Engineering Applications of Computational Fluid Mechanics Vol. 2, No. 3, pp. 284–298 (2008)

A NUMERICAL MODEL FOR MULTIPHASE FLOW BASED ON THE GMPPS FORMULATION, PART II: DYNAMICS

Frank Bierbrauer*[#] and Song-Ping Zhu**

** School of Mathematics, Cardiff University, Cardiff, CF24 4AG, UK*
*** School of Mathematics and Applied Statistics, University of Wollongong, NSW, 2522, Australia*
E-Mail: spz@uow.edu.au (*Corresponding Author*)

ABSTRACT: A numerical method for simulating variable density multiphase fluid flows is presented in this paper. It is a hybrid Eulerian-Lagrangian method solving the incompressible Navier-Stokes equations on a fixed Eulerian grid for velocity and pressure. Fluid phase is tracked in a Lagrangian manner by using marker-particles (Bierbrauer and Zhu, 2007a) to update the grid density and viscosity. A high order Godunov advection scheme, together with an approximate projection scheme to ensure incompressibility, is used to deal with rapid changes in fluid properties. We call this scheme the Godunov-Marker-Particle Projection Scheme (GMPPS), which implements pressure corrections to avoid spurious numerical boundary layers for multiphase flows. We have validated the GMPPS by comparing our numerical results with those calculated from an exact solution, with constant density, as well as some available physical experimental results for a variable density case. It is demonstrated that the scheme has second-order accuracy in time and space for a range of Reynolds numbers. The fact that test results of a two-phase droplet impacting on solid and liquid surfaces compare well with experimental results suggests that the GMPPS has great potential to be used to solve multiphase flow problems.

Keywords: multiphase flow, dynamics, incompressible fluids, projection method

1. INTRODUCTION

There are many inherent difficulties when attempting to model multiphase[1] flow problems arising in nature or in industrial processes. One such process is the impact of a fluid droplet onto a solid or liquid surface and the consequences of such impacts, for example, raindrop-induced erosion in agriculture (Ghadiri, 2004) or spray cooling of galvanised steel sheets in industry (Bierbrauer, 2004). Depending on the characteristic scales of an individual droplet, the impact process is governed by the relative interplay of inertial, viscous and capillary forces within the fluids considered. Accurate depiction and tracking of the interacting fluid interfaces within a robust multi-phase flow algorithm (MPFA) remains a difficult task. One such interface tracking algorithm (ITA), the Marker-Particle method (MP), has been described in a previous paper by the authors (Bierbrauer and Zhu, 2007a). The aim of this paper is to apply the MP method as part of a larger multi-phase flow algorithm and examine its accuracy and efficiency in solving two-phase droplet impact problems: impact of a droplet onto a solid surface and impact of a droplet onto a layer of the same fluid. As an initial test of the method, the current study is limited to two dimensions with three dimensional aspects to be included at a later date.

Typically, any general multi-phase flow algorithm possesses two main aspects: the ITA and the other parts of the MPFA which solve the problem itself. The main aspects that such a flow algorithm must be able to deal with include the non-linear convective term in the Navier-Stokes (NS) equations and the viscous term with variable viscosity and surface tension. Most importantly, it must transfer information, such as velocity, from the MPF algorithm to the ITA, and retrieve information, such as grid density, from it. This paper is concerned with the solution of unsteady, viscous, incompressible two-phase flows of immiscible fluids. Consider, at first, the MPFA.

Current interest rests with incompressible multi-phase flows as these are commonly observed in many natural and industrial processes where the characteristic velocities do not exceed the speed of sound in the media considered. Research efforts have shown that an Eulerian projection method remains one of the best available methods to model incompressible multi-phase flow problems. It does not only maintain an excellent

[#] Most of the work for this paper was conducted while the author was at the University of Wollongong.

[1] Specifically, multi-phase refers to the interaction of two distinct, immiscible fluids, e.g., a water droplet in air, rather than two thermodynamic phases, e.g., liquid water and steam.

Received: 21 Sep. 2007; Revised: 18 Jan. 2008; Accepted: 12 Feb. 2008

solenoidal velocity field but also allows the inclusion of viscous and surface tension forces in a natural way. One way to construct such a model, avoiding the awkward application of interfacial conditions among multiple flow solutions (Lakehal, Meier and Fulgosi, 2002), is to adopt the One-Field or single-domain approach (Anderson, McFadden and Wheeler, 1998; Scardovelli and Zaleski, 1999). All interacting fluid phases are now part of a single fluid (Vincent and Caltagirone, 2000; Rider et al., 1995) and velocity and pressure become global field variables. Bulk fluid properties, such as density and viscosity, remain constant within each phase but vary discontinuously across fluid interfaces. The fluid interface itself is then "immersed" within this global fluid. The fluid phases are dynamically connected through interfacial conditions of impermeability for immiscible fluids (continuity of normal velocity), for viscous flows the dynamic condition (continuity of tangential velocity) and in the presence of surface tension the jump in stress across the interface. These conditions now become part of the NS equations themselves (and may be recovered (Anderson, McFadden and Wheeler, 1998)) through the inclusion of a surface tension force term written as a body force (Brackbill, Kothe and Zemach, 1992), i.e., possessing a value everywhere in the domain, although only non-zero near fluid interfaces.

It is a relatively simple procedure to construct a numerical model by using a finite difference method and the well-known high order, semi-implicit Godunov method to accurately construct the non-linear term across material boundaries (Colella, 1990). This procedure is adopted in this paper. In order to avoid local grid decoupling and improve accuracy, the approximate projection method was developed (Almgren, Bell and Szymczak, 1996; Almgren, Bell and Crutchfield, 2000) where the solenoidal constraint is relaxed so that it is only true to within the truncation error of the mesh spacing (Guy and Fogelson, 2005). Here the fully developed second-order accurate approximate projection method of Rider (1994) and Puckett et al. (1997) is used within a collocated grid for velocity and pressure. In addition, the pressure correction of Brown, Cortez and Minion (2001) is updated here for the variable density case ensuring second-order accuracy in time as well as consistency of the pressure gradient.

The projection itself is imposed through the solution of a Poisson equation for a gauge variable which is used to eliminate non-solenoidal modes within the velocity field obtained from the momentum equation. Unfortunately, as the density field is discontinuous across fluid interfaces, a Poisson equation with discontinuous coefficients must be solved. This means that, for general multi-phase flows, second-order accuracy cannot be expected from a direct application of this method (Szymczak et al., 1993). Such density discontinuities may be lessened by smoothing the sharp gradient in density in the neighbourhood of the discontinuity (Sussman, Smereka and Osher, 1994) and this method is used in the present study. It is also noteworthy that the initial velocity field must be divergence-free so as not to carry non-solenoidal errors forward in the solution. This can be ensured either by obtaining an analytical solution for the solenoidal initial velocity field (Bierbrauer and Zhu, 2007b) or by projecting the initial flow field, obtaining a good initial solenoidal velocity and an approximation to the pressure (Rider, 1994). One of the drawbacks associated with approximate projection methods on collocated grids is the existence and growth of null spaces in the discrete divergence operator giving rise to "checkerboard modes" where the discrete operator fails to recognise non-divergence-free modes (Rider et al., 1998; Nourgaliev, Dinh and Theofanous, 2003). In the current method, these modes are damped through the use of iterated projection and velocity filters as discussed by Rider (1994).

The MPFA, discussed in the previous paragraphs, has been shown to be an accurate, robust method of high fidelity (Rider et al., 1995). It has been used to solve constant density flow problems (Bell, Colella and Glaz, 1989), and has been extended to the variable density case (Bell and Marcus, 1992). Its strengths and weaknesses have been investigated in depth (Almgren, Bell and Crutchfield, 2000; Guy and Fogelson, 2005; Brown, Cortez and Minion, 2001). More recently, it has been used in conjunction with various ITAs, especially Level-Set (Sussman, Smereka and Osher, 1994; Sussman et al., 1999) and VOF methods (Vincent and Caltagirone, 2000; Puckett et al., 1997), to solve two-phase flow problems. While Level-Set methods possess problems in conserving mass and dealing with severe interfacial deformations, VOF methods cannot simulate fluid chunks which are smaller than a grid cell. In order to avoid such problems when reconstructing fluid interfaces, it was noted that pure Lagrangian particle methods such as PIC (Brackbill and Ruppel, 1986) and SPH (Ritchie and Thomas, 2001; Colagrossi and Landrini, 2003) track the interface implicitly without having to

construct the interface explicitly and therefore avoiding large derivatives at interfacial cusp points. As well, they have the potential to maintain fluid identity for all time. The MP method is a particle tracking algorithm which uses aspects of previous PIC methods. Particles are used only to track fluid colour (identity) so that fluid identity is never lost. Fluid colour is interpolated back to the grid as grid volume fractions from which grid density and viscosity may be reconstructed. This makes the combination of Eulerian and Lagrangian aspects an Eulerian-Lagrangian method for multiphase flows. For this reason the method is named the Godunov Marker-Particle Projection Scheme (GMPPS). The MP method has already been validated for flows where the flow field has been explicitly given and no forces are acting. It is the aim of this paper to combine the MP method with the Godunov projection scheme and apply the new method to complicated two-phase flow problems which undergo severe interfacial deformations and disruptions. The droplet-solid and droplet-fluid impact problems satisfy these criteria.

The droplet-solid impact problem (DSIP) has been extensively studied in the literature and requires the method to handle both large discontinuous variations in density and viscosity as well as the disruption of fluid interfaces as the droplet impacts the surface. The experimental data of Rioboo, Marengo and Tropea (2002) are used for comparisons. The droplet-liquid impact problem (DLIP) is less well studied but in addition to the difficulties associated with the DSIP, this problem is ideal for studying the growth of instabilities which may arise over longer calculation times (Rider et al., 1995). The experimental data of Liow (2001) for a deep layer and of Josserand and Zaleski (2003) for a shallow layer are used for comparisons.

