submitted to *Geophys. J. Int.*

Multiphase flow modelling of volcanic ash particle settling in water using adaptive unstructured meshes

C. T. Jacobs,[1,2] G. S. Collins,[2] M. D. Piggott,[2,3] S. C. Kramer,[1,2] C. R. G. Wilson[4]

[1] *Institute of Shock Physics, Imperial College London, London SW7 2AZ, UK*

[2] *Department of Earth Science and Engineering, Imperial College London, London SW7 2AZ, UK*

[3] *Grantham Institute for Climate Change, Imperial College London, London SW7 2AZ, UK*

[4] *Lamont–Doherty Earth Observatory, Columbia University, New York 10964, USA*

SUMMARY

Small-scale experiments of volcanic ash particle settling in water have demonstrated that ash particles can either settle slowly and individually, or rapidly and collectively as a gravitationally unstable ash-laden plume. This has important implications for the emplacement of tephra deposits on the seabed. Numerical modelling has the potential to extend the results of laboratory experiments to larger scales and explore the conditions under which plumes may form and persist, but many existing models are computationally restricted by the fixed mesh approaches that they employ. In contrast, this article presents a new multiphase flow model that uses an adaptive unstructured mesh approach. As a simulation progresses, the mesh is optimised to focus numerical resolution in areas important to the dynamics and decrease it where it is not needed, thereby potentially reducing computational requirements. Model verification is performed using the method of manufactured solutions, which shows the correct solution convergence rates. Model validation and application considers two-dimensional simulations of plume formation in a water tank which replicate published laboratory experiments. The numerically predicted settling velocities for both individual particles and plumes, as well as instability behaviour, agree well with experimental data and observations. Plume settling is clearly hindered by the presence of a salinity gradient, and its influence must therefore be taken into account when considering particles in bodies of saline water. Furthermore, individual particles settle in the laminar flow regime while plume settling is shown (by plume Reynolds numbers greater than unity) to be in the turbulent flow regime, which has a significant impact on entrainment and settling rates. Mesh adaptivity maintains solution accuracy whilst providing a substantial reduction in computational requirements when compared to the same simulation performed using a fixed mesh, highlighting the benefits of an adaptive unstructured mesh approach.

Key words: Numerical solutions; Non-linear differential equations; Volcaniclastic deposits.

1 INTRODUCTION

The settling of particles under the influence of gravity in an aqueous solution has long been an important phenomenon. Starting from the early works of Richardson and Zaki (1954), Kuenen (1968) and Davis and Acrivos (1985), a vast amount of research has investigated the use of the sedimentation process in industrial applications. Grain-size analysis is a common example whereby particle diameters are inferred from the different settling velocities in the fluid. If the particles form distinct layers based on size classes at the bottom of the container, then they can also be separated from one another, or removed from the fluid completely as seen when clarifying waste water. Each of these applications requires a good understanding of the particle settling properties to give accurate and effective results. For example, particles were once thought to only settle individually under Stokes' law but further research has shown that the role of vertical density currents, which significantly affect particle settling velocities, must also be taken into account when performing grain-size analysis (Carey 1997; Kuenen 1968). Without this increased understanding, particle diameters may be poorly estimated.

The need for a better understanding of particle settling properties also extends to the natural world, where an important occurrence of sedimentation is the settling of pyroclastic fragmental material, generally referred to as tephra, in the world's oceans. Tephra particles from past volcanic eruptions settle to form layer after layer of deposits on the seabed. Correct interpretation of these layers can provide important constraints on the duration and frequency of volcanism (Carey and Schneider 2011) but, like the other aforementioned applications, requires a full understanding of the complex multiphase settling and deposition process.

Analogue experiments of tephra settling through a tank of water have demonstrated that small ash particles (i.e. tephra particles smaller than 2 mm in diameter (Rose and Durant 2009)) can either settle individually, or collectively as a gravitationally unstable ash-laden plume (Carey 1997). This behaviour is similar to that of virus particles in a buffer solution and fungal spores in air (Bradley 1965, 1969). These plumes are generated when the concentration of particles exceeds a certain threshold such that the bulk density of the tephra-water mixture is sufficiently large relative to the underlying particle-free water for a gravitational Rayleigh-Taylor instability to develop. Furthermore, these plumes are observed to descend as a vertical density current with a velocity much greater than that of individual particles, which has important implications for the emplacement of tephra deposits on the seabed (Carey 1997; Manville and Wilson 2004); the effects of ocean currents on the distribution of settling tephra particles will be lessened because of the shorter time-scale over which they can act, and information about atmospheric conditions at the time of an eruption is therefore more likely to be preserved by the tephra layers that form at the bottom of the ocean. These implications must also be taken into account when performing settling velocity-based grain-size analysis for particles with diameters smaller than 50 μm (Kuenen 1968).

Numerical modelling provides a method to extend the results of laboratory experiments to large scales and explore the conditions under which vertical density currents may form and persist. One crucial aspect of any numerical model is the discretisation of the domain into a finite number of cells, forming a mesh, where the properties of the flow such as velocity are approximated upon the solution nodes. This is directly related to both accuracy and computational footprint; more cells (or nodes) give better accuracy but at the cost of increased simulation runtime.

Many existing multiphase flow models solve the equations governing the flow dynamics either on a fixed structured grid or a fixed unstructured mesh. The former is not well-suited to handling complex geometries such as the bathymetry of the ocean (Pain *et al.* 2005), and can be inefficient if one only wants to accurately capture the dynamics in particular areas of the domain; since the numerical resolution is inherently uniform, a large number of superfluous nodes will exist. Use of fixed structured grids has restricted the resolution of even the most advanced structured grid-based multiphase models (Neri and Macedonio 1996; Neri *et al.* 2003; Textor *et al.* 2005; Esposti Ongaro *et al.* 2008). In contrast, unstructured meshes have the advantage that nodes can be arbitrarily connected to one another, thus providing the freedom for the resolution to increase or decrease only where desired (Piggott *et al.* 2006). However, a fixed unstructured mesh which concentrates resolution in a region of interest in the flow, such as around a vortex, will no longer be optimal if this vortex changes its position or size as the simulation progresses. The reduced accuracy resulting from this can dramatically affect the numerical solution.

To mitigate the issues associated with fixed meshes, this work developed a multiphase flow model within Fluidity, an open-source, combined finite element / control volume CFD code, which features an adaptive unstructured mesh based approach (Pain *et al.* 2001; Piggott *et al.* 2008, 2009; Davies *et al.* 2011; Imperial College London 2011). Adaptive unstructured meshes have the potential to supply finer numerical resolution only in areas important to the dynamics being studied and coarser resolution in those areas that are not. As the flow progresses the mesh is optimised through a series of local topological operations (such as node addition and edge flipping) to accurately, but efficiently, represent the flow domain throughout time (Piggott *et al.* 2009). This approach has already brought significant benefits to numerical models; for example, Fluidity's single-phase flow model has shown that an adaptive unstructured mesh can yield results equally accurate as those produced with a fixed mesh, with more than an order of magnitude fewer nodes (Hiester *et al.* 2011). Other models that use an adaptive mesh approach have also reported similar benefits, including faster runtimes and reduced computational costs over simulations performed with a uniformly fine mesh (Li and Kong 2009; Ito *et al.* 2011). Clearly the use of adaptive unstructured meshes is potentially very fruitful for modelling multiphase flows, particularly those involving geophysical processes occurring on a wide range of scales in complex domains.

The work herein describes the new multiphase capabilities of Fluidity, presents model verification and validation by simulating experiments of particle settling through a water tank, and highlights the benefits of mesh adaptivity by showing that it provides reduced computational cost compared with a mesh of uniform resolution without compromising solution accuracy. The remainder of this article is set out as follows. Section 2 presents the model equations and the fluid-particle drag term used. The equations are discretised using the finite element method in Section 3, followed by a description of the numerical method used to compute the solution in Section 4. The method of manufactured solutions, described in Section 5, verified the correctness of the model implementation in Fluidity. The experiment of Manville and Wilson (2004) involving particle settling in a tank of water with a salinity gradient was then simulated, and the numerically predicted plume depth and salinity profile were obtained; the model is validated in Section 6 by comparing these quantities against experimental data. Fluidity was also used to simulate similar settling experiments by Carey (1997), which involved several different size ranges of particles. The observed particle settling velocities are compared against those predicted numerically in Section 7. Finally, the key findings of this work are summarised in Section 8.

2 MODEL EQUATIONS

This work developed a model for dispersed multiphase flows composed of a single fluid phase (a connected liquid or gas substance) in which one or more particle phases (comprising solid particles, liquid droplets and/or gas bubbles) are immersed (Crowe *et al.* 1998). Each phase was identified by an index i, where $i = 1, 2, \ldots, N_{\text{phases}}$. By assuming that both the fluid phase and particle phase(s) could be treated as interpenetrating continua (the so-called Eulerian-Eulerian approach (Crowe 2005)), the physical laws of conservation of mass and momentum were used to derive the governing equations. These equations were considered on a domain $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega$ for an interval of time $t \in [0, T]$.

Each phase i was assigned a separate velocity field $\mathbf{u}_i$, and hence a separate momentum equation, to enable mixing and interpenetration. Extra terms were then included to account for interphase interactions. Furthermore, since only low-velocity gravity-driven flows were being considered, the model assumed incompressible flow and a common pressure field p so that only one continuity equation was used. Thus, the continuity equation and momentum equation for phase i (based on the derivation by Ishii (1975) in non-conservative form) were:

$$\sum_{i=1}^{N_{\text{phases}}} \nabla \cdot (\alpha_i \mathbf{u}_i) = 0, \tag{1}$$

$$\alpha_i \rho_i \frac{\partial \mathbf{u}_i}{\partial t} + \alpha_i \rho_i \mathbf{u}_i \cdot \nabla \mathbf{u}_i = -\alpha_i \nabla p + \alpha_i \rho_i \mathbf{g} + \nabla \cdot (\alpha_i \mu_i \nabla \mathbf{u}_i) + \mathbf{f}_i, \tag{2}$$

where ρ_i, μ_i and α_i are the density, isotropic viscosity and volume fraction of phase i respectively, $\mathbf{g}$ is the gravitational vector, and $\mathbf{f}_i$ represents the forces imposed on phase i by the other $N_{\text{phases}} - 1$ phases.

In this article, $\mathbf{f}_i$ represents only the fluid-particle drag force, $\mathbf{d}_i$, defined as

$$\mathbf{d}_i = \frac{3}{4} \mathrm{c}_i \frac{\alpha_f \alpha_i \rho_f |\mathbf{u}_i - \mathbf{u}_f|}{d_i} \left(\mathbf{u}_i - \mathbf{u}_f\right), \tag{3}$$

where the subscript f denotes properties of the fluid phase. A particle diameter d_i was required for each particle phase, but not the fluid phase, since $\mathbf{d}_f$ was always zero. The (Stokes) drag coefficient c_i and the particle Reynolds number Re_i were given by

$$\mathrm{c}_i = \frac{24}{\mathrm{Re}_i}, \tag{4}$$

and

$$\mathrm{Re}_i = \frac{\alpha_f \rho_f d_i |\mathbf{u}_i - \mathbf{u}_f|}{\mu_f}, \tag{5}$$

respectively (Crowe *et al.* 1998; Neri *et al.* 2003). Drag caused by particle-particle interaction was ignored because of the dilute nature of the multiphase flow.

The total value of $\mathbf{f}_i$ depended on whether phase i was the fluid phase or a particle phase; for the fluid phase $\mathbf{f}_f = \sum_{i=1}^{N_{\text{phases}}} \mathbf{d}_i$, whereas for a single particle phase p, $\mathbf{f}_p = -\mathbf{d}_p$ such that the system is closed (i.e. $\sum_{i=1}^{N_{\text{phases}}} \mathbf{f}_i = 0$). Note that only one particle phase was used for the simulations presented in this article (although Fluidity has been designed to handle an arbitrary number of particle phases). Therefore, from this point on properties of the particle phase will be denoted by a subscript p and the properties of the fluid phase with a subscript f.

A non-dimensional scaling analysis of the momentum equation (see Appendix A) showed that as the Reynolds number tends to zero, the stress tensor can become dominant relative to the other terms. For the fluid phase's momentum equation, where the viscosity was a known constant of $O(10^{-3})$ and the volume fraction was close to unity in this work, this term could not be neglected. In the case of the particle phase's momentum equation, the stress tensor depended on the particle phase viscosity which is commonly defined as a (dimensional) constant of $O(1)$ multiplied by the particle phase's volume fraction α_p (Miller and Gidaspow 1992; Neri *et al.* 2003). Since α_p was typically $O(10^{-3})$ in this work (at least in the main area of interest in the domain), the particle phase viscosity was of the same order of magnitude as the fluid phase viscosity. However, this particle phase viscosity gets multiplied by α_p in the particle phase's stress tensor, leading to a quantity at least three orders of magnitude smaller than the fluid phase's stress tensor, assuming the magnitudes of both velocity fields are approximately equal. For this reason the stress tensor in the particle phase's momentum equation was neglected in this work.

The Stokes drag coefficient was deemed appropriate for the simulations in this work because the maximum values of the particle Reynolds number, Re_p, determined *a posteriori*, were $O(10^{-1})$ and therefore implied that the flows under consideration were well within the Stokes flow regime.

It is worth noting that while the model has so far only been applied to particle settling experiments, it could potentially be applied to many other incompressible, dilute fluid-particle systems such as the flow of blood cells in a human body or sediment transport in rivers. However, the validity of the Stokes drag coefficient in these applications would need to be considered carefully.

3 DISCRETISATION

A discrete version of the continuous model equations was formed using the Galerkin finite element method. A full derivation is given in Appendix B, but put briefly, the method began by considering the weak form of the momentum equation:

$$\int_{\Omega} \mathbf{w} \cdot \left(\alpha\rho\frac{\partial \mathbf{u}}{\partial t}\right)\ \mathrm{dV} + \int_{\Omega} \mathbf{w} \cdot (\alpha\rho\mathbf{u} \cdot \nabla\mathbf{u})\ \mathrm{dV} = -\int_{\Omega} \mathbf{w} \cdot (\alpha\nabla p)\ \mathrm{dV} +$$
$$\int_{\Omega} \mathbf{w} \cdot (\alpha\rho\mathbf{g})\ \mathrm{dV} + \int_{\Omega} \mathbf{w} \cdot (\nabla \cdot (\alpha\mu\nabla\mathbf{u}))\ \mathrm{dV} - \int_{\Omega} \mathbf{w} \cdot \frac{3}{4}\mathrm{c}\frac{\alpha_f\alpha\rho_f|\mathbf{u}-\mathbf{u}_f|}{d}(\mathbf{u}-\mathbf{u}_f)\ \mathrm{dV}. \tag{6}$$

In this weak form a solution to the velocity field $\mathbf{u} \in H^1(\Omega)^3$ was sought such that it is valid for all test functions $\mathbf{w} \in H^1(\Omega)^3$ (where $H^1(\Omega)$ is the first Hilbertian Sobolev space) (Elman *et al.* 2005). Note that the subscripted i indexing a particular phase has been dropped for clarity.

