated from the distance function using the convolution method contains errors of magnitude $\mathcal{E}\left(\boldsymbol{n}_{\mathrm{v}}\right) \propto 1/h$. Similarly, the curvature, being calculated by differencing the normals using the convolving kernels again contain errors of magnitude $\mathcal{E}\left(\kappa_{\mathrm{v}}\right) \propto 1/(h^2\Delta x)$. Both of these results have been confirmed numerically. Recall in these expressions that h is the radius of the convolving kernel, expressed in cell dimensions.

To estimate the total error in the product of the normal and curvature, $\mathcal{E}\left(\kappa_{\mathrm{v}}\boldsymbol{n}_{\mathrm{v}}\right)$, we note that provided $\mathcal{O}\left(\boldsymbol{n}_{\mathrm{v}}\right) \gg \mathcal{E}\left(\boldsymbol{n}_{\mathrm{v}}\right)$ and $\mathcal{O}\left(\kappa_{\mathrm{v}}\right) \gg \mathcal{E}\left(\kappa_{\mathrm{v}}\right)$, the error in the product of the normal and the curvature will be proportional to the larger of the errors in each term. Assuming that the cell dimension Δx is small, then $\mathcal{E}\left(\kappa_{\mathrm{v}}\right) \gg \mathcal{E}\left(\boldsymbol{n}_{\mathrm{v}}\right)$ and we find $\mathcal{E}\left(\kappa_{\mathrm{v}}\boldsymbol{n}_{\mathrm{v}}\right) \propto 1/(h^2\Delta x)$.

Returning to equation (12), the error in the surface tension force must be balanced by the viscous term on the right of this equation. Parasitic currents are observed to circulate over dimensions that scale with the cell size, so we would expect the Laplacian on the right of this equation to scale as $U_{\mathrm{P}}/\Delta x^2$. Here U_{P} is a measure of the parasitic current magnitude. Using this and our surface tension force error estimate in equation (12) leads to

$$U_{\mathrm{P}} \propto U_{\mathrm{V}} = \frac{\Delta x}{\mathrm{Ca}h^2} , \tag{13}$$

for currents that are limited by the viscous term.

3.2 Inertial advection limited currents

If we assume that the currents are entirely limited by the inertial advection term in momentum equation (3), we have

$$\mathcal{O}\left(\frac{1}{\mathrm{We}}\mathcal{E}\left(\kappa_{\mathrm{v}}\boldsymbol{n}_{\mathrm{v}}\right)\right) = \mathcal{O}\left(\boldsymbol{\nabla}\cdot\boldsymbol{u}_{\mathrm{P}}\boldsymbol{u}_{\mathrm{P}}\right) . \tag{14}$$

On the left of this equation we use the same curvature and normal product error estimate as before. On the right, we note that when currents are limited by the inertial advection term, they still circulate over dimensions that are proportional to the cell size so that $\mathcal{O}\left(\boldsymbol{\nabla}\cdot\boldsymbol{u}_{\mathrm{P}}\boldsymbol{u}_{\mathrm{P}}\right)\propto U_{\mathrm{P}}^2/\Delta x\,.$ Applying these expressions in equation (14) leads to

$$U_{\mathrm{P}}\propto U_{\mathrm{A}}=\frac{1}{h}\sqrt{\frac{1}{\mathrm{We}}}\,,\tag{15}$$

for currents that are limited by the inertial advection term.

3.3 Inertial transient limited currents

Finally, if we assume that the erroneous surface tension force is entirely limited by the inertial transient term, then

$$\mathcal{O}\left(\frac{1}{\mathrm{We}}\mathcal{E}\left(\kappa_{\mathrm{v}}\boldsymbol{n}_{\mathrm{v}}\right)\right)=\mathcal{O}\left(\frac{\partial\boldsymbol{u}_{\mathrm{P}}}{\partial t}\right).\tag{16}$$

On the left we use the same error estimate as before. On the right, we note that as the inertial transient term in the momentum equations represents an Eulerian rate of change, the temporal derivative will scale as $\mathcal{O}\left(\partial\boldsymbol{u}_{\mathrm{P}}/\partial t\right)\propto U_{\mathrm{P}}/t_{\mathrm{m}}$, where t_{m} is the maximum time that any particular mesh cell contains an interface region [2]. Using this estimate in equation (16) leads to

$$U_{\mathrm{P}}\propto U_{\mathrm{T}}=\frac{t_m}{\mathrm{We}\,h^2\Delta x}\,,\tag{17}$$

for a balance between the erroneous surface tension force and the inertial transient term.