The paper is organised as follows: Section 2 describes the Godunov projection method in detail including the physical system described by the non-dimensionalised NS equations and a semi-discrete version of the projection method including the pressure update for variable density flows. The discrete differential operators used and the allocation of boundary conditions and ghost cells are also detailed. The smoothing of material properties is also briefly discussed. Section 3 describes the dynamical tests used in the validation of the method including an exact solution of the incompressible NS equations for constant density flow in order to demonstrate the efficacy of the Godunov projection part of the method. The second set of tests includes the droplet-solid and droplet-liquid impact problems which test both the MPFA and the MP method. Finally, the paper ends with some concluding remarks in Section 4.

2. THE GODUNOV PROJECTION METHOD

2.1 The NS equations

For each of the problems considered in this paper, the unsteady incompressible Navier-Stokes equations are solved and are defined by the non-dimensionalised system in the domain $\Omega = \{(x,\ y):\ 0 < x < X,\ 0 < y < Y\}$, with the only external force being gravity for the droplet impact problems, i.e., $Fr \neq \infty$.

$$\begin{aligned}
&\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} = -\frac{1}{\rho}\left(\nabla p - \frac{1}{Re}\nabla\cdot\mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right) + \frac{\rho}{Fr}\hat{\mathbf{j}}\right),\\
&\nabla\cdot\mathbf{u} = 0,\\
&\frac{\partial C}{\partial t} + \mathbf{u}\cdot\nabla \mathrm{C} = 0,\\
&\rho = C + (1-C)\rho_{12},\\
&\mu = \left(C + (1-C)\mu_{12}\right)^{-1}
\end{aligned} \tag{1}$$

where we have non-dimensionalised with respect to the scales in: length D, velocity U_0, density ρ_1, viscosity μ_1, pressure $\rho_1 U_0^2$ and the convective time D/U_0. Here $\rho_{12} = \rho_2/\rho_1$ and $\mu_{12} = \mu_2/\mu_1$; C is the volume fraction, the Reynolds number is $Re = \rho_1 U_0 D/\mu_1$ and the Froude number is $Fr = U_0^2 / gD$. While the density update is a direct consequence of the mass conservation equation, i.e., the simple geometric or serial average seen in the fourth equation of (1), the viscosity update is not a natural consequence of a physical law. A harmonic or parallel average was found by Kothe (1999) to give better results in practice whereas a serial average sometimes caused unphysical acceleration of fluid elements. Therefore, the harmonic average shown in the fifth equation of (1) was used. For both the exact solution having constant density we use $\rho = \rho_1 = \rho_2$ and $\mu = \mu_1 = \mu_2$. For the droplet impact problems the droplet density, viscosity, diameter and impact velocity were used as scales with the volume fraction being that of the droplet fluid.

Boundary conditions are a specified velocity on the boundary $\mathbf{u}_b$ of the domain, usually solid walls, and no flux conditions on the density, viscosity and consequently the volume fraction, i.e.,

$$\mathbf{u}|_{\partial\Omega} = \mathbf{u}_b,\ \mathbf{n}\cdot\nabla\rho|_{\partial\Omega} = 0,\ \mathbf{n}\cdot\nabla\mu|_{\partial\Omega} = 0,\ \mathbf{n}\cdot\nabla C|_{\partial\Omega} = 0 \tag{2}$$

All of the test problems, except the exact solution, used solid walls $\mathbf{u}_b = \mathbf{0}$. The exact solution required specified velocity values at the boundary. The initial condition (IC) in each case was problem dependent. The initial condition for the exact solution is determined at $t = 0$ directly from the exact solution. For the two droplet impact problems all initial velocities were zero except the droplet impact velocity, $U_0 \neq 0$. Initial densities and viscosities were constant in each fluid and all initial pressures were zero.

2.2 The semi-discrete projection method

Systems (1) and (2) are solved with the approximate Godunov projection method with second-order Crank-Nicolsen time discretisation. Given the strengths and weaknesses of the various projection methods, we choose to use the variable density version of PMII as described by Guy and Fogelson (2005). In semi-discrete form, the corrected, variable density, second-order PMII approximation of the momentum and mass conservation equations in (1) at time t^{n} reads:

- Step 1: given $\mathbf{u}^n$, $p^{n-1/2}$, ρ^{n+1}, μ^{n+1}, C^{n+1} solve for the intermediate velocity

$$\left(\mathbf{I} - \frac{\Delta t}{2Re}\sigma^{n+1/2}L_\mu^{n+1/2}\right)\mathbf{u}^* = \left(\mathbf{I} + \frac{\Delta t}{2Re}\sigma^{n+1/2}L_\mu^{n+1/2}\right)\mathbf{u}^n - \Delta t\left((u\cdot\nabla u)^{n+1/2} + \sigma^{n+1/2}\nabla p^{n-1/2} + \frac{1}{Fr}\hat{\mathbf{j}}\right) \quad \text{in } \Omega \quad (3)$$

$$\mathbf{u}^* = \mathbf{u}_b^{n+1},\ \mathbf{n}\cdot\nabla\rho^{n+1/2} = 0,\ \mathbf{n}\cdot\nabla\mu^{n+1/2} = 0 \quad \text{on } \partial\Omega$$

- Step 2: project the result

$$L_\sigma^{n+1/2}\phi^{n+1} = \frac{1}{\Delta t}\nabla\cdot\mathbf{u}^* \quad \text{in } \Omega$$
$$\mathbf{n}\cdot\nabla\phi^{n+1} = 0 \quad \text{on } \partial\Omega \quad (4)$$

followed by

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \Delta t\sigma^{n+1/2}\nabla\phi^{n+1} \quad \text{in } \overline{\Omega} \quad (5)$$

- Step 3: update the pressure gradient

$$\nabla p^{n+1/2} = \nabla p^{n-1/2} + \nabla\phi^{n+1} - \frac{\Delta t}{2Re}L_\mu^{n+1/2}\left(\sigma^{n+1/2}\nabla\phi^{n+1}\right) \quad \text{in } \overline{\Omega} \quad (6)$$

where the Laplacian operators are given by $\mathrm{L}_\mu\mathbf{w} = \nabla\cdot\mu(\nabla\mathbf{w} + (\nabla\mathbf{w})^T)$, for some vector $\mathbf{w}$, and $L_\sigma\psi = \nabla\cdot\sigma\nabla\psi$, for some scalar ψ, $\sigma = 1/\rho$, $\sigma^{n+1/2} = (\sigma^n + \sigma^{n+1})/2$, $\mu^{n+1/2} = (\mu^n + \mu^{n+1})/2$ and $\overline{\Omega} = \Omega \cup \partial\Omega$. The $(\mathbf{u}\cdot\nabla\mathbf{u})^{n+1/2}$ term represents an approximation to the non-linear advection term at the half time level and is the one detailed in Puckett et al. (1997). The velocity at half time level has also been used in the viscous term and is given by $\mathbf{u}^{n+1/2} = (\mathbf{u}^n + \mathbf{u}^*)/2$. The extra correction term $-(\Delta t/2Re)L_\mu\sigma\nabla\phi$ was obtained by eliminating the intermediate velocity in the momentum equation (3) using the update given by (5). An analogous expression to that obtained in Brown, Cortez and Minion (2001) may be obtained when this extra term is expanded out and the Poisson equation (4) is used to replace the $L_\sigma\phi$ operator which gives rise to

$$\frac{\Delta t}{2Re}L_\mu\sigma\nabla\phi = \frac{\mu}{2Re}(\nabla\nabla\cdot\mathbf{u}^*) - \frac{\Delta t}{4Re}(\mu\nabla\times\nabla\sigma\times\nabla\phi - 2\nabla\mu\cdot\nabla\sigma\nabla\phi - \nabla\mu\times\nabla\sigma\times\nabla\phi) \quad (7)$$

which reduces to $\mu(\nabla\nabla\cdot\mathbf{u}^*)/2Re$ when $\nabla\mu = \nabla\sigma = 0$ so that the result in Brown, Cortez and Minion (2001) remains true within each individual fluid but not near fluid interfaces where the remaining terms on the right of Eq. (7) effect the result.

Note that in the projection method described above, the time updated density and viscosity are required even in Step 1 where only the values at the n^{th} time level are known, i.e., ρ^n and μ^n. This is calculated with the use of the marker-particle method of Bierbrauer and Zhu (2007a) where fluid particles are advanced forward in time to track individual fluid phases while carrying particle colour information. The time updated volume fraction C^{n+1} is then obtained by interpolation from surrounding fluid particles. This provides the solution of the advection equation for the volume fraction, the third equation in (1). The density and viscosity at the next time level are then given by

$$\rho^{n+1} = C^{n+1} + (1 - C^{n+1})\rho_{12},\ \mu^{n+1} = \left(C^{n+1} + (1 - C^{n+1})\mu_{12}\right)^{-1}$$

The densities and viscosities at the half time level can now be constructed, i.e., $\rho^{n+1/2} = (\rho^n + \rho^{n+1})/2$ and $\mu^{n+1/2} = (\mu^n + \mu^{n+1})/2$.