The test function and solution to the velocity field were represented by a linear combination of piecewise linear basis functions (also known as P1$_{\mathrm{DG}}$ basis functions) that are discontinuous across the cells of the mesh, called elements, where two basis functions overlap. Therefore, within each element e,

$$\mathbf{w} = \sum_{j=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \phi_j\mathbf{w}_j, \tag{7}$$

$$\mathbf{u} = \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \phi_k\mathbf{u}_k, \tag{8}$$

where $N_{\mathbf{u}\_\mathrm{nodes},e}$ is the number of velocity solution nodes in element e, $\mathbf{w}_j$ is the value of the test function at node j, and $\mathbf{u}_k$ is the solution at node k. The basis functions ϕ_j and ϕ_k are unity at nodes j and k respectively, and zero at all other nodes. When seeking the coefficients $\mathbf{u}_k$, the pressure field p also needed to be solved for. This field was represented by continuous piecewise quadratic basis functions (also known as P2 basis functions) such that

$$p = \sum_{l=1}^{N_{p\_\mathrm{nodes}}} \psi_l p_l, \tag{9}$$

where p_l is the value of the pressure field at node l, and ψ_l is the basis function that is unity at node l and zero at all other nodes. Note the summation over all nodes of the domain due to the continuous nature of the basis functions.

The discretised momentum equation gave an $N_{\mathbf{u}\_\mathrm{nodes}} \times N_{\mathbf{u}\_\mathrm{nodes}}$ system of linear equations that could be assembled and solved for the vectors of unknown coefficients $\mathbf{u}$ and p:

$$\mathbf{M}\frac{\partial \mathbf{u}}{\partial t} + \mathbf{A}\mathbf{u} + \mathbf{K}\mathbf{u} + \mathbf{F}_{\mathrm{left}}\mathbf{u} = \mathbf{C}p + \mathbf{b} + \mathbf{f}_{\mathrm{right}}, \tag{10}$$

where the matrices $\mathbf{M}$, $\mathbf{A}$, $\mathbf{K}$ and $\mathbf{C}$ are the mass, advection, stress and gradient matrices respectively.

The matrix $\mathbf{F}_{\text{left}}$ contains the left-hand side part of the drag term. The vectors $\mathbf{b}$ and $\mathbf{f}_{\text{right}}$ represent the buoyancy force and the right-hand side part of the drag term. These terms are defined in Appendix B.

The solution to the discretised momentum equation needed to satisfy the discrete continuity equation, also formed using the finite element method (see Appendix B):

$$\sum_{i=1}^{N_{\text{phases}}} \left(\mathbf{C}_i^{\text{T}}\mathbf{u}_i - \mathbf{r}_i\right) = 0, \tag{11}$$

where $\mathbf{r}_i$ is a surface integral term through which Dirichlet boundary conditions can be applied.

The volume fraction fields α_p and α_f were discretised separately using a node-centred control volume approach (Wilson 2009) and advected with the velocity fields (once known). The face values of each control volume were limited using the Sweby flux limiter (Sweby 1984).

4 SOLUTION METHOD

After the momentum equation was discretised, a method was required to compute the numerical solution. Fluidity solves the single-phase incompressible Navier-Stokes equations using a pressure projection method (see the work of Chorin (1968) and Gresho and Sani (2000) for more details) which has been extended to solve the multiphase model equations employed in this work.

The solution method began by considering the momentum equation that had been discretised in space using the Galerkin finite element method, and in time using the backward Euler method. Each time-step (from time n to $n+1$) comprised a number of Picard iterations used to deal with the nonlinearity in the system, yielding a set of tentative results for $\mathbf{u}_i^{n+1}$ and p^{n+1}, denoted by $\mathbf{u}_i^{\text{tent}}$ and p^{tent}. Therefore, within each time-step, the method sought a new tentative solution at each Picard iteration to

$$\mathbf{M}_i\frac{\mathbf{u}_i^{\text{tent}} - \mathbf{u}_i^n}{\Delta t} + \mathbf{A}_i\mathbf{u}_i^{\text{tent}} + \mathbf{K}_i\mathbf{u}_i^{\text{tent}} + \mathbf{F}_{\text{left},i}\mathbf{u}_i^{\text{tent}} = \mathbf{C}_i p^{\text{tent}} + \mathbf{b}_i + \mathbf{f}_{\text{right},i}, \tag{12}$$

such that the discrete continuity equation

$$\sum_{i=1}^{N_{\text{phases}}} \left(\mathbf{C}_i^{\text{T}}\mathbf{u}_i^{\text{tent}} - \mathbf{r}_i\right) = 0, \tag{13}$$

was satisfied.

The main steps of the method are described briefly below, but a full description is given in Appendix C. To find a new tentative solution:

(i) Compute non-linear approximations to the phase volume fraction and velocity fields using the latest (best available) tentative solutions.

(ii) Make a ‘best guess’ for p^{tent} by solving a pressure Poisson equation or by using the most up-to-date pressure field available.

(iii) Solve (12) for each phase to obtain a set of intermediate velocities $\mathbf{u}_i^*$.

(iv) Since the set of intermediate velocities will not satisfy (13) because of the guess used for the pressure field, find the pressure correction term Δp that enforces continuity by projecting the intermediate velocities onto a divergence-free space and solve:

$$\sum_{i=1}^{N_{\mathrm{phases}}} \left(\mathbf{r}_i - \mathbf{C}_i^{\mathrm{T}} \mathbf{u}_i^*\right) = \Delta t \left(\sum_{i=1}^{N_{\mathrm{phases}}} \mathbf{C}_i^{\mathrm{T}} \mathbf{M}_i^{-1} \mathbf{C}_i \right) \Delta p. \tag{14}$$

(v) Correct the intermediate velocities to obtain $\mathbf{u}_i^{\mathrm{tent}}$ by substituting the recently found pressure correction term into:

$$\mathbf{u}_i^{\mathrm{tent}} = \mathbf{u}_i^* + \Delta t \mathbf{M}_i^{-1} \mathbf{C}_i \Delta p. \tag{15}$$

(vi) Advect all tracer fields (including the phase volume fraction fields) using the new tentative velocities $\mathbf{u}_i^{\mathrm{tent}}$.

Once a desired Picard iteration limit or convergence was reached, $\mathbf{u}_i^{n+1}$ and p^{n+1} took the values of the final tentative solution and the time-step was deemed complete. The above solution method was then repeated until a desired time limit or steady state was attained.

5 MODEL VERIFICATION

One rigorous indication of model correctness came from a convergence analysis, which checked that errors in the numerical solution decreased at the expected rate as the mesh resolution increased. The method of manufactured solutions was used to obtain the solution error for $\mathbf{u}_i$ and p by constructing an analytical solution with which the numerical solution could be compared (Roache 2002), and can be broken down into four steps:

(i) Choose an analytical solution for each $\mathbf{u}_i$, α_i and p such that $\sum_{i=1}^{N_{\mathrm{phases}}} \nabla \cdot (\alpha_i \mathbf{u}_i) = 0$.

(ii) For each phase $i = 1, 2, \ldots, N_{\mathrm{phases}}$:

(a) Substitute $\mathbf{u}_i$, α_i and p into phase i’s momentum equation. Since the analytical solution will not be the exact solution in general, a non-zero residual term will be present on the RHS (i.e. a source term).

(b) Form a new version of phase i’s momentum equation which includes this source term, so that the residual will now be zero. In other words, the analytical solution that was chosen at the beginning is now the exact solution for this new version of the momentum equation.

(iii) Solve the new set of momentum equations which include the source terms.

(iv) Obtain the error for a range of characteristic element lengths, and plot this error to determine the order of convergence.

A two-phase MMS test was created to verify the order of convergence when using the P1$_{\mathrm{DG}}$-P2 element pair. The analytical solutions $\mathbf{u}_p = [\sin(x)\cos(y),\ \sin(y)\sin(x) - \cos(x)\sin(y)]^{\mathrm{T}}$, $\mathbf{u}_f = [0.25\cos(x)\cos(y) - x\cos(y),\ \sin(y)]^{\mathrm{T}}$ and $p = \cos(x)\cos(y)$ were used. The phase volume fractions $\alpha_p = 0.2$ and $\alpha_f = 0.8$ were prescribed across the whole domain and remained constant throughout time. For each velocity field, Dirichlet boundary conditions that agree with the analytical solution were imposed along with the initial condition $\mathbf{u}_p = \mathbf{u}_f = [0, 0]^{\mathrm{T}}$. The dimensions of the domain were $0.0 \leq x \leq \pi$ and $0.0 \leq y \leq \pi$. The physical parameters $\rho_f = 1.0$, $\rho_p = 2.5$, $\mu_f = 0.3$, $\mu_p = 0.3$ and $d_p = 1.0$ were chosen arbitrarily.

Four fixed unstructured meshes composed of triangular elements were produced with Gmsh (Geuzaine and Remacle 2009) using characteristic element lengths of $l = 0.64, 0.32, 0.16$ and 0.08. Decreasing time-step sizes of 0.016, 0.008, 0.004 and 0.002 maintained a constant bound on the Courant number. All simulations were run until the steady state conditions $\max(|\mathbf{u}_f^{n+1} - \mathbf{u}_f^{n}|) \leq 1.0 \times 10^{-8}$, $\max(|\mathbf{u}_p^{n+1} - \mathbf{u}_p^{n}|) \leq 1.0 \times 10^{-8}$ and $\max(|p^{n+1} - p^{n}|) \leq 1.0 \times 10^{-8}$ were attained.

Plots of the error in the velocity fields in Fig. 1 show successful convergence at second order as expected. Since the P1$_{\mathrm{DG}}$-P2 element pair exhibits the same error scaling for the pressure field as an element pair using piecewise linear or even piecewise constant basis functions (Cotter *et al.* 2009) the second order convergence for p was also expected, providing confidence in the model implementation.

6 MODEL VALIDATION

Laboratory-scale particle settling experiments were replicated to test the model's performance. In experiment dsa#3 of Manville and Wilson (2004), silicon carbide particles were introduced into a water tank from above via a settling column, at an average mass flux of $6.31 \times 10^{-2}\ \mathrm{kgm^{-2}s^{-1}}$, for 25 seconds. The water had a linear salinity gradient with a sharp inflexion at a depth of approximately 0.25 m. Plumes that formed near the surface of the water tank were observed to settle as vertical density currents with velocities significantly greater than the predicted Stokes' law velocity of a single particle. Eventually the plumes impinged on the inflexion in the salinity gradient and spread out horizontally, which momentarily hindered the settling process. The particles then continued their rapid descent to

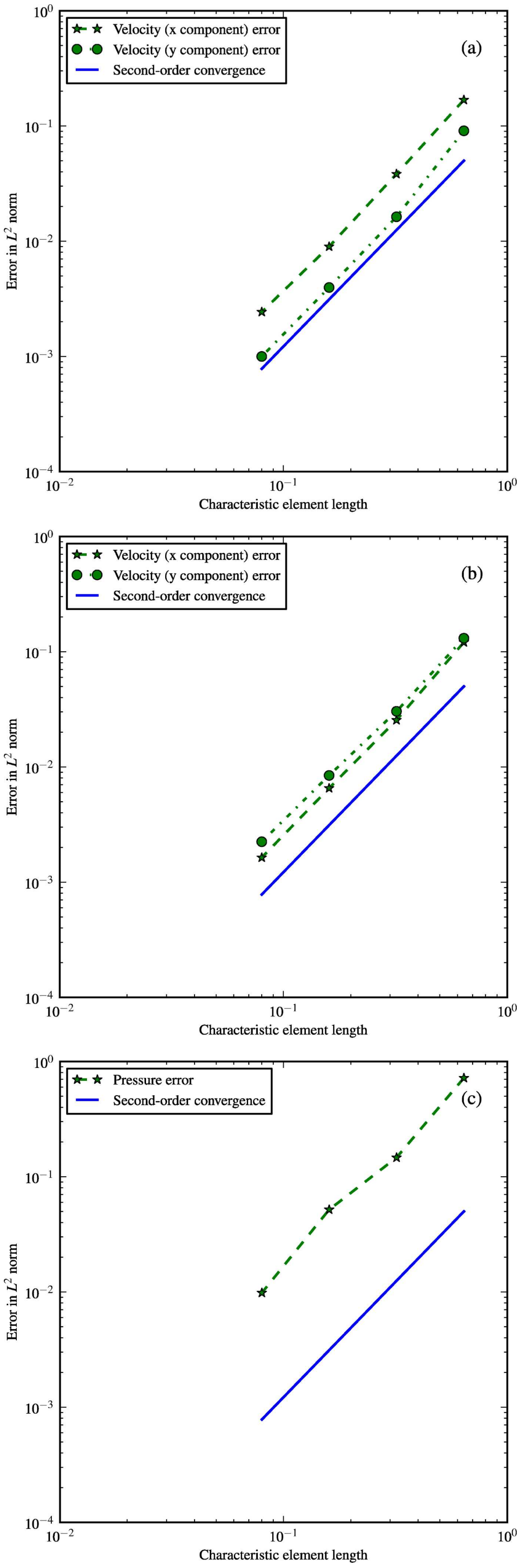

Figure 1. Convergence plots for (a) $\mathbf{u}_f$, (b) $\mathbf{u}_p$ and (c) p using the P1$_{\mathrm{DG}}$-P2 element pair. The velocity and pressure fields converged at second order as expected.

the bottom of the tank. Video recordings were used to measure the plume frontal position as a function of time. The salinity gradients before and after plume settling were also measured.