3.4 Combined correlation

Combining equations (13), (15) and (17), using our assumption that the currents will be limited by the term in the momentum equations that produces the lowest magnitude of currents, we find

$$U_{\mathrm{P}} = \min(a_{\mathrm{T}} U_{\mathrm{T}}, a_{\mathrm{A}} U_{\mathrm{A}}, a_{\mathrm{V}} U_{\mathrm{V}}), \tag{18}$$

where a_{T}, a_{A} and a_{V} are proportionality constants to be determined by numerical experimentation.

4 Numerical experiments

To validate the form of equation (18), and to evaluate the magnitude of the proportionality constants a_{T}, a_{A} and a_{V}, we perform a series of numerical experiments using the CLSVOF method. The problem that we use is of a two dimensional planar droplet (a disc) of radius 1/2, centred in a square domain of size 2×2 . Ideally for this problem the velocities should be zero everywhere. We consider cell dimensions in the range $1/256 \leq \Delta x \leq 1/8$, kernel radii in the range $1.75 \leq h \leq 7$, and allow the dimensionless numbers to vary across the ranges of $10^{-6} \leq \mathrm{We} \leq 10^{2}$, $10^{-6} \leq \mathrm{Ca} \leq 10^{2}$, and $10^{-2} \leq \mathrm{Re} \leq 10^{2}$.

Figure 1 shows the results of these experiments. In this graph we plot the maximum current component magnitude against that predicted by equation (18). The proportionality constants, which are given in the figure, have been evaluated using a regression analysis to minimise the difference between the correlation and measured current magnitudes. The symbol type in the figure indicates which term of the correlation was used to limit the currents. As shown, the correlation predicts the magnitude of the generated parasitic currents for the CLSVOF method as they vary over eight orders of magnitude.

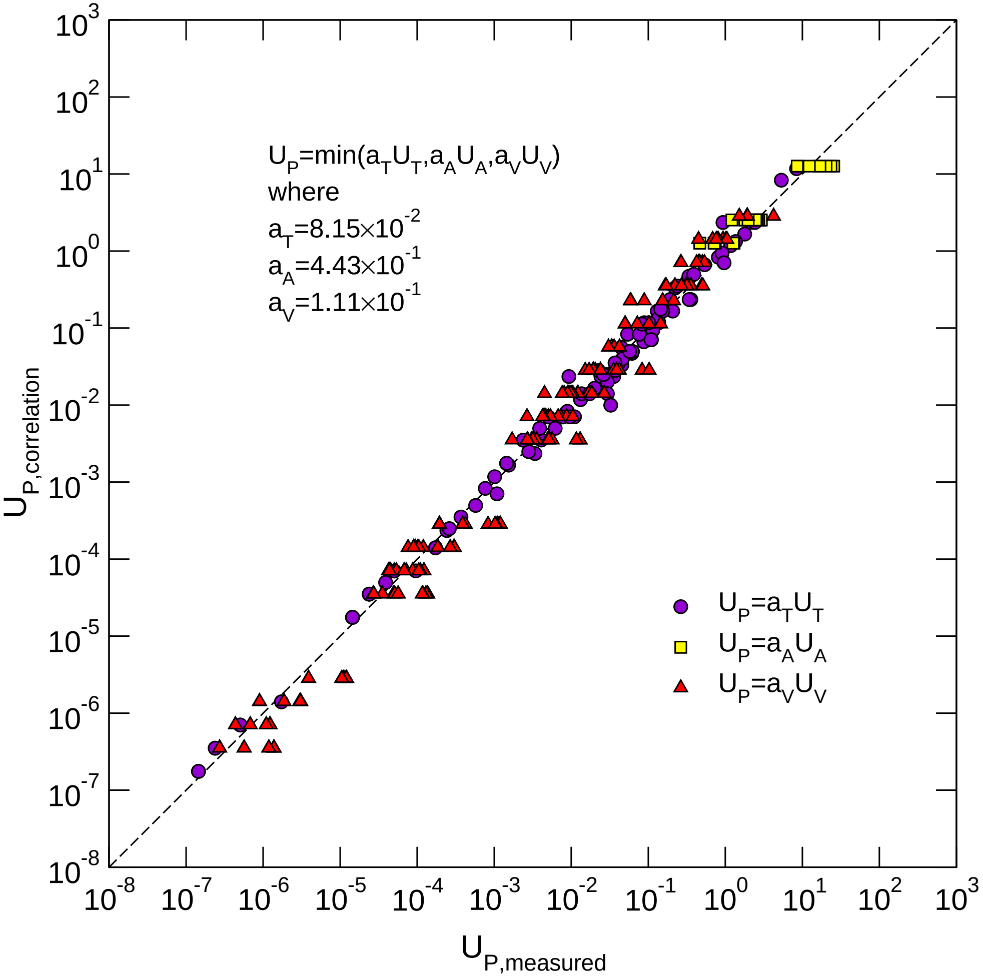

Figure 1: Parasitic current magnitudes predicted by the CLSVOF correlation plotted against those measured in the numerical experiments. The particular term of the correlation that limited the current magnitude is indicated by the symbol type.