2.3 Spatial discretisation of differential operators

A cell-centred computational grid defined in Bierbrauer and Zhu (2007a) with velocity $\mathbf{u}_{i,j} = (u_{i,j}, v_{i,j})$ and pressure $p_{i,j}$ defined at cell centres (x_i, y_j) is used although some use is still made of cell-edge centred values, e.g., $(x_{i+1/2}, y_j)$ is the right cell edge. The integer counters are defined by $i = \{1,2,3,...,I\}$ and $j = \{1,2,3,...,J\}$ so

that there are a total of $I \times J$ number of internal domain cells. The discrete divergence and gradient operators used individually in the momentum and mass conservation equations are second-order accurate cell-centred discretisations defined by

$$(D \cdot \mathbf{u})_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta x} + \frac{v_{i,j+1} - v_{i,j-1}}{2\Delta y} \tag{8}$$

and

$$(\mathbf{G}\phi)_{i,j} = \left(\frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}, \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta y} \right) \tag{9}$$

Approximate projection methods make use of discrete approximations of the continuous operator $\nabla \cdot \sigma \nabla$ instead of using the discrete divergence and gradient operators (8), (9), that is $L_\sigma \neq D \cdot \sigma \mathbf{G}$. This gives rise to a compact five-point Laplacian

$$(L_\sigma \phi)_{i,j} = \frac{\sigma_{i+1/2,j}(\phi_{i+1,j} - \phi_{i,j}) - \sigma_{i-1/2,j}(\phi_{i,j} - \phi_{i-1,j})}{\Delta x^2} + \frac{\sigma_{i,j+1/2}(\phi_{i,j+1} - \phi_{i,j}) - \sigma_{i,j-1/2}(\phi_{i,j} - \phi_{i,j-1})}{\Delta y^2} \tag{10}$$

The analogous form for $L_\mu \mathbf{u}$ has x and y components, see for example (Sussman, 1999)

$$\begin{aligned}(L_\mu \mathbf{u})^x_{i,j} = {} & 2\Big(\mu_{i+1/2,j}(u_{i+1,j} - u_{i,j}) - \mu_{i-1/2,j}(u_{i,j} - u_{i-1,j})\Big)/\Delta x^2 \\ & + \Big(\mu_{i,j+1/2}(u_{i,j+1} - u_{i,j}) - \mu_{i,j-1/2}(u_{i,j} - u_{i,j-1})\Big)/\Delta y^2 \\ & + \big(\mu_{i,j+1/2}(v_{i+1,j+1} - v_{i-1,j+1} + v_{i+1,j} - v_{i-1,j}) \\ & - \mu_{i,j-1/2}(v_{i+1,j} - v_{i-1,j} + v_{i+1,j-1} - v_{i-1,j-1})\big)/4\Delta x \Delta y\end{aligned} \tag{11}$$

$$\begin{aligned}(L_\mu \mathbf{u})^y_{i,j} = {} & 2\big(\mu_{i,j+1/2}(v_{i,j+1} - v_{i,j}) - \mu_{i,j-1/2}(v_{i,j} - v_{i,j-1})\big)/\Delta y^2 \\ & + \big(\mu_{i+1/2,j}(v_{i+1,j} - v_{i,j}) - \mu_{i-1/2,j}(v_{i,j} - v_{i-1,j})\big)/\Delta x^2 \\ & + \big(\mu_{i+1/2,j}(u_{i+1,j+1} - u_{i+1,j-1} + u_{i,j+1} - u_{i,j-1}) \\ & - \big(\mu_{i-1/2,j}(u_{i,j+1} - u_{i,j-1} + u_{i-1,j+1} - u_{i-1,j-1})\big)/4\Delta x \Delta y\end{aligned} \tag{12}$$

where $\sigma_{i+1/2,j} = (\sigma_{i,j} + \sigma_{i+1,j})/2$ and $\mu_{i+1/2,j} = (\mu_{i,j} + \mu_{i+1,j})/2$. Note that this means that the divergence operator on the right hand side of the projection (4) is the cell-centred one (8). Although this is not consistent with the Laplacian on the left hand side, the intention is to approximate the projection that would result if the cell-centred divergence (8) and gradient (9) operators had been used to define it, i.e., if $L_\sigma = D \cdot \sigma \mathbf{G}$.

2.4 Construction of ghost cells

2.4.1 Velocities near the boundary

The domain boundaries are defined at cell edges so that $x = 0 = x_{1/2}$, $x = X = x_{I+1/2}$, $y = 0 = y_{1/2}$ and $y = Y = y_{J+1/2}$. Specified conditions for the physical velocity are defined along left, right, bottom and top edges of the domain, given by

$$\begin{aligned}&\mathbf{u}^*_{1/2,j} = \mathbf{u}^{n+1}_{b_{1/2,j}}, \quad \mathbf{u}^*_{I+1/2,j} = \mathbf{u}^{n+1}_{b_{I+1/2,j}}, \\ &\mathbf{u}^*_{i,1/2} = \mathbf{u}^{n+1}_{b_{i,1/2}}, \quad \mathbf{u}^*_{i,J+1/2} = \mathbf{u}^{n+1}_{b_{i,J+1/2}}\end{aligned} \tag{13}$$

By using averaging, we have

$$\mathbf{u}_{i+1/2,j} = (\mathbf{u}_{i,j} + \mathbf{u}_{i+1,j})/2, \quad \mathbf{u}^*_{0,j} = 2\mathbf{u}^{n+1}_{b_{1/2,j}} - \mathbf{u}^*_{1,j},$$

$$\mathbf{u}^*_{I+1,j} = 2\mathbf{u}^{n+1}_{b_{I+1/2,j}} - \mathbf{u}^*_{I,j}, \quad \mathbf{u}^*_{i,0} = 2\mathbf{u}^{n+1}_{b_{i,1/2}} - \mathbf{u}^*_{i,1},$$

and $\mathbf{u}^*_{i,J+1} = 2\mathbf{u}^{n+1}_{b_{i,J+1/2}} - \mathbf{u}^*_{i,J}$ for intermediate velocity ghost cells. Note that the viscous term also gives rise to velocity boundary terms at corner values such as $u_{1/2,\,1/2}$ which in turn require ghost cell values at $u_{0,\,0}$ which must be extrapolated from existing ghost cells.

2.4.2 Physical parameters near the boundary

The boundary conditions for the density and viscosity use zero normal gradient at the boundaries which are satisfied by

$$\begin{aligned}&\rho_{0,j} = \rho_{1,j}, \quad \rho_{I+1,j} = \rho_{I,j}, \quad \rho_{i,0} = \rho_{i,1}, \quad \rho_{i,J+1} = \rho_{i,J}, \\ &\mu_{0,j} = \mu_{1,j}, \quad \mu_{I+1,j} = \mu_{I,j}, \quad \mu_{i,0} = \mu_{i,1}, \quad \mu_{i,J+1} = \mu_{i,J}\end{aligned}$$

which immediately defines the inverse density BC's, for example, $\sigma_{0,j} = 1/\rho_{0,j}$, and is also used for the volume fraction C.

2.4.3 The gauge variable near the boundary

A consequence of the setting of the boundary condition for the intermediate velocity to that of the physical velocity BC is that the boundary condition for the normal gradient of the gauge variable that is consistent with the velocity update (5), i.e., $\mathbf{n} \cdot \nabla \phi^{n+1} = \mathbf{n} \cdot \rho^{n+1/2}(\mathbf{u}^* - \mathbf{u}^{n+1})/\Delta t = 0$, implies a homogeneous Neumann BC for ϕ. That is, along each boundary we have the BC seen in (4).

$$(G\phi)^x_{1/2,j} = (G\phi)^x_{I+1/2,j} = 0, \quad (G\phi)^y_{i,1/2} = (G\phi)^y_{i,J+1/2} = 0 \tag{14}$$

A simple constant extrapolation (Bell and Marcus, 1992) gives $\phi_{0,j}=\phi_{1,j}$, $\phi_{I+1,j}=\phi_{I,j}$, $\phi_{i,0}=\phi_{i,1}$ and $\phi_{i,J+1}=\phi_{i,J}$ for the gauge variable ghost cells.

2.4.4 The pressure near the boundary: the constant density case

Provided the density and viscosity are constant within the entire domain, the so-called constant density case, equation (6) allows a direct pressure update in $\overline{\Omega}$, so that

$$p^{n+1/2}=p^{n-1/2}+\phi^{n+1}-\frac{\Delta t}{2Re}\nabla^2\phi^{n+1}$$

However, no boundary conditions are ordinarily defined for the pressure which must then be extrapolated from cells internal to the domain.
A second-order linear extrapolation of the gradient (Rider et al., 1998) evaluated at the boundary is used, i.e.,

$$\begin{aligned}
&(Gp)^x_{1/2,j}=2(Gp)^x_{3/2,j}-(Gp)^x_{5/2,j},\\
&(Gp)^x_{I+1/2,j}=2(Gp)^x_{I-1/2,j}-(Gp)^x_{I-3/2,j},\\
&(Gp)^y_{i,1/2}=2(Gp)^y_{i,3/2}-(Gp)^y_{i,5/2},\\
&(Gp)^y_{i,J+1/2}=2(Gp)^y_{i,J-1/2}-(Gp)^y_{i,J-3/2}
\end{aligned}\qquad(15)$$

giving a quadratic extrapolation of the pressure
$p_{0,j}=p_{3,j}-3(p_{2,j}-p_{1,j})$,
$p_{I+1,j}=p_{I-2,j}-3(p_{I-1,j}-p_{I,j})$ and
$p_{i,0}=p_{i,3}-3(p_{i,2}-p_{i,1})$,
$p_{i,J+1}=p_{i,J-2}-3(p_{i,J-1}-p_{i,J})$, for the pressure ghost cells.