The salinity inflexion clearly had a significant impact on the plume dynamics. Although the effects were relatively short-lived in the laboratory experiment, the presence of a salinity gradient on a much larger scale could greatly influence the timescale of particle settling, hence the need for accurate and efficient numerical models. To validate the multiphase model presented here and evaluate the effectiveness of mesh adaptivity, a suite of simulations were performed in Fluidity which replicated the conditions under which experiment dsa#3 was performed. Both fixed and adaptive meshes were used.

6.1 The domain, initial conditions and boundary conditions

The domain was a rectangular box representing a cross-section of the water tank used by Manville and Wilson (2004), defined by $0.0 \leq x \leq 0.61$ m and $0.0 \leq y \leq 0.45$ m. A zero velocity field was imposed at $t = 0$ s for both phases, and an initial condition was also defined for the particle phase's volume fraction; a value of 1.0×10^{-7} was used throughout the domain, apart from the section $0.18 \leq x \leq 0.43$ m of the top boundary where nodal values were randomly perturbed such that $1.0 \times 10^{-7} \leq \alpha_p \leq 1.0 \times 10^{-5}$. This section represented the diameter of the circular container through which particles fell onto the body of water below. Such a perturbation encouraged plumes to form, and a minimum value of 1.0×10^{-7} was used instead of zero to avoid singularities in the system of linear equations. The following initial condition was used for the salinity field S to give a linear increase until a depth of 0.2441 m where an inflexion occurs, after which the salinity gradient becomes sharper (see Fig. 2):

$$S(x,y) = \begin{cases} 17.45 + 109.52(0.2059 - y) & \text{if } y \leq 0.2059 \\ 7 + 42.81(0.45 - y) & \text{otherwise} \end{cases} \tag{16}$$

No-normal flow boundary conditions, $\mathbf{u}_f \cdot \mathbf{n} = 0$ and $\mathbf{u}_p \cdot \mathbf{n} = 0$, were enforced on each boundary of the domain. A flux boundary condition for α_p was used to represent the introduction of particles; this boundary condition enforced

$$\frac{\partial}{\partial t}\int_{\Omega_v} \alpha_p \, \mathrm{dV} + \int_{\partial\Omega_v \setminus \partial\Omega} \widehat{\mathbf{n}_v \cdot \mathbf{u}_p \alpha_p} \, \mathrm{dS} = \int_{\partial\Omega_v \cap \partial\Omega} r \, \mathrm{dS}, \tag{17}$$

for all control volumes v on the boundary of Ω, and a given volumetric flux r. A volumetric flux of $r = 2.034 \times 10^{-5}$ ms^{-1}, found by dividing the mass flux of 6.31×10^{-2} kgm^{-2}s^{-1} used by Manville and Wilson (2004) by ρ_p, was applied along the top section $0.18 \leq x \leq 0.43$ m until t = 25 s; a zero

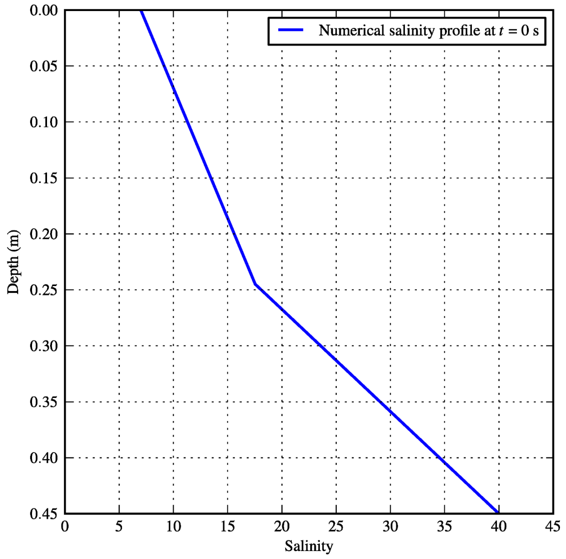

Figure 2. Variation of salinity with depth at t = 0 s. The inflexion occurs at a depth of 0.2441 m.

flux (i.e. $r = 0$ ms^{-1}) was applied everywhere else on $\partial\Omega$ to prevent particles entering or leaving. For $t > 25$ s, a zero flux was enforced everywhere on $\partial\Omega$ so that no more particles were introduced. Note also that the term $\mathbf{n}_v \cdot \widehat{\mathbf{u}_p \alpha_p}$ represents the flux of α_p through each control volume but excludes the contribution from the boundary condition which is provided separately by the term on the RHS.

The effects of representing the cylindrical particle inlet by a Cartesian geometry for the numerical simulations could not be quantified without running the model in a three-dimensional domain and comparing the results. However, possible differences may occur in the plume frontal speed, in the spreading at the salinity inflexion, and in the development of instabilities in the particle-water layer, despite the constant flux of particles through the inlet being the same in both two and three-dimensions. If particles were to experience drag effects from another dimension, this could cause differences in instability growth and plume dynamics. Furthermore, in two dimensions a plume only entrains particles from the left or right of the plume's tail, but in three dimensions there are more possible sources of particles that can become entrained, resulting in different entrainment rates which in turn affect the frontal speed and longevity.

6.2 Physical parameters

The physical parameters used were: $\rho_p = 3{,}100$ kgm^{-3}, $\mu_f = 0.001$ Pas, $d_p = 62$ μm (the mean diameter of particles used in the experiments) and $\mathbf{g} = [0, -9.8]^{\mathrm{T}}$ ms^{-2}. The particle phase was assumed to be inviscid (i.e. $\mu_p = 0$ Pas). The fluid density ρ_f obeyed a linear equation of state:

Table 1. Number of nodes in the fixed unstructured meshes.

l (m)	N_{vertices}	$N_{\mathbf{u}\_\mathrm{nodes}}$	N_{p_nodes}	Reference
0.02	820	4,602	924	F1
0.01	3,090	17,910	3,298	F2
0.005	12,799	75,528	13,219	F3
0.0025	50,943	303,120	51,787	F4

$$\rho_f = \rho_0 \left(1.0 + \beta \left(S - S_0\right)\right) \tag{18}$$

where ρ_0, S_0 and β are the fluid reference density, reference salinity, and saline contraction coefficient respectively. For this work, $\rho_0 =$ 1,000 kgm^{-3}, $S_0 = 0$, and $\beta = 7.2088 \times 10^{-4}$ (a generally accepted test value from McDougall (1987)).

6.3 Spatial discretisation and time-stepping

Further to the discretisation of the model equations and the volume fraction fields described in Section 3, the salinity field was discretised using a node-centred control volume approach (Wilson 2009). Once again, the Sweby flux limiter (Sweby 1984) was used.

The implicit backward Euler scheme marched the equations forward in time for 180 s. After an initial time-step of 0.001 s, Fluidity's adaptive time-stepping method permitted larger time-steps whilst enforcing a maximum Courant number of 0.5. Furthermore, within each time-step, two Picard iterations dealt with the non-linearity when solving the governing equations.

6.4 Meshes

6.4.1 Fixed meshes

Gmsh (Geuzaine and Remacle 2009) was used to generate unstructured meshes composed of triangular elements with a user-defined characteristic element length l. Four different values of l were used in the fixed mesh simulations, listed in Table 1 with the corresponding number of vertices N_{vertices}, velocity nodes $N_{\mathbf{u}\_\mathrm{nodes}}$ and pressure nodes N_{p_nodes}.

6.4.2 Mesh adaptivity

Simulations using mesh adaptivity were supplied an initial mesh with $l = 0.0025$ m, also generated using Gmsh. The mesh was then adapted every 20 time-steps using the libmba2d library (Vasilevski and Lipnikov 1999).

Fluidity seeks an accurate representation of the dynamics by optimising the mesh; this process is driven by three main components:

(i) *Topological operations*: Each element is considered individually and may be improved through a series of topological operations; edges can be split to introduce new elements and refine the mesh, or collapsed to remove elements and coarsen the mesh, for example (Piggott *et al.* 2009).

(ii) *Quality functional*: The decision to improve an element or not, and how, is guided by a quality functional Q which considers an element's size and shape. The adaptivity libraries optimise this functional by seeking an ideal element, defined as having unit edge length with respect to a given metric.

(iii) *Metric*: The quality functional is evaluated using a metric based on the curvature of a solution field of interest (in this case, α_p) and a user-defined weight ϵ (Pain *et al.* 2001). Areas of high solution field curvature will attract more resolution, and less so elsewhere. The weight ϵ stems from interpolation error theory; decreasing ϵ generally yields finer resolution because of the tighter error tolerance, while increasing it provides coarser resolution (Hiester *et al.* 2011). ϵ therefore controls the extent to which elements are refined.

After each adapt, the solution field required interpolation between the pre- and post-adapt meshes; this work used a linear interpolation scheme called consistent interpolation (Farrell 2009). The upper and lower bounds on the element size were set to $l_{\text{min}} = 0.00001$ m and $l_{\text{max}} = 0.1$ m throughout the domain, apart from the region defined by $0 \leq x \leq 0.61$ m and $0 \leq y \leq 0.02$ m where $l_{\text{min}} = 0.0025$ m; this was to prevent Fluidity from over-resolving the build-up of a particle layer that was not as important as the dynamics elsewhere in the tank.

Four different values of ϵ were used in the adaptive mesh simulations, listed in Table 2. These were absolute values of α_p, rather than relative values (i.e. the weight was defined in terms of a fixed volume fraction and not a percentage of the volume fraction field).

6.5 Results

The high particle influx caused the build-up of a particle layer along the surface of the tank which quickly became unstable because of the rapid increase in bulk density. Small plumes formed after approximately 3–4 s, most of which eventually became entrained in the flow to form one large plume

Table 2. Solution field weights for the adaptive unstructured mesh simulations.

ϵ	Reference
9.5×10^{-5}	A1
7.5×10^{-5}	A2
6.0×10^{-5}	A3
5.0×10^{-5}	A4

which descended with a velocity of around 0.01–0.02 ms^{-1}; this is greater than the predicted Stokes' law velocity of 0.0044 ms^{-1} for a single particle, as expected. After 15–20 s the plume head impinged on the salinity inflexion and spread out laterally because the plume reached a point where its bulk density was no longer greater than that of the underlying particle-free water. Such buoyancy effects caused a small amount of rebound also visible in the experiments. The particles then continued their descent to the bottom of the tank. Numerical model time frames showing plume evolution compare well with the experimental results (see Fig. 3; time frames of the experiment are taken from Manville and Wilson (2004)).

The numerical model accurately predicted the position of the plume front as a function of time, including the subtle change around 20 seconds when the plume impinged on the salinity inflexion (see Fig. 4). Throughout the simulation, the plume's passage led to entrainment of less saline water which in turn caused significant changes to the salinity profile. After the plume settled, however, the salinity gradient in the water returned to its initial state as observed in the experiments.

6.5.1 Plume speed comparison

To quantitatively assess the accuracy of the fixed and adaptive mesh approaches, and to further validate the numerical model, each simulation was repeated five times and the results were averaged. A linear least squares fit was applied to the averaged data points between $t = 4$ s (when plumes had formed) and $t = 15$ s (when the plume head first hit the salinity inflexion), and also between $t = 15$ s and $t = 31$ s. The gradients of these two linear fits gave an average plume frontal speed to be used as a measure of accuracy. A similar procedure was performed on the experimental data points to give a benchmark value with which to compare the numerical results against.

Figure 5 presents the average plume frontal speed against the average number of velocity nodes, for the two separate time intervals, using results from all 8 simulations. Vertical error bars represent the margin of error found from computing a 95% confidence interval for the slope of each linear fit.

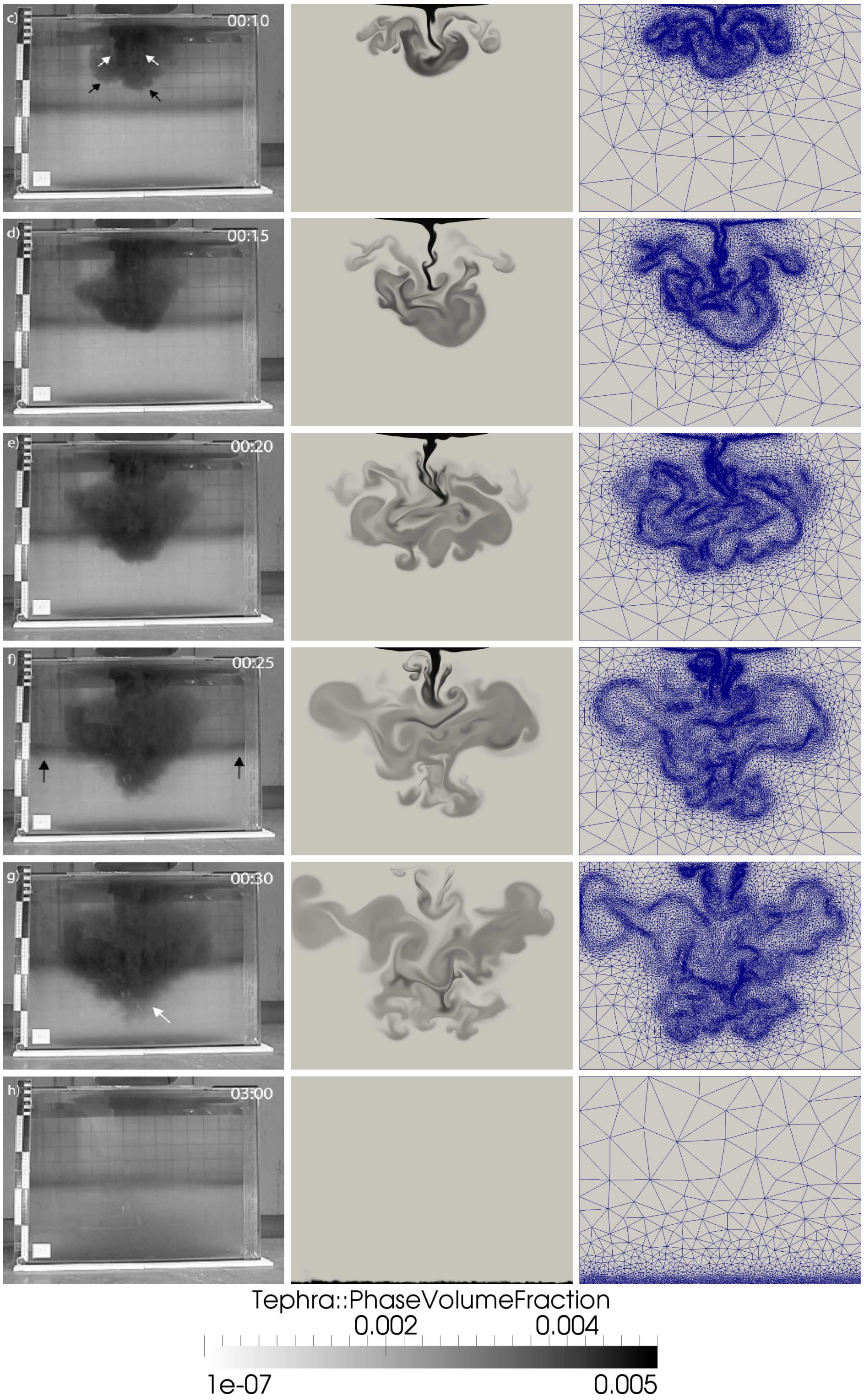

Figure 3. Simulation of particle settling through a salinity gradient at $t = 10, 15, 20, 25, 30$ and 180 s. Data taken from simulation A4. From left to right: video stills of the experiment from the article by Manville and Wilson (2004); visualisation of the phase volume fraction α_p; the adaptive unstructured mesh. Note that the colour bar is saturated at $t = 180$ s due to the build up of a dense particle layer on the surface, with a maximum α_p value of ∼0.4. All visualisations show the whole 0.61 m by 0.45 m domain.