5 Discussion

The parasitic current correlation for the CLSVOF method can be written as

$$U_{\mathrm{P}} = \min\left(a_{\mathrm{T}}\frac{t_m}{\mathrm{We}\,h^2\Delta x}, a_{\mathrm{A}}\frac{1}{h}\sqrt{\frac{1}{\mathrm{We}}}, a_{\mathrm{V}}\frac{\Delta x}{\mathrm{Ca}\,h^2}\right). \tag{19}$$

Examining this equation, we see that as $\Delta x \to 0$, $U_{\mathrm{P}} = a_{\mathrm{V}}\Delta x/(\mathrm{Ca}\,h^2) \to 0$. Thus, under the CLSVOF method, the current magnitude decreases to zero as the cell size is reduced. This is in contrast to the VOF method, where we find that as $\Delta x \to 0$, $U_{\mathrm{P}} \to a_{\mathrm{V}}/(\mathrm{Ca}\,h^2)$, a constant [2].

That the magnitude of parasitic currents decreases with increasing mesh refinement under the CLSVOF method is the principal result of this article. It means that, in theory, any given level of velocity field accuracy can be attained in an immiscible fluid simulation, given enough computational resources. In practical terms, it means that via the CLSVOF method, it will be possible to model a wider variety of physically relevant systems than via the VOF method, particularly systems which involve small length scales and strong surface tension forces.

Acknowledgements This research was supported by the Australian Research Council Grants Scheme.

References

[1] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modelling surface tension. *Journal of Computational Physics*, 100:335–354, 1992. C869

[2] Dalton J. E. Harvie, M. R. Davidson, and Murray Rudman. An analysis of parasitic current generation in volume of fluid simulations. *Appl. Math Mod.*, 30(10):1056–1066, 2006. doi:10.1016/j.apm.2005.08.015 C870, C875, C878, C881

[3] D. B. Kothe and R. C. Mjolsness. RIPPLE: A new model for incompressible flows with free surfaces. *American Institue of Aeronautics and Astronautics*, 30 (11):2694–2700, 1992. C870

[4] Bruno Lafaurie, Carlo Nardone, Ruben Scardovelli, Stéphane Zaleski, and Gianluigi Zanetti. Modelling merging and fragmentation in multiphase flows with SURFER. *Journal of Computational Physics*, 113:134–147, 1994. C870

[5] Murray Rudman. A volume-tracking method for incompressible multifluid flows with large density variations. *International Journal for Numerical Methods in Fluids*, 28:357–378, 1998. C870, C871, C875

[6] Ruben Scardovelli and Stèpahe Zaleski. direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, 31:567–603, 1999. C870

[7] J. A. Sethian and Peter Smereka. Level set methods for fluid interfaces. *Annual Review of Fluid Mechanics*, 35:341–372, 2003. doi:10.1146/annurev.fluid.35.101101.161105 C870, C873, C874

[8] Mark Sussman. A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. *Journal of Computational Physics*, 187:110–136, 2003. doi:10.1016/S0021-9991(03)00087-1 C870

[9] Mark Sussman and Elbridge Gerry Puckett. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *Journal of Computational Physics*, 162:301–337, 2000. doi:10.1006/jcph.2000.6537 C870

[10] Mark Sussman, Peter Smereka, and Stanley Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114:146–159, 1994. C870, C874

[11] S. P. van der Pijl, A. Segal, C. Vuik, and P. Wesseling. A mass-conserving level-set method for modelling of multi-phase flows. *International Journal for Numerical Methods in Fluids*, 47:339–361, 2005. doi:10.1002/fld.817 C870

Author addresses

1. **D. J. E. Harvie**, Dept. of Chemical and Biomolecular Engineering, University of Melbourne, Australia.
   mailto:daltonh@unimelb.edu.au

2. **M. Rudman**, CSIRO Mathematical and Information Sciences, Melbourne, Australia.

3. **M. R. Davidson**, Dept. of Chemical and Biomolecular Engineering, University of Melbourne, Australia.