2.4.5 The pressure near the boundary: the variable density case

For the variable density case, equation (6) now only allows the gradient of pressure to be updated since the gradient operator does not commute with the Laplacian as it did in the constant density case, $L\mu(\sigma\nabla\phi)=\sigma\mu\nabla^2\nabla\phi=\sigma\mu\nabla\nabla^2\phi$. Then for the variable density case, in $\overline{\Omega}$

$$\nabla p^{n+1/2}=\nabla p^{n-1/2}+\nabla\phi^{n+1}-\frac{\Delta t}{2Re}L_\mu^{n+1/2}\left(\sigma^{n+1/2}\nabla\phi^{n+1}\right)$$

When this is true, the gradient of pressure does not require any boundary conditions. However, the presence of homogeneous Neumann boundary conditions on the gauge variable, $\mathbf{u}\cdot\nabla\phi|_{\partial\Omega}=0$ used in (4), effects the value of ∇p near the boundary through its influence on $L_\mu(\sigma\nabla\phi)$ there. Projection methods naturally define the normal component of velocity at the boundary, $\mathbf{n}\cdot\mathbf{u}^{n+1}|_{\partial\Omega}=\mathbf{n}\cdot\mathbf{u}_b^{n+1}$; however, the tangential component is not guaranteed to be homogeneous since $\boldsymbol{\tau}\cdot\mathbf{u}^{n+1}|_{\partial\Omega}=\boldsymbol{\tau}\cdot(\mathbf{u}_b^{n+1}-\Delta t\sigma^{n+1/2}\nabla\phi^{n+1}|_{\partial\Omega})$ which is in error by $\Delta t\sigma^{n+1/2}\boldsymbol{\tau}\cdot\nabla\phi^{n+1}|_{\partial\Omega}$. In this paper, we adopt the procedure of Brown, Cortez and Minion (2001) and simply reset the component to the correct value at the end of the time step. This is also carried forward to the boundary conditions for $\sigma\nabla\phi$ when evaluating the pressure gradient.
Evaluated at the boundary, the normal and tangential boundary conditions on $\nabla\phi$ give the conditions shown earlier in (14) which, upon using a vector defined by $(\psi^x,\psi^y)^T=\sigma\nabla\phi$, gives

$$\begin{aligned}
&\psi^{x,y}_{0,j}=-\psi^{x,y}_{1,j}, \qquad \psi^{x,y}_{I+1,j}=-\psi^{x,y}_{I,j},\\
&\psi^{x,y}_{i,0}=-\psi^{x,y}_{i,1}, \qquad \psi^{x,y}_{i,J+1}=-\psi^{x,y}_{i,J}
\end{aligned}\qquad(16)$$

and the corner ghost cell values

$$\begin{aligned}
&\psi^{x,y}_{0,J+1}=\psi^{x,y}_{1,J}, \qquad \psi^{x,y}_{I+1,0}=\psi^{x,y}_{I,1},\\
&\psi^{x,y}_{0,0}=\psi^{x,y}_{1,1}, \qquad \psi^{x,y}_{I+1,J+1}=\psi^{x,y}_{I,J}
\end{aligned}\qquad(17)$$

which are used in the evaluation of the vector $\left(\left(L_\mu\sigma\nabla\phi\right)^x_{i,j},\left(L_\mu\sigma\nabla\phi\right)^y_{i,j}\right)^T$ near the boundary. See Appendix for details.

2.4.6 Initialisation of the pressure field

The exact test already possesses solenoidal initial conditions; however, the two droplet impact tests do not. They are both initiated by a non-zero initial vertical velocity for the impacting droplet with the surrounding media at zero velocity, collectively called $\tilde{\mathbf{u}}^0$. This represents a discontinuity in velocity as well as the already present discontinuity in density and viscosity, and must be alleviated. A global solenoidal initial velocity field is essential for starting the calculation without carrying forward any initial errors. If there are any external forces acting, they must also be included. The velocity field may be made divergence free and a pressure field be initiated by projecting the discontinuous field using the projection (4) developed earlier. That is, given the discontinuous non-solenoidal field $\tilde{\mathbf{u}}^0$ and zero initial pressure $\tilde{p}^0$, we add in the external force in the present case gravity and define a new initial velocity field $\mathbf{u}^{*,0}$

$$\mathbf{u}^{*,0}=\tilde{\mathbf{u}}^0-\frac{\Delta t^0}{Fr}\hat{\mathbf{j}}\qquad(18)$$

with a stability constraint implied by the force acting so that $\Delta t^0<\sqrt{Fr\ min(\Delta x,\Delta y)}$. By the

Hodge decomposition, any vector $\mathbf{u}^{*,0}$ may be decomposed into a div-free, $\mathbf{u}^0$, and a curl-free part, $\mathbf{G}\varphi$, i.e.,

$$\mathbf{u}^{*,0} = \mathbf{u}^0 - \Delta t^0 \sigma^0 \mathbf{G}\varphi \tag{19}$$

and by observing the fact that the new div-free velocity $\mathbf{u}^0$ must also obey the boundary condition $\mathbf{u}^0|_{\partial\Omega} = \mathbf{u}_b$, we solve the Neumann problem

$$L_\sigma \varphi = \frac{1}{\Delta t^0} D \cdot \mathbf{u}^{*,0} \tag{20}$$

subject to the consistent boundary condition

$$\mathbf{n} \cdot \mathbf{G}\varphi|_{\partial\Omega} = \frac{\rho^0}{\Delta t^0}\left(\mathbf{n} \cdot \mathbf{u}^{*,0}|_{\partial\Omega} - \mathbf{n} \cdot \mathbf{u}_b\right) \tag{21}$$

if $\mathbf{u}^{*,0}|_{\partial\Omega} \equiv \mathbf{u}_b$, we have homogeneous Neumann conditions; otherwise, (21) must be used. With the use of (19), the div-free velocity becomes

$$\mathbf{u}^0 = \mathbf{u}^{*,0} - \Delta t \sigma^0 \mathbf{G}\varphi \tag{22}$$

and the pressure initialisation is

$$p^0 = \varphi \tag{23}$$

This will be adopted for the start of computation, i.e., for $\mathbf{u}^0$ and $p^{-1/2}$.

2.5 The convective term

The Godunov method was developed to provide a robust treatment of the non-linearities in the NS equations at high Reynolds number while adjusting to rapid changes in fluid properties across interfaces without producing severe instabilities. It is a high resolution upwinding scheme and was originally designed to deal with the change in physical properties across shocks. It is constructed in a predictor-corrector fashion with several stages in each. The non-linear convective term in the NS equations is approximated here using a second-order Godunov method which has been previously detailed in both Puckett et al. (1997) and Sussman et al. (1999) and will not be repeated here.

2.6 Time step restrictions

Linear stability requires restrictions on the time step taking into account the stability requirements for the Godunov method, i.e., the CFL condition. As well, an extra constraint is needed for the inclusion of the viscous term, which was not used in the construction of the transverse derivatives and a restriction arising from the source terms (Sussman et al., 1999). So the final choice for the minimum time-step is the simultaneous minimisation of all of these with a safety factor of a half added.

$$\Delta t < \frac{1}{2} \min_{\substack{1 \le i \le I \\ 1 \le j \le J}} \left(\frac{\Delta x}{|u^n_{i,j}|}, \frac{\Delta y}{|v^n_{i,j}|}, \frac{3Re}{14} \frac{\rho^n_{i,j}(\Delta x^2, \Delta y^2)}{\mu^n_{i,j}}, \sqrt{\frac{2(\Delta x, \Delta y)}{F^n_{i,j}}} \right) \tag{24}$$

where

$$F^n_{i,j} = \left| -\sigma^n_{i,j} (\mathbf{G}p)^{n-1/2}_{i,j} + \frac{\sigma^n_{i,j}}{Re} (L_\mu \mathbf{u})^n_{i,j} - \frac{1}{Fr}\hat{\mathbf{j}} \right|$$

2.7 The smoothing of discontinuities

The smoothing of discontinuities in the physical properties, such as the density and viscosity, can be achieved by concentrating on the volume fraction as it is used to update both. Typically, the volume fraction is smoothed by convolution with a kernel so that the smoothed volume fraction $\widetilde{C}(\mathbf{x})$ is

$$\widetilde{C}(\mathbf{x}) = \int_{\Omega_K} C(\mathbf{x}')\mathbf{K}(r;\varepsilon)\, d\mathbf{x}' \tag{25}$$

where $r = |\mathbf{x}' - \mathbf{x}|$ and $\mathbf{K}(r;\varepsilon)$ is an integration kernel which becomes the surface delta function δ_S as $\varepsilon \to 0$. Here, Ω_K is the compact support of the kernel, or the points for which $\mathbf{K}(r;\varepsilon) \neq 0$ and ε is the size of the support. Ideally, a kernel should possess compact support, be monotonically decreasing with respect to r, be radially symmetric, be sufficiently smooth, be a good representation of the delta function as $|\Omega_K| \to 0$ and possess the normalisation property $\int_{\Omega K} \mathbf{K}(r;\varepsilon) d\mathbf{x} = 1$ (Williams, Kothe and Puckett, 1998). In the present case, we choose the eighth degree polynomial kernel because it closely matches all of the properties listed above, being superior to say the Nordmark kernel which is not monotonically decreasing and tends to produce highly singular oscillations as $|\Omega| \to 0$ (Williams, Kothe and Puckett, 1998). The kernel is defined by

$$\mathbf{K}_8(r;\varepsilon) = \begin{cases} \frac{5}{\pi\varepsilon^2}[1-(r/\varepsilon)^2]^4 & \text{if } r < \varepsilon \\ 0 & \text{otherwise} \end{cases}$$

At any point (x, y) in the domain the volume fraction there may be convolved by considering a circle of radius ε, $(x'-x)^2+(y'-y)^2 = \varepsilon^2$, the support of the kernel, and integrating (25). Evaluation of this integral requires a little care when examining points near the boundary, within

one cell width, as the support of the kernel may overlap the boundary and therefore requiring ghost cells as part of the evaluation. A finite number of ghost cells indicate that the smoothing length ε should not exceed two to three cell widths.