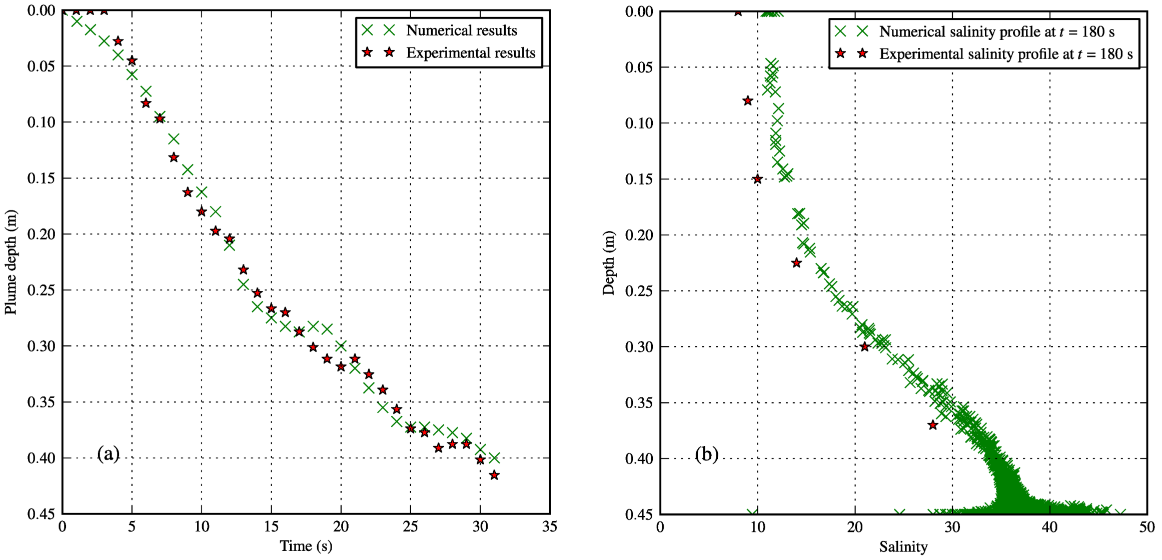

Figure 4. (a): Plume depth against time. (b): Variation of salinity with depth. Resolution is focused on the bottom of the tank at t = 180 s, so more data points are present there. Both plots use data from simulation A4.

Horizontal error bars show the maximum and minimum number of nodes used within the time interval under consideration.

In general, the plume frontal speeds from the adaptive mesh simulations closely matched the experimental data whilst requiring fewer nodes than the fixed mesh runs to attain the same solution accuracy. For example, the accuracy of simulation F4 is comparable to that of simulation A4 in the first

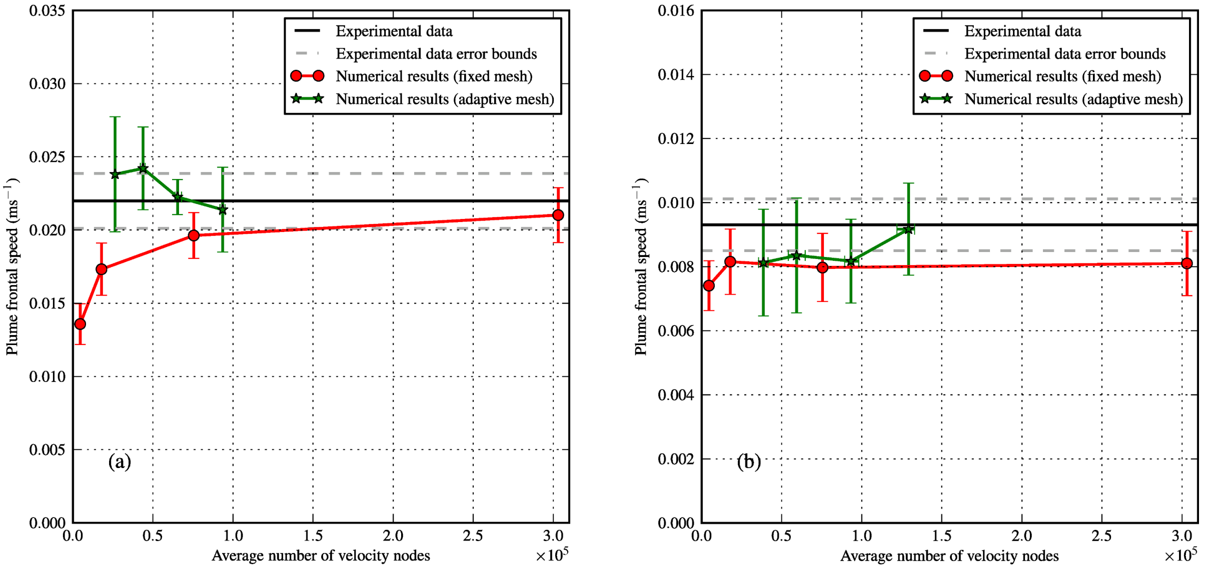

Figure 5. Average plume frontal speed against average number of velocity nodes, between t = 4 s and t = 15 s (a), and also between t = 15 s and t = 31 s (b).

Table 3. Number of velocity nodes required for the fixed and adaptive meshes from $t = 0$ s to $t = 180$ s.

Reference	$N^{\mathrm{min}}_{\mathbf{u}\_\mathrm{nodes}}$	$N^{\mathrm{max}}_{\mathbf{u}\_\mathrm{nodes}}$	$N^{\mathrm{average}}_{\mathbf{u}\_\mathrm{nodes}}$	Time steps	% time in adaptivity
F1	4,602	4,602	4,602	798	—
F2	17,910	17,910	17,910	2,023	—
F3	75,528	75,528	75,528	4,624	—
F4	303,120	303,120	303,120	11,246	—
A1	9,705	303,102	21,024	13,618	10% (680 adapts)
A2	9,894	303,102	32,972	16,270	13% (813 adapts)
A3	9,303	303,102	53,966	20,027	7% (1,001 adapts)
A4	10,410	303,102	76,506	22,009	11% (1,100 adapts)

time interval, and to that of simulation A3 in the second. In both cases, the adaptive meshes contain at least 4 times as fewer velocity nodes on average.

While a certain amount of variability in the fixed mesh results came from randomly perturbing the initial value of α_p along the top boundary, the most likely cause of increased variability in the adaptive mesh results was the further numerical perturbation and diffusion introduced by altering the mesh throughout each simulation which in turn encouraged bigger differences in plume evolution. However, this did not greatly alter the overall behaviour.

6.5.2 *The effectiveness of mesh adaptivity*

The computational savings from using fewer nodes should also be weighed up against the cost of adaptivity. As a percentage of the total runtime, Fluidity spent around 10% in the adaptivity routines (as shown in Table 3) which included the assembly of the metric and the interpolation as well as the mesh optimisation itself. The cost of one adapt was approximately the same as that of two time-steps for all adaptive mesh simulations. Despite this, since the mesh only adapted every 20 time-steps, the extra cost was insignificant when taking the benefits of using fewer nodes without significant loss of accuracy (if any) into account.

The other extra cost, although not directly related to mesh adaptivity, came from the presence of smaller elements which generally caused Fluidity's adaptive time-stepping method to enforce smaller time-steps to prevent the CFL condition from being breached, which explains why the adaptive mesh simulations required more time-steps than those using a fixed mesh.

To evaluate both of the costs described above, the elapsed wall time per time-step and per adapt was obtained from simulation A3; the results are given in Fig. 6.

Simulations F4 and A3 gave comparable accuracy for the second time interval and, on average,

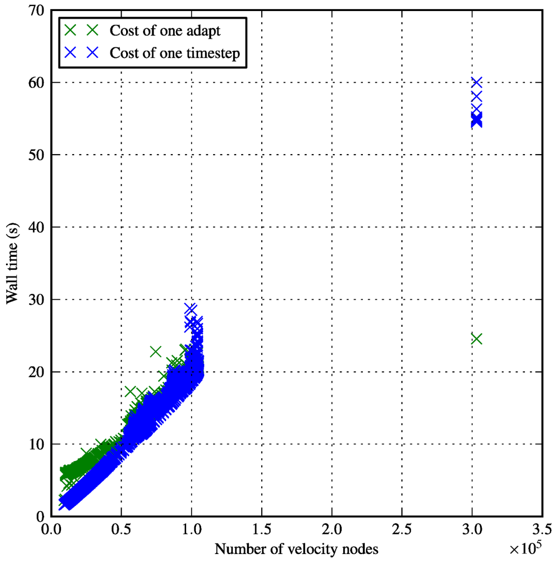

Figure 6. Wall time against number of velocity nodes.

used 303,120 and 53,966 velocity nodes respectively. Figure 6 shows that approximately 10 seconds of wall time were spent per time-step when using 53,966 nodes, but since A3 took around twice the number of time-steps, a measure of the overall cost relative to F4 was found by multiplying the 10 seconds of wall time by 2 and adding on the 7% extra time taken by the adaptivity routines. This gave 21 seconds of elapsed wall time in total — a saving of around 60% when compared with the 55 seconds of wall time required for one time-step in simulation F4. It is worth noting that this overall computational saving was achieved for simulations of plumes settling in a restricted, laboratory-scale domain. In this case, a large amount of resolution was placed in the majority of the domain at $t = 30$ s to resolve the laterally-spreading plume. If a similar plume settling problem was applied to the ocean scale, the area requiring high resolution would be much smaller relative to the overall size of the domain. The potential advantages that mesh adaptivity can offer therefore become more apparent with problems requiring resolution on a range of scales in larger, more complex domains, especially when three-dimensional domains are considered and when more computationally-demanding setups (e.g. multiple particle phases with different particle diameters) are desired.

7 MODEL APPLICATION

The laboratory experiments by Carey (1997) studied the dynamics of tephra particles in water by introducing them into a tank from above using a delivery system and a particle disperser, at an average mass flux of $4.72 \times 10^{-4}\ \mathrm{kgm}^{-2}\mathrm{s}^{-1}$. In these experiments, two stages of particle settling were observed. At early times when the concentration of particles was low, the particles abruptly decelerated upon their entry to the water and settled individually under Stokes' law at an average velocity of 0.002 ms^{-1} (for experiment 96-1), forming a layer of particle-rich water that increased in particle concentration with time. However, after about 30–60 seconds, gravitationally unstable particle-laden plumes grew from this layer as Rayleigh-Taylor instabilities, transporting particles to the bottom of the tank at speeds 10 times greater than that of single particles.

To measure the success of the numerical simulations presented herein, the model predictions of the onset time of pluming and the speed of the plumes were compared with estimates from the experiments.

7.1 Simulation setup

The domain was a rectangular column representing the water tank used by Carey (1997), defined by $0.0 \leq x \leq 0.3$ m, $0.0 \leq y \leq 0.7$ m and $0.0 \leq z \leq 0.3$ m in three dimensions. For the two-dimensional simulations presented in this work, a cross-section from the x-y plane was used.

A zero velocity field was imposed at time $t = 0$ s for both phases, and an initial condition was also defined for the particle phase's volume fraction; a value of 1.0×10^{-7} was used throughout the domain, apart from along the top boundary where nodal values were randomly perturbed such that $1.0 \times 10^{-7} \leq \alpha_p \leq 1.0 \times 10^{-5}$. This perturbation was done to encourage plumes to form, and a minimum value of 1.0×10^{-7} was used instead of zero to avoid singularities in the system of linear equations.

No-normal flow boundary conditions, $\mathbf{u}_f \cdot \mathbf{n} = 0$ and $\mathbf{u}_p \cdot \mathbf{n} = 0$, were enforced on each boundary of the domain. Along the top boundary a flux boundary condition, defined in Section 6, enforced a constant volumetric flux of 2.018×10^{-7} ms^{-1} which corresponded to a mass flux of 4.72×10^{-4} kgm^{-2}s^{-1}.

The physical parameters used were: $\rho_p =$ 2,340 kgm^{-3}, $\rho_f =$ 1,000 kgm^{-3}, $\mu_f = 0.001$ Pas, and $\mathbf{g} = [0, -9.8]^{\mathrm{T}}$ ms^{-2}. The particle diameter d_p varied between simulations, but ranged from 26 μm to 64 μm inclusive (within the range used by Carey (1997)). The particle phase was assumed to be inviscid (i.e. $\mu_p = 0$ Pas).

The setup of the time-stepping method was the same as that given in Section 6, except the simulation was performed only until $t = 120$ s.

The unstructured mesh, composed of triangular and tetrahedral elements in two and three dimensions respectively, was produced with Gmsh (Geuzaine and Remacle 2009). For fixed mesh simulations, the characteristic element length l was fixed at 0.0025 m unless stated otherwise. This setup produced a mesh containing 39,512 vertices, 234,678 velocity nodes and 40,308 pressure nodes. Simulations using mesh adaptivity were supplied an initial mesh with $l = 0.0025$ m, also generated using

Gmsh. The mesh was then adapted every 20 time-steps using the libmba2d library (Vasilevski and Lipnikov 1999). The mesh was not refined below a fixed minimum element length of $l_{\text{min}} = 0.00001$ m nor coarsened above a fixed maximum element length of $l_{\text{max}} = 0.1$ m. After each adapt, the solution fields were interpolated using consistent interpolation (Farrell 2009) as in Section 6. The weight ϵ was set to 1.0×10^{-2} in the section defined by $0.0 \leq x \leq 0.3$ and $0.0 \leq y \leq 0.05$ to prevent the build-up of a particle layer on the bottom which was not important in this study. Elsewhere, ϵ was set to 5.0×10^{-5}.