3. DYNAMICAL TESTS AND DISCUSSION

3.1 Multiphase fluid dynamics

The validation of the GMPPS, developed in this and the previous papers (Bierbrauer and Zhu, 2007a), involves tests of the accuracy of the individual Godunov approximate projection scheme and a test of the stability of such a scheme. The first test studies the unsteady constant density incompressible NS equations. The known time-dependent exact solution of the incompressible NS equation of Chorin (1968) is used to construct error measures and order of accuracy estimates.

3.2 Exact solution of the NS equations

To test the GMPPS, we need to conduct numerical experiments and ideally compare results with those generated from an exact solution. One such exact solution for the time-dependent non-dimensionalised Navier-Stokes equations for the viscous flow of incompressible fluids with constant density and viscosity, $\rho = \rho_1 = \rho_2$ and $\mu = \mu_1 = \mu_2$, is given by (Chorin, 1968)

$$\begin{aligned} u(x,y,t) &= -\cos\pi x \sin\pi y \exp(-2\pi^2 t/Re) \\ v(x,y,t) &= \sin\pi x \cos\pi y \exp(-2\pi^2 t/Re) \\ p(x,y,t) &= -\frac{1}{4}(\cos 2\pi x + \cos 2\pi y) \exp(-4\pi^2 t/Re) \end{aligned} \tag{26}$$

For a unit square domain $(0,1)\times(0,1)$, a set of appropriate boundary and initial conditions can be constructed from this solution, at the boundaries $x = 0, 1$, $y = 0, 1$ and $t = 0$, respectively. Clearly, the flow velocity decays to zero as $t \to \infty$. For very small Reynolds numbers, it also decays quickly to zero except for small times when $t/Re \simeq 0$. For large Reynolds numbers, the initial state is maintained until $t/Re >> 0$.

With an exact solution, absolute errors can be easily evaluated by defining the error $E^n_{i,j} = F^n_{i,j} - f(x_i, y_j, t^n)$, where $f(x_i, y_j, t^n)$ is the true solution for u, v or p evaluated at the grid point (x_i, y_j) at time t^n and $F^n_{i,j}$ is the numerical solution. We contrast the L^1 error norm $\| E^n \|_1 = \sum_{j=1}^{J} \sum_{i=1}^{I} | E^n_{i,j} | \Delta x \Delta y$ with the max norm $\| E^n \|_\infty = \max_{\substack{1 \le i \le I \\ 1 \le j \le J}} | E^n_{i,j} |$. Relative errors are avoided given the fact that the true solution is zero in particular regions, e.g., u is zero along the lines $x = 1/2$, $y = 0, 1$.

The solutions (26) exponentially decay over time with a half life $T = Re \ln 2 / 2\pi^2 \simeq 0.0351\, Re$. One way of choosing a stopping criterion, in order to evaluate numerical errors, is to stop the computation within a number of half-lives. Here we choose a single half life, e.g., for $Re = 100$, the stopping criterion is $t = T = 3.5$ time units.

Both the exact and multiphase problems must solve two sets of linear systems, one for the Poisson equation and another for the momentum equation. These systems are solved iteratively using the Bi-CGSTAB linear solver. Here the smallest matrix elements are limited by a threshold value of 10^{-7}; all smaller values were taken as zero. This allows the use of sparse storage mode where only non-zero entries are used to solve the problem consequently speeding up calculation times. Iterations were stopped for convergence tolerances determined by the grid used, e.g., for 16×16 grid $tol \simeq 10^{-2}$. The time step was always half of that required by linear stability analysis (24) except where this did not allow time stepping to coincide with the half-life and the time step was consequently lowered.

The order of accuracy of the method is calculated as was done in Bierbrauer and Zhu (2007a) by obtaining the numerical solution $f^{(n)}_h$ on a grid $h = \Delta x = \Delta y$ compared with the exact solution on the same grid $f^{(e)}_h$ so that the error in the L^1 norm is $e(h) = \| f^{(n)}_h - f^{(e)}_h \|_1$ on that grid. Then, by repeating the procedure on a refined grid with half of the grid spacing, $h/2$, we get $e(h/2)$. The order of accuracy is now given by $\log_2[e(h)/e(h/2)]$.

Another way of analysing the solution quality is through the decay of kinetic energy in the problem. The integral of kinetic energy is in fact an invariant of solutions to the Euler equations and so its variation in time for computed solutions is a gross measure of the accuracy of the method (Bell Colella and Glaz, 1989). That is,

$$K = \int_0^1 \int_0^1 \mathbf{u} \cdot \mathbf{u}\, dxdy = \frac{1}{2} \exp(-4\pi^2 t/Re) \tag{27}$$

The correct physical behavior of a method may be judged by how it dissipates kinetic energy and in this case, it should decay exponentially.

Note that the constant density case does not require the use of particles to track fluid phases. However, the method is designed to be used in either case and particles are used to update density and viscosity. In addition, no smoothing of discontinuities is necessary. Filters are also not used.

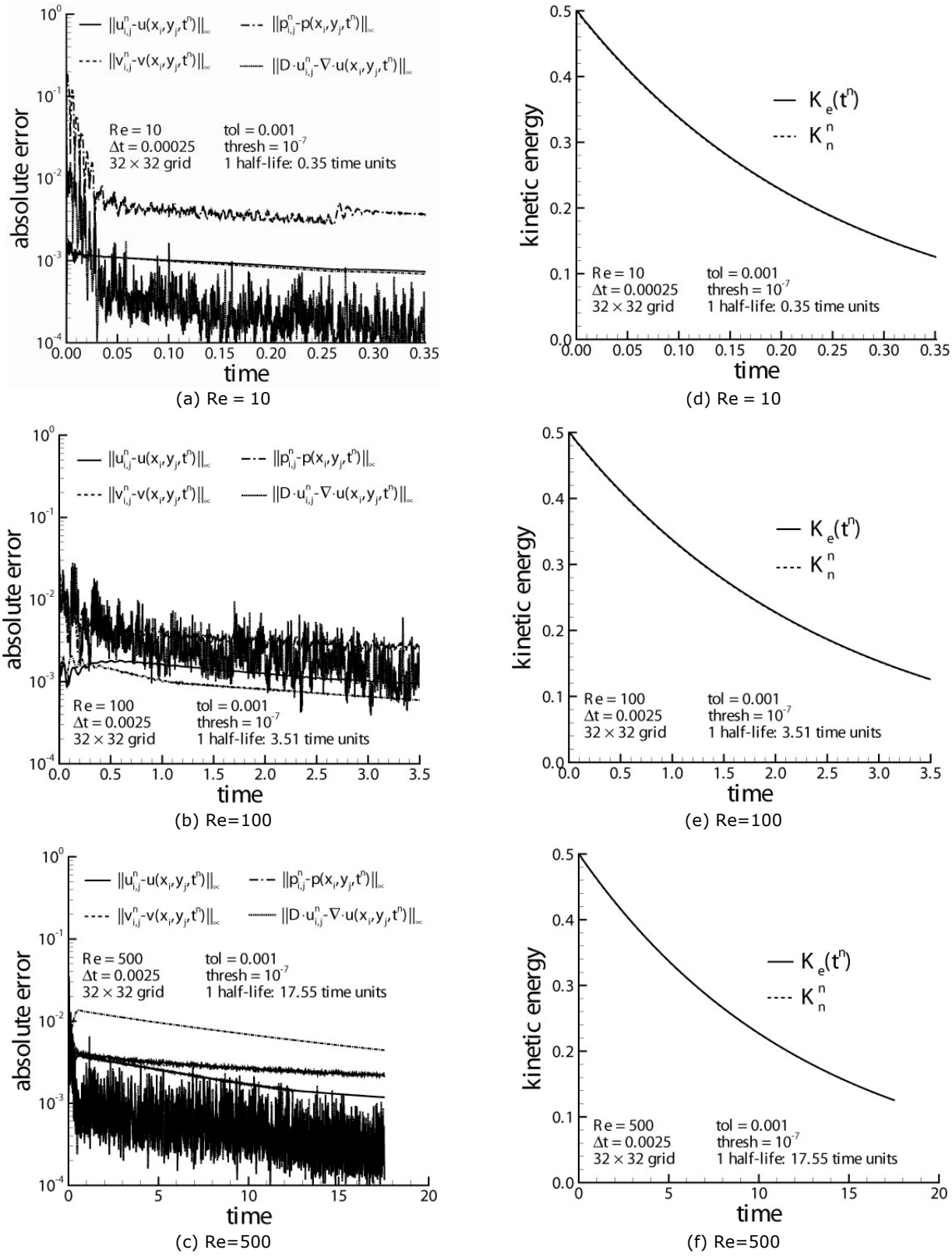

Fig. 1 Absolute errors and kinetic energy decay for velocity components, pressure and divergence over time for Reynolds numbers of $Re = 10$, 100, 500 on a 32×32 grid with a quadratic pressure gradient.

3.2.1 Results: absolute errors for a single half-life

Fig. 1 shows the variation in absolute errors over a single half-life on a 32×32 grid, in the max norm, for the two velocity components—pressure and divergence of velocity, as well as the decay of kinetic energy for the three Reynolds numbers $Re=10, 100, 500$. Clearly, the velocity errors for the $Re=10$ and $Re=100$ cases remain within the expected second-order error, a multiple of $h^2\simeq10^{-3}$, whereas the $Re=500$ case shows velocity errors a little higher than these. In all cases, except for $Re=500$, the pressure error is larger than the velocity errors. Given the problems associated with approximate projection methods near no-slip boundaries, this may be expected. In addition, although the fluctuations in the divergence error are larger than the minor oscillations in the pressure error, they nevertheless are still very small. For all cases the kinetic energy decays in a physical way.

3.2.2 Order of accuracy

The order of accuracy is calculated as described in Section 3 using a pair of grids: the first pair comparing a 16×16 grid as the h case to a 32×32 grid as the $h/2$ case and the second pair a 32×32 grid for the h case and 64×64 grid for the $h/2$ case. The absolute errors $E(h)$, $E(h/2)$ and the corresponding order of accuracy is obtained. The results for each pair are shown for the three different Reynolds numbers $Re=10, 100, 500$, for both the L^1 and max norms, in Tables 1 and 2. The tables show the absolute errors for the horizontal and vertical components of velocity as well as the pressure. For each Reynolds number a single half-life was used as the endpoint for each calculation: for $Re=10$, $T=0.351$, $Re=100$, $T=3.51$ and for $Re=500$, $T=17.55$ time units.

Table 1 Absolute errors in the L^1 norm with the corresponding order of accuracy for 16^2-32^2 and 32^2-64^2 grids for three different Reynolds numbers.

		16^2-32^2			32^2-64^2		
Re		*e(h)*	*e(h/2)*	Order	*e(h)*	*e(h/2)*	Order
10	*u*	1.76E-03	2.93E-04	2.59	2.93E-04	6.24E-05	2.23
	v	1.71E-03	2.89E-04	2.56	2.89E-04	5.95E-05	2.28
	p	1.62E-02	1.64E-03	3.30	1.64E-03	3.09E-04	2.41
100	*u*	1.43E-03	3.45E-04	2.05	3.45E-04	7.66E-05	2.17
	v	1.20E-03	2.42E-04	2.31	2.42E-04	6.15E-05	1.98
	p	2.90E-03	7.02E-04	2.05	7.02E-04	1.39E-04	2.34
500	*u*	2.86E-03	7.43E-04	1.94	7.43E-04	1.53E-04	2.28
	v	3.00E-03	6.25E-04	2.20	6.25E-04	8.54E-05	2.87
	p	4.97E-03	8.15E-04	2.61	8.15E-04	2.14E-04	1.93

Studying Table 1, in the L^1 norm, shows that the method is generally second-order. A similar set of results are seen in Table 2 for the max norm.

Table 2 Absolute errors in the max norm with the corresponding order of accuracy for 16^2-32^2 and 32^2-64^2 grids for three different Reynolds numbers.

		16^2-32^2			32^2-64^2		
Re		*e(h)*	*e(h/2)*	Order	*e(h)*	*e(h/2)*	Order
10	*u*	4.71E-03	8.06E-04	2.55	8.06E-04	1.54E-04	2.39
	v	4.85E-03	7.29E-04	2.73	7.29E-04	1.51E-04	2.27
	p	4.43E-02	4.81E-03	3.20	4.81E-03	9.31E-04	2.37
100	*u*	4.14E-03	1.22E-03	1.76	1.22E-03	2.56E-04	2.25
	v	6.56E-03	5.82E-04	3.49	5.82E-04	1.50E-04	1.96
	p	8.43E-03	2.78E-03	1.60	2.78E-03	6.81E-04	2.03
500	*u*	1.20E-02	2.08E-03	2.53	2.08E-03	5.44E-04	1.93
	v	2.13E-02	8.82E-03	1.27	8.82E-03	1.13E-03	2.96
	p	1.32E-02	2.52E-03	2.39	2.52E-03	8.74E-04	1.53

3.3 The droplet-solid impact problem

The DSIP and DLIP are both two-phase fluid flow problems and require the use of marker-particle tracking, as described in Bierbrauer and Zhu (2007a). The data of this previous work suggested using 16 particles per cell which is used throughout the present study. In addition, minimal smoothing is used with a smoothing length of two. Throughout, only the velocity filter of Rider (1994) is used to ensure an approximate solenoidal velocity field.

Experiments have shown that the DSIP is governed by the droplet's inertial properties, impact kinetic energy (velocity and Reynolds number), viscosity and surface tension in that order. There are several main phases of droplet impact on a solid surface: the kinematic phase controlled by the impact velocity U_0 and droplet diameter D up to a time $t^*=tU_0/D<0.1$. This is followed by the spreading phase: where a thin-film lamella spreads outwards from the edges of the droplet impact zone bounded by a rim. The kinetic energy is now dissipated into potential energy by surface tension and viscosity (Rioboo, Marengo and Tropea, 2002) as well as some internal kinetic energy of the droplet (Clanet et al., 2004). Surface tension effects decelerate the lamella spread after $t^*>\sqrt{We}$. Since the present simulation does not possess surface tension, the calculation is halted before surface tension effects become significant.

The DSIP itself is solved in a domain with $X=8$, $Y=3$ with solid wall boundaries, $\mathbf{u}|_{\partial\Omega}=\mathbf{0}$, having zero initial pressure and velocity except for the

droplet which possesses an initial impact velocity of U_0. Calculations are performed using a 80×30 grid and, using the definition of Josserand and Zaleski (2003), the spreading radius is defined as the radius of the point where the velocity of the fluid was maximal at a given time. The droplet is initially assumed to be spherical and is positioned just contacting the surface at the start of the calculation. The velocity field must be made divergence free and so requires an initial projection on the discontinuous initial data.

3.3.1 Results: spread factor

One way to measure the present method's ability to simulate droplet impacts on solid surfaces is through the time-dependent radial spread factor *r*, here non-dimensionalised with respect to the length scale *D*, so that $r^* = r/D$. This measures the time-dependent radial distance $r^*(t)$ from the impact point that the droplet has spread across the surface as a function of dimensionless time, $t^* = tU_0/D$. A comparison of numerical results, shown as solid, dashed, dash-dotted and dotted lines, and the experimental data, shown as square, circular and cross symbols, of Rioboo, Marengo and Tropea (2002) for the impact of droplets of water, glycerin and silicon oil is shown in Fig. 2. The physical data used here is summarised in Table 3.

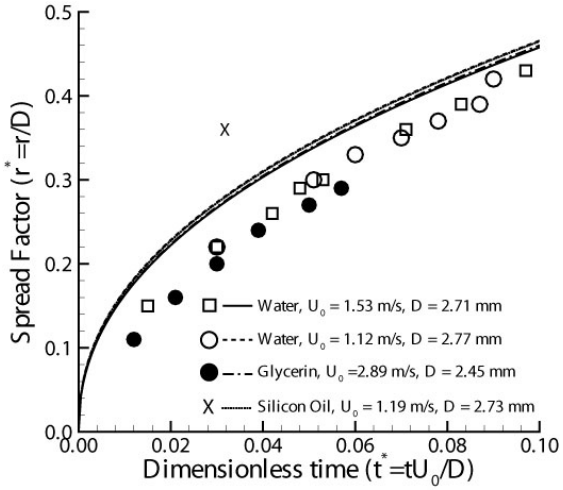

Fig. 2 Radial spread factor, $r^* = r/D$, as a function of dimensionless time, $t^* = tU_0/D$, for numerical simulation and the experimental results of Rioboo, Marengo and Tropea (2002).

Table 3 Physical data used in the numerical solution for various droplet fluids in the DSIP.

Fluid	ρ_1 (kg/m^3)	μ_1 (kg/ms)	D (mm)	U_0 (m/s)	*We*
water	1000	0.001	2.77	1.12	48
water	1000	0.001	2.71	1.53	88
silicon oil	966	0.019	2.73	1.19	178
glycerin	1261	1.410	2.45	2.89	396

The density and viscosity of air used in the solution are $1\ \text{kg/m}^3$ and 1.8×10^{-5} kg/ms, respectively.

The numerical results compare quite well with the experimental values. Numerical solution curves are clearly of the same shape as the experimental data although they are always larger than the experimental data, for example, at $t^* = 0.02$ the difference in $\Delta r^* \simeq 0.04$ whereas at $t^* = 0.10$, it is $\Delta r^* \simeq 0.02$. The only case that does not follow this trend is the silicon oil result although it is difficult to judge given that there is only one data point. The Weber numbers in each case are $We = 48, 87, 178, 396$ for the two water, silicon oil and glycerine droplets respectively so we do not expect surface tension effects to become severe until $t^* \simeq 7$, well out of the range of the current timescale. However, surface effects are expected to hinder radial spread even at earlier times. Given that the numerical results are computed without surface tension, it is to be expected that these results should show a larger spread factor as compared to the experimental data which is always under the restrictive influence of surface forces. It should be noted that while spreading occurs, the kinetic energy of the drop is being converted to surface energy; this is what eventually stops the spreading and is absent from the numerical solution. While the solid walls at the left and right boundaries of the domain are not expected to influence the results at such short timescales, it is to be expected that more distant walls will improve on these results. It is also expected that finer grids will improve these results.

3.4 The droplet-liquid impact problem

The DLIP may be categorised into two kinds of impact: droplet impact onto shallow layers of the same liquid, when the layer thickness, *h*, is comparable to the diameter of the droplet, i.e., with $H = h/D \leq 1$; and droplet impact onto deep liquid layers or pools of the same liquid, where the solid wall at the layer bottom has little

influence on the impact taking place at the layer top, i.e., when $H >> 1$.