7.2 Results

Results from the tephra settling simulations are shown in Fig. 7. A near-surface layer of tephra particles formed during the first 15 s in the case where $d_p = 26\ \mu\text{m}$, and during the first 30 s for $d_p = 48\ \mu\text{m}$. In these early stages, the particles in the layer settled individually at the predicted Stokes' law velocity, as expected. From Stokes' law, the predicted settling velocities for particles with $d_p = 26\ \mu\text{m}$ and $d_p = 48\ \mu\text{m}$ are 0.00049 ms^{-1} and 0.00168 ms^{-1} respectively, and the numerical results in Fig. 8 agree well with this.

Just as Carey (1997) witnessed, as more tephra entered the water and the particle concentration increased the layer eventually became gravitationally unstable and plumes began to form, descending with velocities more than ten times greater than those of individual particles. Each plume was characterised by a vertical current of particles. As the velocity of the plume increased downwards, it displaced fluid that flowed around it with an equal and opposite velocity. This return flow caused drag effects that gave a variety of sharp and bulbous plume heads, and caused the longest plumes to entrain smaller plumes either side of them, thus furthering plume depth and longevity.

The smaller particle diameter of $d_p = 26\ \mu\text{m}$ caused the system to become unstable much sooner because the slower Stokes' law settling caused the near-surface layer concentration to build up quicker. The smaller layer thickness initially resulted in decreased plume length and diameter, when compared to the simulation with $d_p = 48\ \mu\text{m}$. Furthermore, a significant amount of particle entrainment was observed as the plumes grew from the layer and travelled towards the bottom of the water tank.

It is worth considering the differences between the development of Rayleigh-Taylor instabilities at the interface between two stratified immiscible fluids, and plume development in the tephra-water mixture. In the former case, if the denser fluid is lying on top of the lighter fluid then instabilities will form immediately and grow at an exponential rate (Rayleigh 1883; Duff *et al.* 1962). In the latter case, at early times the individual particles settle at Stokes' law velocity through the water as they are denser. However, finger-like instabilities may form at the interface between the upper, particle-laden layer and the lower, particle-free layer, if the concentration of particles in the upper layer builds up high

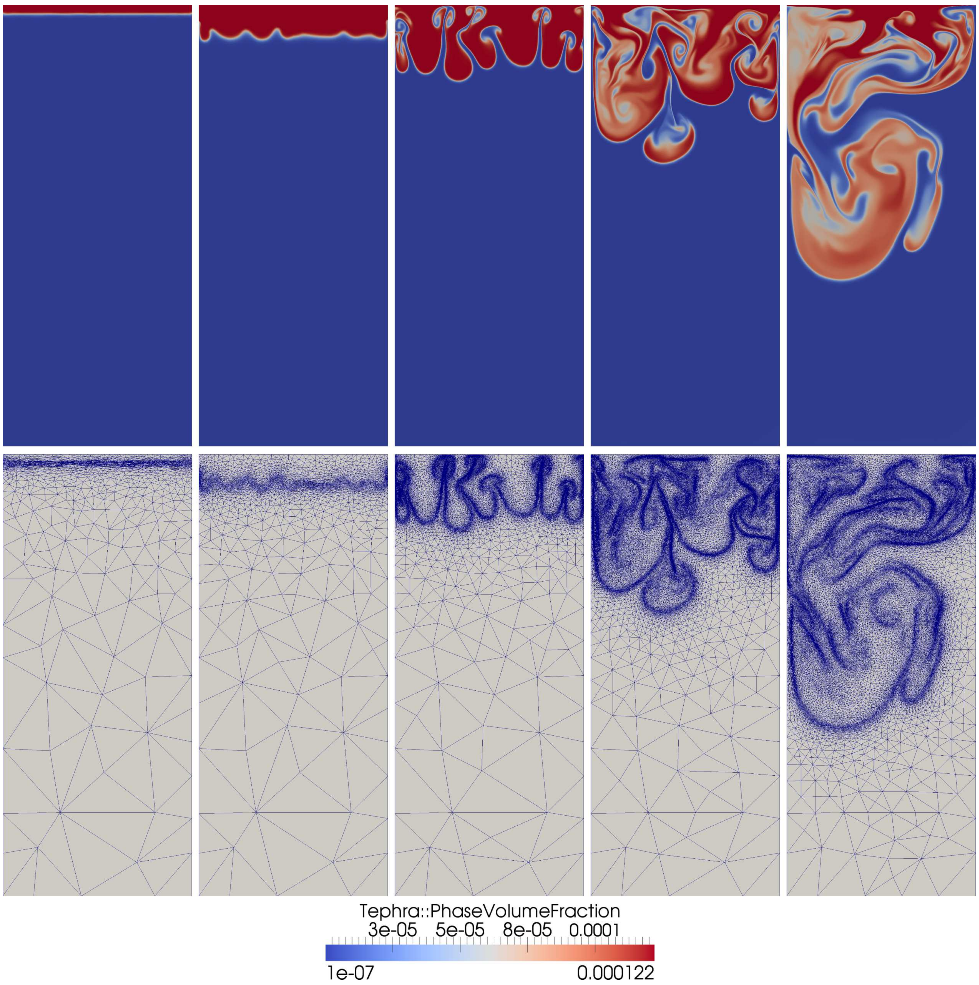

Figure 7. Visualisation of a two-dimensional adaptive mesh simulation with $d_p = 48\ \mu\mathrm{m}$, at $t = 10, 30, 50, 80$ and 120 s (from left to right). Warmer colours represent a higher volume fraction of the particle phase. All visualisations show the whole 0.3 m by 0.7 m domain.

enough for the particles to start having an effect on each other (through drag reduction and drifting). If the collective settling velocity is much faster than the Stokes' law velocity, the initial instability growth is likely to be similar to the case of two stratified immiscible fluids. On the other hand, as particle concentration within the plume changes with time, because of mixing and/or additional influx of particles, the buoyancy and hence the settling velocity of the plume also change with time, in a more complex manner than the idealised case of two immiscible fluids.

Experiments of the settling of a sand-water mixture through an underlying water layer (Lange *et al.* 1998), with similar physical parameters to the simulations presented in this article, showed an initial instability growth rate consistent with that predicted by linear stability analysis applied to two immiscible fluids (Rayleigh 1883). However, after only 200 ms of instability growth, the growth rate

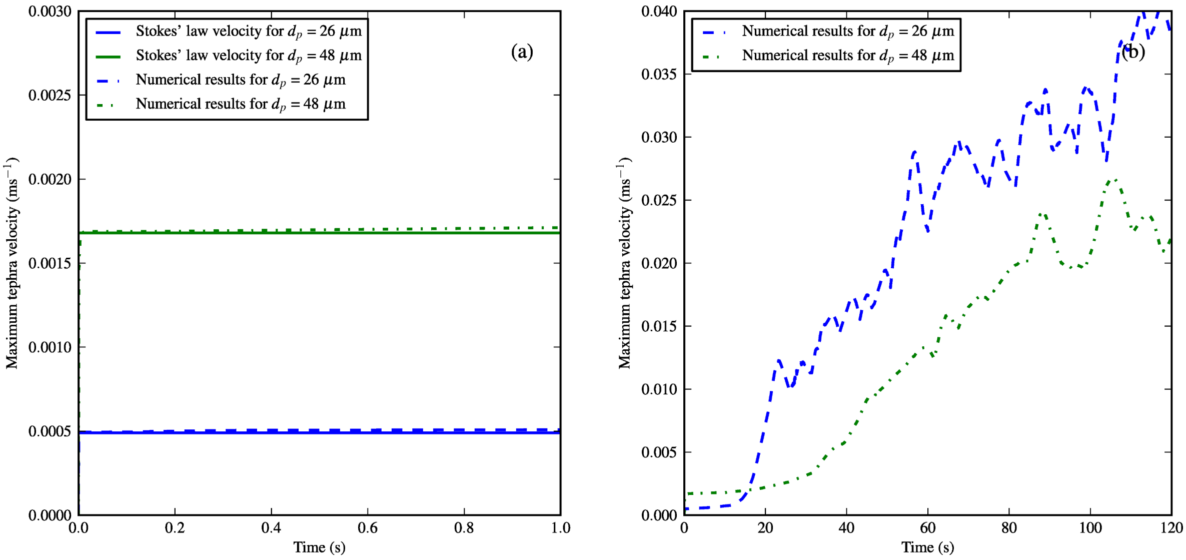

Figure 8. Maximum velocity of tephra particles against time, for $d_p = 26\ \mu\text{m}$ and $d_p = 48\ \mu\text{m}$. Tephra particles initially settle at the predicted Stokes' law velocity (a). As more tephra fluxes in, the layer becomes unstable and plumes begin to form, resulting in settling velocities over 10 times greater than that of an individual particle (b).

deviated from the ideal growth rate, reflecting the fact that the system rapidly became too non-linear for the stability analysis to hold.

An additional complexity of multiphase instability growth is that, unless the particle concentration remains high as a result of continual particle influx, entrainment of particle-free water into the plume reduces the buoyancy of the plume, slowing its descent. Moreover, the relative buoyancy of the plume may be further reduced by any increase in the fluid density with depth. As already seen in Section 6, plume settling was hindered when the surrounding water became denser than the plume due to an increasing salinity with depth, causing the plume to disperse. Running further simulations might allow the formulation of an empirical correlation to predict the onset of Rayleigh-Taylor instabilities as a function of layer concentration, particle diameters, influx rates and density contrasts.

Throughout each simulation the particle Reynolds number Re_p remained small ($O(10^{-1})$) when considering a single particle with diameter d_p. At the onset of plume formation, a new Reynolds number was defined at the plume scale by $\text{Re}_{\text{plume}} = \frac{\alpha_f \rho_f d_{\text{plume}} |\mathbf{u}_p - \mathbf{u}_f|}{\mu_f}$, where $d_{\text{plume}} >> d_p$ is the diameter of the plume. As the instabilities rapidly grew, Re_{plume} became much greater than unity because of the larger length scale. This implied that individual particle settling was in the laminar flow regime, while plumes were in the turbulent flow regime, which is consistent with the behaviour reported in the literature (Manville and Wilson 2004).

Plume formation occurred after 60 s for experiment 96-1 (which used a mean particle diameter of

Table 4. Number of nodes in the fixed unstructured meshes.

l (m)	N_{vertices}	$N_{\mathbf{u}\_\mathrm{nodes}}$	N_{p_nodes}	Reference
0.02	608	3,354	704	F1
0.01	2,423	13,944	2,619	F2
0.005	9,770	57,426	10,166	F3
0.0025	39,512	234,678	40,308	F4
0.00125	155,661	929,172	157,257	F5

$d_p = 48\ \mu\mathrm{m}$), and after 30 s for experiment 96-5 (which used a mean particle diameter of $d_p = 26\ \mu\mathrm{m}$) (Carey 1997); this qualitatively agrees with the behaviour seen in both simulations. For $d_p = 26\ \mu\mathrm{m}$, the wavelength of the growing instabilities was smaller compared to the $d_p = 48\ \mu\mathrm{m}$ case, and the resulting plumes were initially shorter and thinner but quickly merged via entrainment into larger plumes that settled with a velocity of around 0.04 ms^{-1} after 120 s. This was also expected because the spacing between growing instabilities and their diameters are related to the thickness of the particle-rich layer (Marsh 1988; Manville and Wilson 2004).

Eventually the particles were deposited at the bottom of the tank. Some particles were picked up by the return flow of fluid and recycled, feeding other plumes that are making their way to the bottom, while other particles remained stationary as expected.

7.3 Plume onset

To be confident that instabilities formed because of the physics and were not numerical artefacts, a convergence analysis was performed using both fixed and adaptive (two-dimensional) meshes up until the onset of plume formation. The integral of the kinetic energy density of the particle phase was measured as a function of time and was expected to converge to a particular value as the mesh was refined. Five fixed mesh simulations were run using decreasing characteristic element lengths, given in Table 4. Similarly, five decreasing values of ϵ were used for the adaptive mesh simulations, given in Table 5.

The simulation setup was as before apart from three modifications. Firstly, the initial particle phase volume fraction was perturbed along the top boundary using a sine function (instead of randomly) to avoid any stochastic effects between the data sets. Specifically, the volume fraction at node i along the top boundary was defined as $\alpha_p(x_i) = 10^{-7} + 10^{-5}\sin(\frac{\pi x_i}{0.3})$ such that only half a period was used to ensure the initial condition is independent of the mesh resolution. Otherwise, coarser meshes would poorly resolve (or miss out) higher frequencies which may in turn affect the dynamics. Secondly,

Table 5. Solution field weights for the adaptive unstructured mesh simulations.

ϵ	Reference
5.0×10^{-5}	A1
2.5×10^{-5}	A2
1.25×10^{-5}	A3
6.25×10^{-6}	A4
3.125×10^{-6}	A5

the frequency of an adapt was increased to once every 10 time-steps; since there are higher levels of numerical diffusion in cases where ϵ is relatively large, adapting more frequently will prevent the already-diffused tephra from moving too far out of the area of highest resolution between adapts, thus limiting further numerical diffusion. Finally, the particle diameter was set to 64 μm to increase the stability of the system, as it can be very difficult to identify convergence if instabilities form and become non-linear and turbulent too quickly. As the mesh resolution is increased, even a small difference in the perturbation of the particle phase volume fraction field can yield plumes of widely varying shape, size and position due to their chaotic nature. This means that there is no unique solution to converge to. Only the development of the particle layer up until the point at which instabilities start to form may be suitable to check for convergence because the dynamics are in the laminar flow regime and relatively linear. Quantitative values with which convergence can be judged were therefore only considered up until this point.

The integral of the kinetic energy density of the particle phase over a subsection of Ω, defined by $0.0 \leq x \leq 0.3$ and $0.05 \leq y \leq 0.7$, was plotted throughout time only up until the point where instabilities noticeably form at around 30 s; the dynamics became too turbulent and non-linear to show convergence after this time for the reasons explained in the previous sub-section. This upper subsection of Ω was used in order to make a fair comparison between fixed and adaptive meshes, because as l decreases in the fixed mesh simulations the particle layer that forms on the bottom of the domain will become better resolved, whereas the resolution will always stay coarse and remain the same in the adaptive mesh simulations despite a decreasing value of ϵ in the upper subsection.