For the case of deep layers, after the impacting droplet has collided with the free surface of the pool liquid, a film of liquid is expelled upward and outward from the edge of the colliding region (Liow, 2001). A rim is formed at the top of the film and a cavity develops below the level of the original liquid surface (Fedorchenko and Wang, 2004). The cavity develops about the vertical line of symmetry about the point of impact which allows the cavity depth to be measured. In this paper, the measurement was performed along this line of symmetry as measured from the original horizontal surface of the liquid. Calculations were performed in a domain with $X = 20$, $Y = 10$ using a 60×30 grid with solid wall boundaries. The calculations were terminated before surface tension effects became predominant. Once again, the problem requires pre-projection before the start of computations with the same initial and boundary conditions as the DSIP except with the droplet now initially touching the layer instead of the solid surface.

For the case of shallow liquid layers, the calculation was performed in a 80×40 grid and with a domain of size $X = 4$, $Y = 2$ in order to capture the shallow layer realistically. Given the limited number of time steps, the influence of the walls on the results should be minimal. Again, calculations were initiated by placing the impacting spherical droplet, impact velocity U_0, just in contact with the layer with the rest of the domain at zero initial velocity and pressure. An initial projection was required to obtain a div-free velocity field. Again, the spreading radius was used as a quantitative measure of the impact process and was calculated as part of the problem solution and defined in the same way as for the DSIP.

3.4.1 Results: deep layer-cavity depth

Experimental data for deep layers is often in the form of graphs of cavity depth, $R^* = R/D$, as a function of time, $t^* = tU_0/D$. In the numerical simulation, we use $H = 5$ to approximate a deep layer. The depth of the cavity generated is measured from the initial quiescent layer position along the symmetry axis. Fig. 3 shows how the cavity depth varies over time. Numerical results are shown as open symbols and experimental ones as filled symbols. Both the experimental and numerical data are shown as a discrete set of points through which polynomial curves have been fitted. Here the experimental data of Engel (1967), Morton, Rudman and Liow (2000) and the 16G result of Liow (2001) are used. See Table 4 for details.

Fig. 3 shows that the numerical solutions for the Morton case, represented in triangles, compare very well with the experimental data. Any surface force effects are expected to have a greater influence in this case given its smaller impact velocity although such effects are not expected to become significant until $t^* \simeq 15$. The smaller impact velocity also means that the influence of a finite layer depth will be minimal in this case. Note that the surface tension effects are considerably less important for the other two cases: not until $t^* \simeq 27$ (Liow, circles) and $t^* \simeq 141$ (Engel, squares). The difference between the numerical and experimental results for both of these cases must then be mainly due to a finite layer depth which alters the impact characteristics as the droplet starts to penetrate deeper into the layer. This is shown in the slightly diverging tendencies of both the 16G and Engel results. As expected, the Engel result shows the greatest difference since its impact velocity is the largest.

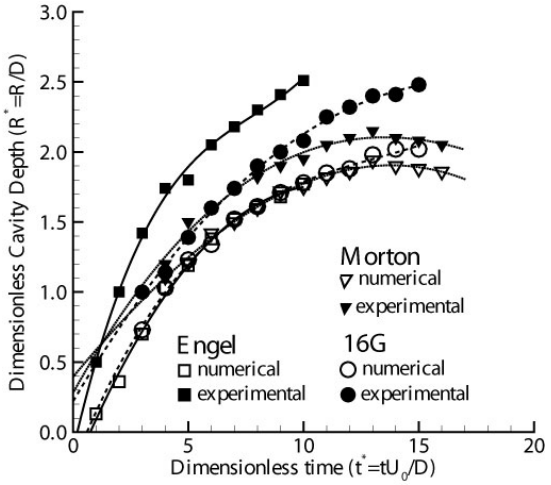

Fig. 3 Dimensionless cavity depth, $R^* = R/D$, as a function of dimensionless time, $t^* = tU_0/D$, for the numerical simulation and the experimental results of Engel (1967), Morton, Rudman and Liow (2000) and Liow (16G) (2001).

Table 4 Water droplet diameters and impact velocities as used in the experiments of Engel, Morton and Liow for the deep-layer DLIP.

Result	D (mm)	U_0 (m/s)	We	Fr	Re
Morton (2000)	2.90	2.50	251	219	7250
Liow (2001)	3.77	3.78	748	386	14250
Engel (1967)	4.55	17.61	19597	6947	80125

3.4.2 Results: shallow layer-spread factor

As for the solid-liquid impact case, it is possible to calculate, as part of the numerical solution for a shallow layer droplet impact, the spreading radius (r) where the magnitude of velocity is a maximum at a given instant. Fig. 4 plots the spread factor, $r^* = r/D$, with (non-dimensional) time $t^* = tU_0/D$, obtained for the impact of a 2 mm water-like droplet onto a shallow layer of the same liquid of depth $h = 0.3$ mm ($H = 0.15$). See Table 5 for details. Numerical results are shown in a solid line (overlapping the dash-dotted and dotted lines) and experimental ones in a dashed line. This follows the tests of Josserand and Zaleski (2003) whose results are shown in a straight line $r^* = \sqrt{t^*}$, which is a well established experimental result for the impact radius at small times (Yarin and Weiss, 1995).

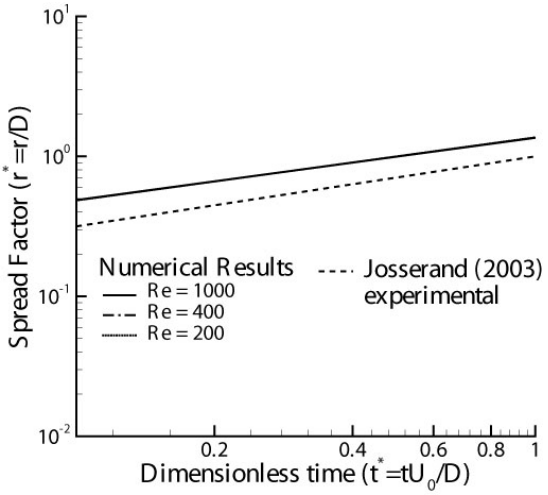

Fig. 4 Log-log graph of radial spread factor, $r^* = r/D$, as a function of dimensionless time, $t^* = tU_0/D$, for numerical simulations and the experimental results of Josserand and Zaleski (2003).

Table 5 Physical data used in the numerical solution for various droplet fluids in the shallow-layer DLIP.

ρ_1 (kg/m^3)	μ_1 (kg/ms)	ρ_2 (kg/m^3)	μ_2 (kg/ms)	D (mm)	U_0 (m/s)	h (mm)
1000	0.02	2	0.0005	2.00	10	0.3
1000	0.05	2	0.0005	2.00	10	0.3
1000	0.10	2	0.0005	2.00	10	0.3

Note that for this set of physical data (Table 5 refers), the results are expected to lie close together. This is shown by the fact that the numerical results for the various Reynolds numbers overlap each other.

Displayed in Fig. 4 is a comparison of the experimentally obtained spread factor $r^* = \sqrt{t^*}$ with the numerical results. Clearly, the numerical simulation has captured the variation of the spread factor as a function of time (in this case, the relationship appears to be linear as the lines are all straight lines). The two results appear to be in good agreement. Our numerically calculated spread factor is also independent of the Reynolds number (droplet viscosity), which is in agreement with the experimental result. In this case, $We = 8000$ so that surface tension should play little role over the times considered. Most likely, it is the very small domain size used which affected the results through the solid wall boundaries.

4. CONCLUSIONS

In this paper, the GMPPS method has been adopted to solve the incompressible Navier-Stokes equations both for the constant density and variable density cases. The constant density case, with a given exact solution, was used to obtain accurate error measures. The variable density version is used to study multiphase flow problems including the impact of a fluid droplet onto solid and liquid surfaces.

We have implemented the pressure corrections of Brown, Cortez and Minion (2001), in an approximate Godunov projection method, for the case of variable density multiphase flows and confirmed the method's ability to accurately and robustly solve multiphase flow problems.

The computational results demonstrate that the GMPPS formulation can accurately solve the NS equations up to second-order accuracy. Given that the resolution of the current computational work is limited by the available computational resources, the results presented here clearly demonstrate that the GMPPS formulation has a great potential to be used to solve the droplet-solid and droplet-liquid impact problems. It is expected that an extension of the current method to allow outflow boundaries at the sides and top of the domain will improve the results significantly. The addition of surface tension would also allow the study of more varied multiphase flow problems.