Convergence was observed at $t = 30$ s, for both fixed and adaptive meshes, as shown in Figs 9 and 10 respectively. The error in Fig. 9 is the absolute difference of the integral of the kinetic energy density between simulation F5 and simulations F1 to F4 inclusive, after $t = 30$ s. Similarly, the error in Fig. 10 is the absolute difference of the integral of the kinetic energy density between simulation A5 and simulations A1 to A4 inclusive.

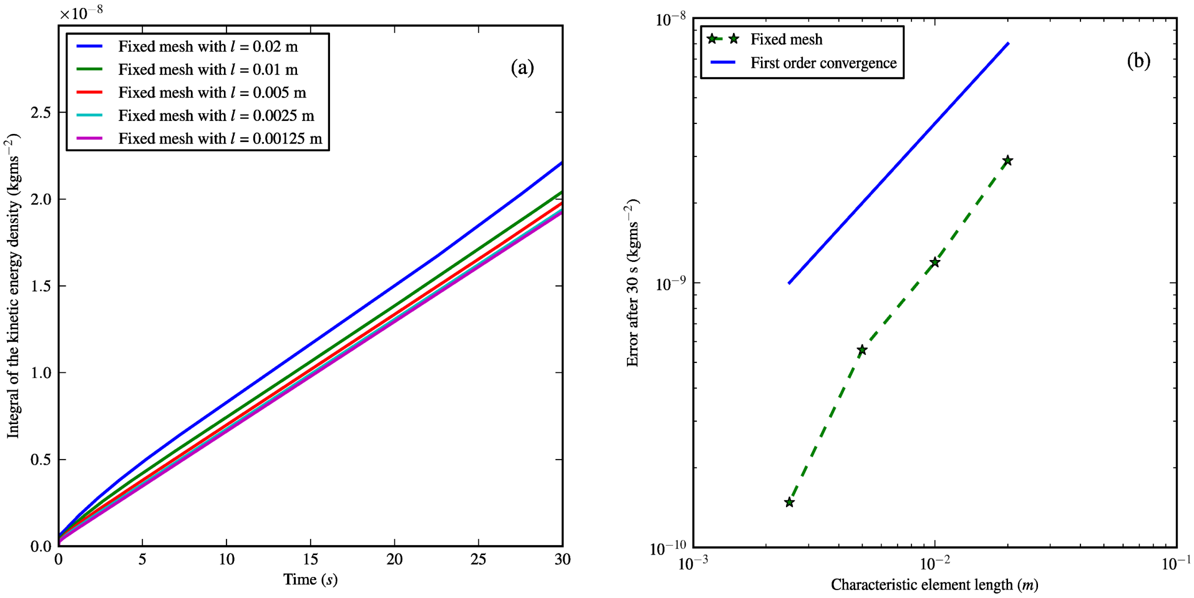

Figure 9. (a): Integral of the particle phase's kinetic energy density against time, on fixed unstructured meshes with different values of l. (b): Error in the integral of the kinetic energy density after $t = 30$ s.

A plot of the error against a common quantity — the number of velocity nodes — in Fig. 11 shows faster convergence with adaptive meshes whilst using approximately an order of magnitude fewer nodes.

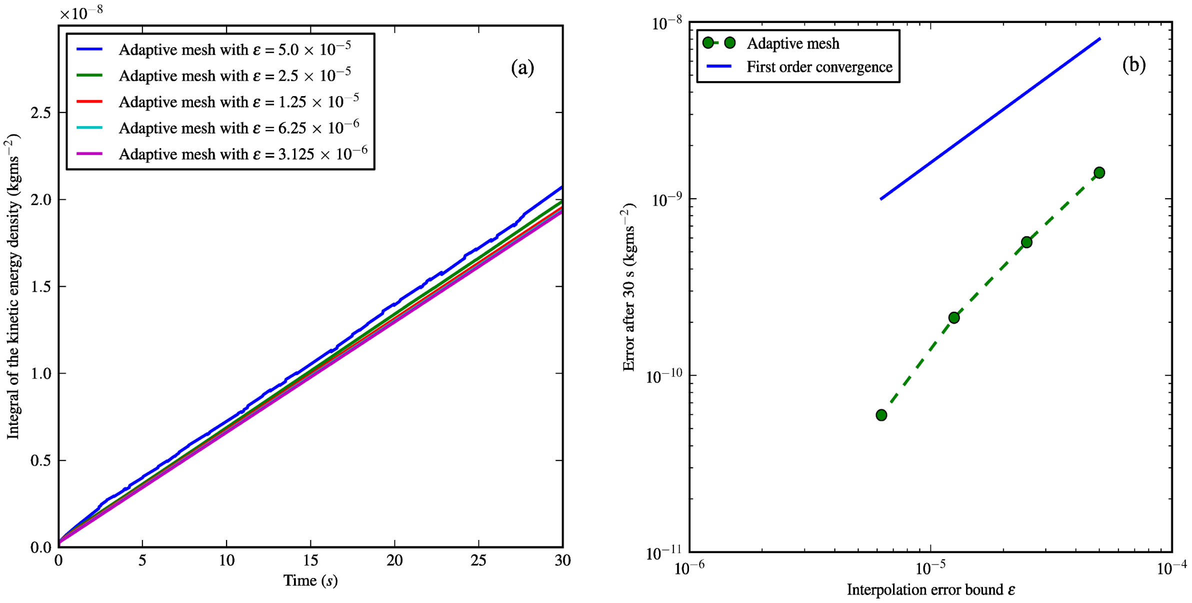

Figure 10. (a): Integral of the particle phase's kinetic energy density against time, on adaptive unstructured meshes with different values of ϵ. (b): Error in the integral of the kinetic energy density after $t = 30$ s.

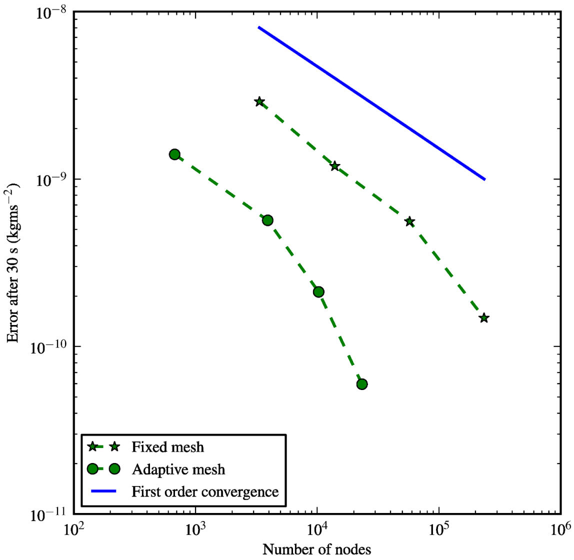

Figure 11. Convergence plot for fixed and adaptive meshes. Note that at $t = 30$ s, the integral of the kinetic energy density is 1.927×10^{-8} kgms^{-2} for F5 and 1.9347×10^{-8} kgms^{-2} for A5, which are close enough together for a reasonable comparison of errors to be made between the fixed and adaptive mesh simulations.

8 CONCLUSION

This work described the development of a dispersed multiphase flow model, implemented in the Fluidity CFD code, which used an adaptive unstructured mesh approach. The governing equations were introduced along with their discretisation via the finite element method. A pressure projection method was used to solve the resulting system of equations, and two Picard iterations dealt with the non-linearity. The model implementation's correctness was verified with the method of manufactured solutions, which showed second order convergence for the velocity and pressure fields as expected from the P1$_{\text{DG}}$-P2 element pair.

Experiment dsa#3 by Manville and Wilson (2004), which considered particle settling in a tank of water with a salinity gradient, was simulated to demonstrate that plume descent is hampered by the presence of a salinity (and therefore density) gradient. In both fixed and adaptive mesh simulations, the plumes that formed in the first 4–5 seconds later impinged on the salinity inflexion after approximately 15–20 seconds and spread out laterally, after which the particles continued their descent to the bottom. The salinity profile at $t = 180$ s showed that it remains mostly unchanged by the passage of the plume, agreeing well with experimental observations. The numerically predicted frontal speed of the plumes was also obtained from plume depth data during two time intervals: 4 to 15 seconds, and 15 to 31 seconds; the same was done using the experimental data. The fixed and adaptive mesh simulations converged towards the two values derived from the experiment, thereby providing a first step towards model validation, but the adaptive mesh approach allowed the use of at least 4 times as fewer velocity nodes when compared with a fixed unstructured mesh simulation. This provided an approximate computational saving of around 60% when compared to a fixed mesh simulation, without loss of solution accuracy. The benefits of mesh adaptivity are expected to be even more pronounced when simulating plume settling through a large-scale ocean column.

Adaptive unstructured mesh simulations of the experiments by Carey (1997), which considered different size ranges of tephra particles, were also performed for $48\ \mu\mathrm{m}$ and $26\ \mu\mathrm{m}$ particle diameters. Particles fluxed in and began settling individually at the correct velocity predicted by Stokes' law, but once the bulk density of the tephra-water mixture was large enough, plumes formed with the help of the small perturbations randomly seeded in the particle phase volume fraction field. The velocities increased to over 10 times those of individual particles settling at Stokes' law velocity. For $d_p = 26\ \mu\mathrm{m}$ and $d_p = 48\ \mu\mathrm{m}$ this happened after approximately 30 s and 60 s respectively, which closely matched experimental observations. A convergence analysis showed that the adaptive unstructured mesh used approximately one order of magnitude fewer nodes than the fixed uniform unstructured mesh would to reach the same level of accuracy, further illustrating the potential advantages that mesh adaptivity can bring to the numerical modelling community.

ACKNOWLEDGMENTS

C. T. Jacobs and S. C. Kramer were funded by the Institute of Shock Physics at Imperial College London and the Atomic Weapons Establishment; G. S. Collins was funded by the Natural Environment Research Council, Fellowship Grant NE/E013589/1. Support from the Imperial College High Performance Computing Service was gratefully received. Special thanks go to T. J. Goldin for helpful discussions regarding tephra settling, and to V. Manville for a copy of the experimental data presented in the article by Manville and Wilson (2004). The reviews received from V. Manville and an anonymous reviewer were greatly appreciated and significantly improved the quality of this article.

REFERENCES

Bassi, F., Rebay, S., 1997. A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier-Stokes Equations. Journal of Computational Physics 131 (2), 267–279.

Bradley, W. H., 1965. Vertical Density Currents. Science 150 (3702), 1423–1428.

Bradley, W. H., 1965. Vertical Density Currents-II. Limnology and Oceanography 14 (1), 1–3.

Carey, S., 1997. Influence of convective sedimentation on the formation of widespread tephra fall layers in the deep sea. Geology 25 (9), 839–842.

Carey, S. N., Schneider, J.-L., 2011. Volcaniclastic Processes and Deposits in the Deep-Sea. In: Hüneke, H., Mulder, T. (Eds.), Deep-Sea Sediments. Vol. 63 of Developments in Sedimentology. Elsevier, Ch. 7, pp. 457–515.

Chorin, A. J., 1968. Numerical Solution of the Navier-Stokes Equations. Mathematics of Computation 22 (104), 745–762.

Cotter, C. J., Ham, D. A., Pain, C. C., Reich, S., 2009. LBB stability of a mixed Galerkin finite element pair for fluid flow simulations. Journal of Computational Physics 228 (2).

Crowe, C. T., 2005. Multiphase Flow Handbook. CRC Press.

Crowe, C. T., Sommerfeld, M., Tsuji, Y., 1998. Multiphase Flows with Droplets and Particles. CRC Press.

Davies, D. R., Wilson, C. R., Kramer, S. C., 2011. Fluidity: A fully unstructured anisotropic adaptive mesh computational modeling framework for geodynamics. Geochemistry Geophysics Geosystems 12 (6).

Davis, R. H., Acrivos, A., 1985. Sedimentation of Noncolloidal Particles at Low Reynolds Numbers. Annual Review of Fluid Mechanics 17, 91–118.

Duff, R. E., Harlow, F. H., Hirt, C. W., 1962. Effects of diffusion on interface instability between gases. Physics of Fluids 5 (4), 417–425.

Elman, H. C., Silvester, D. J., Wathen, A. J., 2005. Finite Elements and Fast Iterative Solvers: with applications in incompressible fluid dynamics. Oxford University Press.

Esposti Ongaro, T., Neri, A., Menconi, G., de'Michieli Vitturi, M., Marianelli, P., Cavazzoni, C., Erbacci, G., Baxter, P. J., 2008. Transient 3D numerical simulations of column collapse and pyroclastic density current scenarios at Vesuvius. Journal of Volcanology and Geothermal Research 178 (3), 378–396.

Farrell, P. E., 2009. Galerkin projection of discrete fields via supermesh construction. Ph.D. thesis, Imperial College London.

Geuzaine, C., Remacle, J.-F., 2009. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. International Journal for Numerical Methods in Engineering 79 (11), 1309–1331.

Gresho, P. M., Sani, R. L., 2000. Incompressible Flow and the Finite Element Method. Vol. 2: Isothermal Laminar Flow. Wiley.

Hiester, H. R., Piggott, M. D., Allison, P. A., 2011. The impact of mesh adaptivity on the gravity current front speed in a two-dimensional lock-exchange. Ocean Modelling 38 (1-2), 1–21.

Imperial College London, 2011. Fluidity 4.1 Manual.
URL `http://amcg.ese.ic.ac.uk/Fluidity`

Ishii, M., 1975. Thermo-Fluid Dynamic Theory of Two-Phase Flow. Eyrolles.

Ito, K., Kunugi, T., Ohshima, H., 2011. A high-precision unstructured adaptive mesh technique for gas–liquid two-phase flows. International Journal for Numerical Methods in Fluids 67 (11), 1571–1589.

Kuenen, Ph. H., 1968. Settling convection and grain-size analysis. Journal of Sedimentary Research 38 (3), 817–831.

Lange, A., Schröter, M., Scherer, M. A., Engel A., Rehberg, I., 1998. Fingering instability in a water-sand mixture. The European Physical Journal B 4 (4), 475–484.

Li, Y., Kong, S.-C., 2009. Mesh refinement algorithms in an unstructured solver for multiphase flow simulation using discrete particles. Journal of Computational Physics 228 (17), 6349–6360.