REFERENCES

1. Almgren AS, Bell JB, Crutchfield WY (2000). Approximate Projection Methods: Part I, Inviscid Analysis. *SIAM J. Sci. Compu*t. 22:1139–1159.
2. Almgren AS, Bell JB, Szymczak WG (1996). A Numerical Method for the Incompressible Navier-Stokes Equations Based on an Approximate Projection. *SIAM J. Sci. Comput*. 17:358–369.
3. Anderson DM, McFadden GB, Wheeler AA (1998). Diffuse-Interface Methods in Fluid Mechanics. *Ann. Rev. Fluid Mech*. 30:139–165.
4. Bell JB, Colella P, Glaz HM (1989). A Second-Order Projection Method for the Incompressible Navier-Stokes Equations. *J. Comput. Phys*. 85:257–283.
5. Bell JB, Marcus DL (1992). A Second-Order Projection Method for Variable-Density Flows. *J. Comput. Phys*. 101:334–348.
6. Bierbrauer F (2004). *Mathematical Modelling of Water-Droplet Impact on Hot Galvanised Steel Surfaces*. PhD Thesis, University of Wollongong, Australia.
7. Bierbrauer F, Zhu S-P (2007a). A Numerical Model for Multiphase Flow Based on the GMPPS Formulation, Part I: Kinematics. *Comput. Fluids* 36:1199–1212.
8. Bierbrauer F, Zhu S-P (2007b). A Solenoidal Initial Condition for the Numerical Solution of the Navier-Stokes Equations for Two-Phase Incompressible Flow. *Comput. Model in Eng. Sci*. 19:1–22.
9. Brackbill JU, Kothe DB, Zemach A (1992). A Continuum Model for Modeling Surface Tension. *J. Comput. Phys*. 100:335–354.
10. Brackbill JU, Ruppel HM (1986). Flip: A Method for Adaptively Zoned, Particle-In-Cell Calculations of Fluid Flows in Two Dimensions. *J. Comput. Phys.* 65(2):314–343.
11. Brown DL, Cortez R, Minion ML (2001). Accurate Projection Methods for the Incompressible Navier-Stokes Equations. *J. Comput. Phys*. 168:464–499.
12. Chorin AJ (1968). Numerical Solution of the Navier-Stokes Equations. *Math. Comput*. 22:745–762.
13. Clanet C, Beguin C, Richard D, Quere D (2004). Maximal Deformation of an Impacting Drop. *J. Fluid Mech*. 517:199–208.
14. Colagrossi A, Landrini M (2003). Numerical Simulation of Interfacial Flows by Smoothed Particle Hydrodynamics. *J. Comput. Phys*. 191:448–475.
15. Colella P (1990). A Multidimensional Second Order Godunov Scheme for Conservation Laws. *J. Comput. Phys.* 87:171–200.
16. Engel OG (1967). Initial Pressure, Initial Flow Velocity, and the Time Dependence of Crater Depth in Fluid Impacts. *J. Appl. Phys.* 38:3935–3940.
17. Fedorchenko AI, Wang A-B (2004). On Some Common Features of Drop Impact on Liquid Surfaces. *Phys. Fluids* 16:1349–1365.
18. Ghadiri H (2004). Crater Formation in Soils by Raindrop Impact. *Earth Surf. Process. Landforms* 29:77–89.
19. Guy RD, Fogelson AL (2005). Stability of Approximate Projection Methods on Cell-Centred Grids. *J. Comput. Phys.* 203:517–538.
20. Josserand C, Zaleski S (2003). Droplet Splashing on a Thin Liquid Film. *Phys. Fluids* 15:1650–1657.
21. Kothe DB (1999). "Perspectives on Eulerian Finite Volume Methods for Incompressible Interfacial Flows." In *Free Surface Flows*. Ed. Kuhlmann HC, Rath H-J. New York: Springer, 267–331.
22. Lakehal D, Meier M, Fulgosi M (2002). Interface Tracking Towards the Direct Simulation of Heat and Mass Transfer in Multiphase Flows. *Int. J. Heat Fluid Flow* 23:242–257.
23. Liow J-L (2001). Splash Formation by Spherical Drops. *J. Fluid Mech.* 427:73–105.
24. Morton, DE, Rudman M, Liow J-L (2000). Bubble Entrainment Resulting from Splashing Drops. *Phys. Fluids* 12:747–763.
25. Nourgaliev RR, Dinh TN, Theofanous TG (2003). *Godunov-Bi-Grid-Interpolation-Based Projection Method*, CRSS Research Report 07/04-1.
26. Puckett EG, Almgren A, Bell JB, Marcus DL, Rider WJ (1997). A High-Order Projection Method for Tracking Fluid Interfaces in Variable Density Incompressible Flows. *J. Comput. Phys.* 130:269–282.
27. Rider WJ (1994). *Approximate Projection Methods for Incompressible Flow: Implementation, Variants and Robustness*, Technical Report, LA-UR-2000, LANL.
28. Rider WJ, Kothe DB, Mosso SJ, Cerutti JH, Hochstein JI (1995). Accurate Solution Algorithms for Incompressible Multiphase Flows. AIAA Paper No. 95-0699, *33rd Aerospace Sciences Meeting*, Reno, NV.
29. Rider WJ, Kothe DB, Puckett EG, Aleinov ID (1998). "Accurate and Robust methods for Variable Density Incompressible Flows with Discontinuities." In *Workshop on Barriers*

and Challenges in Computational Fluid Dynamics. Ed. Venkatakrishnan V, Salas MD, Chakravarthy SR. Boston MA, pp. 213–230.
30. Rioboo R, Marengo M, Tropea C (2002). Time Evolution of Liquid Drop Impact Onto Solid, Dry Surfaces. *Exp. Fluids* 33:112–124.
31. Ritchie BW, Thomas PA (2001). Multiphase Smoothed-Particle Hydrodynamics. *Mon. Not. R. Astron. Soc.* 323:743–756.
32. Scardovelli R, Zaleski S (1999). Direct Numerical Simulation of Free-Surface and Interfacial Flow. *Annu. Rev. Fluid Mech.* 31:567–603.
33. Sussman M, Almgren AS, Bell JB, Colella P, Howell LH, Welcome ML (1999). An Adaptive Level Set Approach for Incompressible Two-Phase Flow. *J. Comput. Phys.* 148:81–124.
34. Sussman M, Smereka P, Osher S (1994). A Level-Set Approach for Computing Solutions to Incompressible Two-Phase Flow. *J. Comput. Phys.* 114:146–159.
35. Szymczak WG, Rogers JCW, Solomon JM, Berger AEB (1993). A Numerical Algorithm for Hydrodynamic Free Boundary Problems. *J. Comput. Phys.* 106:319–336.
36. Vincent S, Caltagirone J-P (2000). A One-Cell Local Multigrid Method for Solving Unsteady Incompressible Multiphase Flow. *J. Comput. Phys.* 163:172–215.
37. Williams MW, Kothe DB, Puckett EG (1998). Convergence and Accuracy of Kernel-Based Continuum Surface Tension Models. *Proc 13th US Natl Congress of Appl Mech*, Gainesville, June 16-21, FL.
38. Yarin AL, Weiss DA (1995). Impact of Drops on Solid Surfaces: Self-Similar Capillary Waves, and Splashing as a New Type of Kinematic Discontinuity. *J. Fluid Mech.* 283:141–173.

APPENDIX

The expression for the Laplacian of the vector $\sigma\nabla\phi=(\psi^x, \psi^y)^T$ is, using the earlier constructed x and y components (11) and (12),

$$(L_\mu\sigma\mathbf{G}\phi)^x_{i,j} = 2\left[\mu_{i+1/2,j}(\psi^x_x)_{i+1/2,j} - \mu_{i-1/2,j}(\psi^x_x)_{i-1/2,j}\right]/\Delta x + \left[\mu_{i,j+1/2}\left((\psi^x_y)+(\psi^y_x)\right)_{i,j+1/2} - \mu_{i,j-1/2}\left((\psi^x_y)+(\psi^y_x)\right)_{i,j-1/2}\right]/\Delta y$$
$$= 2\left[\mu_{i+1/2,j}(\psi^x_{i+1,j}-\psi^x_{i,j}) - \mu_{i-1/2,j}(\psi^x_{i,j}-\psi^x_{i-1,j})\right]/\Delta x^2 + \left[\mu_{i,j+1/2}\left(\psi^x_{i,j+1}-\psi^x_{i,j}\right) - \mu_{i,j-1/2}\left(\psi^x_{i,j}-\psi^x_{i,j-1}\right)\right]/\Delta y^2 + \left[\mu_{i,j+1/2}(\psi^y_x)_{i,j+1/2} - \mu_{i,j-1/2}(\psi^y_x)_{i,j-1/2}\right]/\Delta y$$

$$(L_\mu\sigma\mathbf{G}\phi)^y_{i,j} = 2\left[\mu_{i,j+1/2}(\psi^y_y)_{i,j+1/2} - \mu_{i,j-1/2}(\psi^y_y)_{i,j-1/2}\right]/\Delta y + \left[\mu_{i+1/2,j}\left((\psi^x_y)+(\psi^y_x)\right)_{i+1/2,j} - \mu_{i-1/2,j}\left((\psi^x_y)+(\psi^y_x)\right)_{i-1/2,j}\right]/\Delta x$$
$$= 2\left[\mu_{i,j+1/2}(\psi^y_{i,j+1}-\psi^y_{i,j}) - \mu_{i,j-1/2}(\psi^y_{i,j}-\psi^y_{i,j-1})\right]/\Delta y^2 + \left[\mu_{i+1/2,j}\left(\psi^y_{i+1,j}-\psi^y_{i,j}\right) - \mu_{i-1/2,j}\left(\psi^y_{i,j}-\psi^y_{i-1,j}\right)\right]/\Delta x^2 + \left[\mu_{i+1/2,j}(\psi^x_y)_{i+1/2,j} - \mu_{i-1/2,j}(\psi^x_y)_{i-1/2,j}\right]/\Delta x$$

where the x and y subscripts refer to partial derivatives. Here we use $(\psi^{x,y}_x)_{i+1/2,j} = (\psi^{x,y}_{i+1,j} - \psi^{x,y}_{i,j})/\Delta x$ and $(\psi^{x,y}_y)_{i,j+1/2} = (\psi^{x,y}_{i,j+1} - \psi^{x,y}_{i,j})/\Delta y$. The expressions $(\psi^y_x)_{i,j+1/2}$ and $(\psi^x_y)_{i+1/2,j}$ are evaluated through averaging from known expressions:

$$(\psi^x_y)_{i+1/2,j} = \frac{1}{4}\Big((\psi^x_y)_{i,j+1/2} + (\psi^x_y)_{i,j-1/2} + (\psi^x_y)_{i+1,j+1/2} + (\psi^x_y)_{i+1,j-1/2}\Big)$$

$$(\psi^x_y)_{i,j+1/2} = \frac{1}{4}\Big((\psi^x_y)_{i+1/2,j} + (\psi^x_y)_{i-1/2,j} + (\psi^x_y)_{i+1/2,j+1} + (\psi^x_y)_{i-1/2,j+1}\Big)$$