Manville, V., Wilson, C. J. N., 2004. Vertical density currents: a review of their potential role in the deposition and interpretation of deep-sea ash layers. Journal of the Geological Society, London 161, 947–958.

Marsh, B. D., 1988. Crystal Capture, Sorting, and Retention in Convecting Magma. Geological Society of America Bulletin 100, 1720–1737.

McDougall, T. J., 1987. Neutral Surfaces. Journal of Physical Oceanography 17, 1950–1964.

Miller, A., Gidaspow, D., 1992. Dense, vertical gas-solid flow in a pipe. AIChE Journal 38 (11), 1801–1815.

Neri, A., Esposti Ongaro, T., Macedonio, G., Gidaspow, D., 2003. Multiparticle simulation of collapsing volcanic columns and pyroclastic flow. Journal of Geophysical Research 108 (B4), 2202.

Neri, A., Macedonio, G., 1996. Numerical simulation of collapsing volcanic columns with particles of two sizes. Journal of Geophysical Research 101 (B4), 8153–8174.

Pain, C. C., Piggott, M. D., Goddard, A. J. H., Fang, F., Gorman, G. J., Marshall, D. P., Eaton, M. D., Power, P. W., de Oliveira, C. R. E., 2005. Three-dimensional unstructured mesh ocean modelling. Ocean Modelling 10 (1-2), 5–33.

Pain, C. C., Umpleby, A. P, de Oliveira, C. R. E., Goddard, A. J. H., 2001. Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations. Computer Methods in Applied Mechanics and Engineering 190 (29-30), 3771–3796.

Piggott, M. D., Farrell, P. E., Wilson, C. R., Gorman, G. J., Pain, C. C., 2009. Anisotropic mesh adaptivity for multi-scale ocean modelling. Philosophical Transactions of the Royal Society A 367 (1907), 4591–4611.

Piggott, M. D., Gorman, G. J., Pain, C. C., Allison, P. A., Candy, A. S., Martin, B. T., Wells, M. R., 2008. A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes. International Journal for Numerical Methods in Fluids 56 (8), 1003–1015.

Piggott, M. D., Pain, C. C., Gorman, G. J., Power, P. W., Goddard, A. J. H., 2006. *h*, *r*, and *hr* adaptivity with applications in numerical ocean modelling. Ocean Modelling 10 (1-2), 95–113.

Rayleigh, L., 1883. Investigation of the character of the equilibrium of an incompressible heavy fluid of variable density. Proceedings of the London Mathematical Society 14, 170–177.

Richardson, J. F., Zaki, W. N., 1954. Sedimentation and fluidisation. Part 1.. Transactions of the Institution of Chemical Engineers 32, 35-53.

Roache, P. J., 2002. Code Verification by the Method of Manufactured Solutions. Journal of Fluids Engineering 124 (1), 4–10.

Rose, W. I., Durant, A. J., 2009. Fine ash content of explosive eruptions. Journal of Volcanology and Geothermal Research 186 (1-2), 32–39.

Sweby, P. K., 1984. High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws. SIAM Journal on Numerical Analysis 21 (5), 995–1011.

Textor, C., Graf, H.-F., Longo, A., Neri, A., Esposti Ongaro, T., Papale, P., Timmreck, C., Ernst, G. G. J., 2005. Numerical simulation of explosive volcanic eruptions from the conduit flow to global atmospheric scales. Annals of Geophysics 48 (4-5), 817–842.

Table A1. Scaling parameters

Scale	Notation	Dimension
Length	L	$[L]$
Time	T	$[T]$
Mass	M	$[M]$
Speed	U	$[LT^{-1}]$
Pressure	P	$[MUL^{-2}T^{-1}]$
Gravitational acceleration	G	$[LT^{-2}]$
Force (per unit volume)	F	$[ML^{-2}T^{-2}]$

Vasilevski, Y. V., Lipnikov, K. N., 1999. An adaptive algorithm for quasioptimal mesh generation. Computational Mathematics and Mathematical Physics 39 (9), 1468–1486.

Wilson, C., 2009. Modelling Multiple-Material Flows on Adaptive Unstructured Meshes. Ph.D. thesis, Imperial College London.

Zienkiewicz, O. C., Taylor, R. L., 2000. The Finite Element Method, 5th Edition. Vol. 1: The Basis. Butterworth-Heinemann.

APPENDIX A: NON-DIMENSIONAL SCALING ANALYSIS

To non-dimensionalise (2), the scaling parameters in Table A1 were first defined.

Non-dimensional variables were then defined in terms of these scaling parameters and the dimensional variables in (2):

$$\tilde{t} = \frac{1}{T}t, \tag{A.1}$$

$$\tilde{p} = \frac{1}{P}p, \tag{A.2}$$

$$\tilde{\mathbf{u}}_i = \frac{1}{U_i}\mathbf{u}_i, \tag{A.3}$$

$$\tilde{\mathbf{g}} = \frac{1}{G}\mathbf{g}, \tag{A.4}$$

$$\tilde{\mathbf{f}}_i = \frac{1}{F_i}\mathbf{f}_i, \tag{A.5}$$

$$\tilde{\nabla} = L\nabla. \tag{A.6}$$

Substituting the non-dimensional variables into (2) gave:

$$\alpha_i\rho_i\frac{U_i}{T}\frac{\partial\tilde{\mathbf{u}}_i}{\partial\tilde{t}} + \alpha_i\rho_i\frac{U_i^2}{L}\tilde{\mathbf{u}}_i\cdot\tilde{\nabla}\tilde{\mathbf{u}}_i = -\alpha_i\frac{P}{L}\tilde{\nabla}\tilde{p} + \alpha_i\rho_i G\tilde{\mathbf{g}} + \frac{U_i}{L^2}\tilde{\nabla}\cdot\left(\alpha_i\mu_i\tilde{\nabla}\tilde{\mathbf{u}}_i\right) + F\tilde{\mathbf{f}}_i. \tag{A.7}$$

This equation was non-dimensionalised by dividing each term by $\frac{\rho_i U_i^2}{L}$ which has dimension $[ML^{-2}T^{-2}]$, yielding:

$$\alpha_i \frac{\partial \tilde{\mathbf{u}}_i}{\partial \tilde{t}} + \alpha_i \tilde{\mathbf{u}}_i \cdot \tilde{\nabla} \tilde{\mathbf{u}}_i = -\alpha_i \tilde{\nabla} \tilde{p} + \alpha_i \tilde{\mathbf{g}} + \frac{1}{\mathrm{Re}_i} \tilde{\nabla} \cdot \left(\alpha_i \tilde{\nabla} \tilde{\mathbf{u}}_i \right) + \tilde{\mathbf{f}}_i, \qquad \text{(A.8)}$$

where Re_i is the Reynolds number of phase i defined as $\frac{\rho_i U_i L}{\mu_i}$.

APPENDIX B: DISCRETISATION

B1 Weak form of the momentum equation

A discrete version of the continuous model equations was formed using the Galerkin finite element method which began by considering the weak form of the momentum equation, derived by multiplying (2) through by a vector-valued test function $\mathbf{w} \in H^1(\Omega)^3$ (where $H^1(\Omega)$ is the first Hilbertian Sobolev space) and integrating over the domain Ω as follows (Elman *et al.* 2005):

$$\int_\Omega \mathbf{w} \cdot \left(\alpha\rho \frac{\partial \mathbf{u}}{\partial t} \right) \,\mathrm{dV} + \int_\Omega \mathbf{w} \cdot (\alpha\rho\mathbf{u} \cdot \nabla\mathbf{u}) \,\mathrm{dV} = -\int_\Omega \mathbf{w} \cdot (\alpha\nabla p) \,\mathrm{dV} +$$
$$\int_\Omega \mathbf{w} \cdot (\alpha\rho\mathbf{g}) \,\mathrm{dV} + \int_\Omega \mathbf{w} \cdot (\nabla \cdot (\alpha\mu\nabla\mathbf{u})) \,\mathrm{dV} - \int_\Omega \mathbf{w} \cdot \frac{3}{4}\mathrm{c}\frac{\alpha_f \alpha \rho_f |\mathbf{u} - \mathbf{u}_f|}{d} (\mathbf{u} - \mathbf{u}_f) \,\mathrm{dV}. \qquad \text{(B.1)}$$

Note that the subscripted i indexing a particular phase has been dropped in this appendix for clarity. The particle phase form of the drag term has also been assumed here for simplicity, but the derivation of the weak form for the fluid phase follows the same methodology.

Integrating the advection and stress terms by parts and applying the divergence theorem yielded

$$\int_\Omega \mathbf{w} \cdot \left(\alpha\rho \frac{\partial \mathbf{u}}{\partial t} \right) \,\mathrm{dV} - \int_\Omega \mathbf{u} \cdot (\alpha\rho\mathbf{u}\nabla \cdot \mathbf{w}) \,\mathrm{dV} - \int_\Omega \mathbf{u} \cdot (\mathbf{w}\nabla \cdot (\alpha\rho\mathbf{u})) \,\mathrm{dV}$$
$$+ \int_{\partial\Omega} (\mathbf{u}\,(\mathbf{w} \cdot (\alpha\rho\mathbf{u}))) \cdot \mathbf{n} \,\mathrm{dS} = -\int_\Omega \mathbf{w} \cdot (\alpha\nabla p) \,\mathrm{dV} + \int_\Omega \mathbf{w} \cdot (\alpha\rho\mathbf{g}) \,\mathrm{dV} \qquad \text{(B.2)}$$
$$- \int_\Omega (\nabla\mathbf{w}) \cdot (\alpha\mu\nabla\mathbf{u}) \,\mathrm{dV} + \int_{\partial\Omega} (\mathbf{w} \cdot (\alpha\mu\nabla\mathbf{u})) \cdot \mathbf{n} \,\mathrm{dS} - \int_\Omega \mathbf{w} \cdot \frac{3}{4}\mathrm{c}\frac{\alpha_f \alpha \rho_f |\mathbf{u} - \mathbf{u}_f|}{d} (\mathbf{u} - \mathbf{u}_f) \,\mathrm{dV},$$

where $\mathbf{n}$ denotes the unit normal vector pointing outwards from $\partial\Omega$. In this weak form a solution to the velocity field $\mathbf{u} \in H^1(\Omega)^3$ was sought such that it is valid for all $\mathbf{w} \in H^1(\Omega)^3$. Dirichlet or Neumann boundary conditions for $\mathbf{u}$ could be enforced via the surface integrals.

B2 Basis functions

Instead of searching the whole (infinite) function space $H^1(\Omega)^3$ for a solution, the space of test functions and solutions were restricted to a finite-dimensional subspace $H_h^1(\Omega)^3 \subset H^1(\Omega)^3$ such that the test function and solution were represented by a linear combination of interpolating basis functions $\{\phi_k\}_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes}}}$ (Elman *et al.* 2005), where $N_{\mathbf{u}\_\mathrm{nodes}}$ is the number of velocity solution nodes. These basis functions can be continuous or discontinuous across the small subdomains of the mesh, called elements, where two basis functions overlap. Furthermore, basis functions have limited support between nodes, such that ϕ_k has a value of unity only at node k, and a value of zero at all other nodes (Elman *et al.* 2005) in order to perform the interpolation. Hence, the solution is essentially formed by piecing together the interpolating polynomials and the coefficients across each element in the domain.

This work used discontinuous piecewise linear basis functions (also known as P1$_{\mathrm{DG}}$ basis functions) for the velocity field such that solution nodes were not shared between elements; each element was essentially an independent problem. Within an element e the test function and solution were therefore given by

$$\mathbf{w} = \sum_{j=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \phi_j \mathbf{w}_j, \tag{B.3}$$

$$\mathbf{u} = \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \phi_k \mathbf{u}_k, \tag{B.4}$$

where $N_{\mathbf{u}\_\mathrm{nodes},e}$ is the number of velocity solution nodes in element e, and $\mathbf{w}_j$ and $\mathbf{u}_k$ are the values of the test function and solution at node j and node k respectively. Note that in the summations j and k are the local node index with respect to the element e, not the global node index with respect to the entire domain Ω, because each element was considered individually when using discontinuous basis functions.

When seeking the coefficients $\mathbf{u}_k$, the pressure field p also needed to be solved for. In this work p was represented by its own set of continuous piecewise quadratic basis functions $\{\psi_l\}_{l=1}^{N_{p\_\mathrm{nodes}}} \subset H^1(\Omega)$ (also known as P2 basis functions) such that

$$p = \sum_{l=1}^{N_{p\_\mathrm{nodes}}} \psi_l p_l, \tag{B.5}$$

where p_l are the values of the pressure field at node l. Note the summation over all nodes of the domain due to the continuous nature of the basis functions. The particular choice of basis functions used for the velocity and pressure fields formed the P1$_{\mathrm{DG}}$-P2 element pair, which was chosen because of its desirable LBB stability property. Further details can be found in the article by Cotter *et al.* (2009).

Substituting (B.3), (B.4) and (B.5) into (B.2), and using the fact that the vectors $\mathbf{w}_j$ were arbitrary (Zienkiewicz and Taylor 2000), yielded the discretised version of the weak form:

$$\begin{aligned}
&\sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \int_{\Omega_e} \phi_j \alpha\rho\phi_k \,\mathrm{dV}\, \frac{\partial \mathbf{u}_k}{\partial t} - \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \int_{\Omega_e} \nabla\phi_j \cdot \alpha\rho\mathbf{u}\phi_k \,\mathrm{dV}\, \mathbf{u}_k \\
&- \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \int_{\Omega_e} \phi_j \nabla\cdot(\alpha\rho\mathbf{u})\,\phi_k \,\mathrm{dV}\, \mathbf{u}_k + \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \int_{\partial\Omega_e} (\phi_j \mathbf{u}\alpha\rho)\cdot \mathbf{n}_e \mathbf{u}|_{\partial\Omega_e} \,\mathrm{dS} \\
&+ \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \int_{\Omega_e} \nabla\phi_j \cdot \alpha\mu\nabla\phi_k \,\mathrm{dV}\, \mathbf{u}_k - \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \int_{\partial\Omega_e} \phi_j \alpha\mu\mathbf{n}_e \left(\nabla\mathbf{u}\right)|_{\partial\Omega_e} \,\mathrm{dS} \qquad \text{(B.6)} \\
&+ \sum_{k=1}^{N_{\mathbf{u}\_\mathrm{nodes},e}} \int_{\Omega_e} \phi_j \frac{3}{4}\mathrm{c}\frac{\alpha_f\alpha\rho_f|\mathbf{u}-\mathbf{u}_f|}{d}\phi_k \,\mathrm{dV}\, \mathbf{u}_k = - \sum_{l=1}^{N_{p\_\mathrm{nodes}}} \int_{\Omega} \phi_j \alpha\nabla\psi_l \,\mathrm{dV}\, p_l + \int_{\Omega_e} \phi_j \alpha\rho\mathbf{g} \,\mathrm{dV} \\
&\qquad + \int_{\Omega_e} \phi_j \frac{3}{4}\mathrm{c}\frac{\alpha_f\alpha\rho_f|\mathbf{u}-\mathbf{u}_f|}{d}\mathbf{u}_f \,\mathrm{dV},
\end{aligned}$$

for all ϕ_j in each element e. The integrals involving the velocity field were restricted to the domain and boundary of element e, denoted Ω_e and $\partial\Omega_e$ respectively, because of the discontinuities in the basis functions used. However, in order for the surface integrals to be well-defined, this work evaluated $\mathbf{u}|_{\partial\Omega_e}$ using upwinding (see the work by Wilson (2009) for more information) and the derivative $\left(\nabla\mathbf{u}\right)|_{\partial\Omega_e}$ using the scheme of Bassi and Rebay (1997).

B3 System of linear equations

The discretised momentum equation (B.6) gave an $N_{\mathbf{u}\_\mathrm{nodes}} \times N_{\mathbf{u}\_\mathrm{nodes}}$ system of linear equations that could be assembled and solved for the vectors of unknown coefficients $\mathbf{u}$ and p:

$$\mathbf{M}\frac{\partial \mathbf{u}}{\partial t} + \mathbf{A}\mathbf{u} + \mathbf{K}\mathbf{u} + \mathbf{F}_{\text{left}}\mathbf{u} = \mathbf{C}p + \mathbf{b} + \mathbf{f}_{\text{right}}, \tag{B.7}$$

where the matrices $\mathbf{M}$, $\mathbf{A}$, $\mathbf{K}$ and $\mathbf{C}$ are the mass, advection, stress and gradient matrices respectively. The matrix $\mathbf{F}_{\text{left}}$ contains the left-hand side part of the drag term. The vectors $\mathbf{b}$ and $\mathbf{f}_{\text{right}}$ represent the buoyancy force and the right-hand side part of the drag term. These terms are defined as

$$\mathbf{M}_{jk} = \int_{\Omega_e} \phi_j \alpha\rho\phi_k \,\mathrm{dV}, \tag{B.8}$$

$$\begin{aligned}
\mathbf{A}_{jk} = &- \int_{\Omega_e} \nabla\phi_j \cdot \alpha\rho\mathbf{u}\phi_k \,\mathrm{dV} - \int_{\Omega_e} \phi_j \nabla\cdot(\alpha\rho\mathbf{u})\,\phi_k \,\mathrm{dV} \\
&+ \int_{\partial\Omega_e} (\phi_j \mathbf{u}\alpha\rho)\cdot \mathbf{n}_e \mathbf{u}|_{\partial\Omega_e} \,\mathrm{dS},
\end{aligned} \tag{B.9}$$

$$\mathbf{K}_{jk} = \int_{\Omega_e} \nabla\phi_j \cdot \alpha\mu\nabla\phi_k \,\mathrm{dV} - \int_{\partial\Omega_e} \phi_j \alpha\mu\mathbf{n}_e \left(\nabla\mathbf{u}\right)|_{\partial\Omega_e} \,\mathrm{dS}, \tag{B.10}$$

$$\mathbf{F}_{\text{left},jk} = \int_{\Omega_e} \phi_j \frac{3}{4}\mathrm{c}\frac{\alpha_f \alpha \rho_f |\mathbf{u} - \mathbf{u}_f|}{d} \phi_k \,\mathrm{dV}, \tag{B.11}$$

$$\mathbf{C}_{jk} = -\int_{\Omega} \phi_j \alpha \nabla \psi_k \,\mathrm{dV}, \tag{B.12}$$

$$\mathbf{b}_j = \int_{\Omega_e} \phi_j \alpha \rho \mathbf{g} \,\mathrm{dV}, \tag{B.13}$$

$$\mathbf{f}_{\text{right},j} = \int_{\Omega_e} \phi_j \frac{3}{4}\mathrm{c}\frac{\alpha_f \alpha \rho_f |\mathbf{u} - \mathbf{u}_f|}{d} \mathbf{u}_f \,\mathrm{dV}. \tag{B.14}$$

The solution to the discretised momentum equation also needed to satisfy the discrete continuity equation

$$\sum_{i=1}^{N_{\text{phases}}} \left(\mathbf{C}_i^{\mathrm{T}} \mathbf{u}_i - \mathbf{r}_i\right) = 0, \tag{B.15}$$

which is the discretised version of the weak form of (1) after being integrated by parts. It can be shown that the matrix $\mathbf{C}^{\mathrm{T}}$ acts as a divergence operator (Gresho and Sani 2000). The surface integral resulting from the integration by parts formed the vector **r** through which Dirichlet velocity boundary conditions could be applied:

$$\mathbf{r}_j = -\int_{\partial\Omega} \alpha \psi_j \mathbf{u} \cdot \mathbf{n} \,\mathrm{dS}. \tag{B.16}$$

The volume fraction fields α_p and α_f were discretised separately using a node-centred control volume approach (Wilson 2009) and advected with the velocity fields (once known). The face values of each control volume were limited using the Sweby flux limiter (Sweby 1984).

APPENDIX C: SOLUTION METHOD

The solution method began by considering the momentum equation that had been discretised in space using the Galerkin finite element method, and in time using the backward Euler method:

$$\mathbf{M}_i(\alpha_i^{n+1})\frac{\mathbf{u}_i^{n+1} - \mathbf{u}_i^n}{\Delta t} + \mathbf{A}_i(\alpha_i^{n+1}, \mathbf{u}_i^{n+1})\mathbf{u}_i^{n+1} + \mathbf{K}_i(\alpha_i^{n+1})\mathbf{u}_i^{n+1}$$
$$+\mathbf{F}_{\text{left},i}(\alpha_i^{n+1}, \mathbf{u}_i^{n+1}, \mathbf{u}_f^{n+1})\mathbf{u}_i^{n+1} = \mathbf{C}_i(\alpha_i^{n+1})p^{n+1} + \mathbf{b}_i(\alpha_i^{n+1}) + \mathbf{f}_{\text{right},i}(\alpha_i^{n+1}, \mathbf{u}_i^{n+1}, \mathbf{u}_f^{n+1}), \tag{C.1}$$

where $\mathbf{u}_i^{n+1}$ and p^{n+1} are to be found. The arguments of the matrices and vectors highlight the non-linearity coming from the dependence on the volume fraction, and also the velocity in the advection and drag terms, at time $n + 1$.

Each time-step (from time n to $n + 1$) was broken down into a series of Picard iterations to deal with the non-linearity in the system. This yielded a set of tentative results for $\mathbf{u}_i^{n+1}$ and p^{n+1}, denoted by $\mathbf{u}_i^{\mathrm{tent}}$ and p^{tent}. Therefore, instead of seeking a solution to (C.1) for a whole time-step, the method sought a new tentative solution at each Picard iteration to

$$\begin{aligned}\mathbf{M}_i(\tilde{\alpha}_i^{n+1})\frac{\mathbf{u}_i^{\mathrm{tent}} - \mathbf{u}_i^n}{\Delta t} + \mathbf{A}_i(\tilde{\alpha}_i^{n+1}, \tilde{\mathbf{u}}_i^{n+1})\mathbf{u}_i^{\mathrm{tent}} + \mathbf{K}_i(\tilde{\alpha}_i^{n+1})\mathbf{u}_i^{\mathrm{tent}} \\ +\mathbf{F}_{\mathrm{left},i}(\tilde{\alpha}_i^{n+1}, \tilde{\mathbf{u}}_i^{n+1}, \tilde{\mathbf{u}}_f^{n+1})\mathbf{u}_i^{\mathrm{tent}} = \mathbf{C}_i(\tilde{\alpha}_i^{n+1})p^{\mathrm{tent}} + \mathbf{b}_i(\tilde{\alpha}_i^{n+1}) + \mathbf{f}_{\mathrm{right},i}(\tilde{\alpha}_i^{n+1}, \tilde{\mathbf{u}}_i^{n+1}, \tilde{\mathbf{u}}_f^{n+1}),\end{aligned} \tag{C.2}$$

such that the discrete continuity equation

$$\sum_{i=1}^{N_{\mathrm{phases}}} \left(\mathbf{C}_i^{\mathrm{T}}(\tilde{\alpha}_i^{n+1})\mathbf{u}_i^{\mathrm{tent}} - \mathbf{r}_i(\tilde{\alpha}_i^{n+1})\right) = 0, \tag{C.3}$$

was satisfied. At the beginning of an iteration, the latest (best available) tentative solution was used to compute the non-linear approximations to the phase volume fraction, velocity of phase i and velocity of the fluid phase f, denoted by $\tilde{\alpha}_i^{n+1}$, $\tilde{\mathbf{u}}_i^{n+1}$ and $\tilde{\mathbf{u}}_f^{n+1}$ respectively, as follows:

$$\tilde{\mathbf{u}}_i^{n+1} = \theta_{\mathrm{nl}}\mathbf{u}_i^{\mathrm{tent}} + (1 - \theta_{\mathrm{nl}})\mathbf{u}_i^n, \tag{C.4}$$

$$\tilde{\mathbf{u}}_f^{n+1} = \theta_{\mathrm{nl}}\mathbf{u}_f^{\mathrm{tent}} + (1 - \theta_{\mathrm{nl}})\mathbf{u}_f^n, \tag{C.5}$$

$$\tilde{\alpha}_i^{n+1} = \theta_{\mathrm{nl}}\alpha_i^{\mathrm{tent}} + (1 - \theta_{\mathrm{nl}})\alpha_i^n. \tag{C.6}$$

Initial values for $\mathbf{u}_i^{\mathrm{tent}}$, $\mathbf{u}_f^{\mathrm{tent}}$ and α_i^{tent} were set at the beginning of each time-step: $\mathbf{u}_i^{\mathrm{tent}} = \mathbf{u}_i^n$, $\mathbf{u}_f^{\mathrm{tent}} = \mathbf{u}_f^n$ and $\alpha_i^{\mathrm{tent}} = \alpha_i^n$. For all the simulations described in this article, $\theta_{\mathrm{nl}} = 0.5$. Note also that from this point on the arguments of the matrices and vectors will be dropped for clarity.

To find the new tentative solution, the method first made a ‘best guess’ for p^{tent}, denoted p^*, either by solving a pressure Poisson equation or by using the most up-to-date pressure field available. Equation (C.7) was then solved for each phase to obtain a set of intermediate velocities $\mathbf{u}_i^*$.

$$\mathbf{M}_i\frac{\mathbf{u}_i^* - \mathbf{u}_i^n}{\Delta t} + \mathbf{A}_i\mathbf{u}_i^* + \mathbf{K}_i\mathbf{u}_i^* + \mathbf{F}_{\mathrm{left},i}\mathbf{u}_i^* = \mathbf{C}_i p^* + \mathbf{b}_i + \mathbf{f}_{\mathrm{right},i}. \tag{C.7}$$

In general, the intermediate velocities $\mathbf{u}_i^*$ do not satisfy the continuity equation (B.15) (i.e. $\sum_{i=1}^{N_{\mathrm{phases}}} \left(\mathbf{C}_i^{\mathrm{T}}\mathbf{u}_i^* - \mathbf{r}_i\right) \neq 0$) because of the guess used for the pressure. However, the velocities $\mathbf{u}_i^{\mathrm{tent}}$ which, by definition, should satisfy $\sum_{i=1}^{N_{\mathrm{phases}}} \left(\mathbf{C}_i^{\mathrm{T}}\mathbf{u}_i^{\mathrm{tent}} - \mathbf{r}_i\right) = 0$ could be found from a Helmholtz decomposition of $\mathbf{u}_i^*$; this split the vector up into a divergence-free and a curl-free component:

$$\mathbf{u}_i^* = \mathbf{u}_i^{\mathrm{tent}} + \nabla\lambda_i, \tag{C.8}$$

where $\nabla \lambda_i$ is an unknown vector. By choosing the form $\nabla \lambda_i = \frac{\Delta t}{\rho_i} \nabla \left(p^{\text{tent}} - p^*\right)$, (C.8) was rearranged and discretised as follows:

$$\mathbf{M}_i \frac{\mathbf{u}_i^{\text{tent}} - \mathbf{u}_i^*}{\Delta t} = \mathbf{C}_i \left(p^{\text{tent}} - p^*\right). \tag{C.9}$$

The next step of the projection method sought the pressure correction term $\Delta p = (p^{\text{tent}} - p^*)$. Inverting $\mathbf{M}_i$ in (C.9) and multiplying both sides by the divergence matrix $\mathbf{C}_i^{\text{T}}$ gave

$$\mathbf{C}_i^{\text{T}} \left(\mathbf{u}_i^{\text{tent}} - \mathbf{u}_i^*\right) = \Delta t \mathbf{C}_i^{\text{T}} \mathbf{M}_i^{-1} \mathbf{C}_i \Delta p. \tag{C.10}$$

Using the fact that $\sum_{i=1}^{N_{\text{phases}}} \left(\mathbf{C}_i^{\text{T}} \mathbf{u}_i^{\text{tent}} - \mathbf{r}_i\right) = 0$ (from the discrete continuity equation), Δp was obtained by solving

$$\sum_{i=1}^{N_{\text{phases}}} \left(\mathbf{r}_i - \mathbf{C}_i^{\text{T}} \mathbf{u}_i^*\right) = \Delta t \left(\sum_{i=1}^{N_{\text{phases}}} \mathbf{C}_i^{\text{T}} \mathbf{M}_i^{-1} \mathbf{C}_i \right) \Delta p. \tag{C.11}$$

Hence, Δp was found by projecting the intermediate velocities onto a divergence-free space.

The velocities $\mathbf{u}_i^*$ could now be corrected (to obtain $\mathbf{u}_i^{\text{tent}}$) by substituting in the recently found pressure correction term into

$$\mathbf{u}_i^{\text{tent}} = \mathbf{u}_i^* + \Delta t \mathbf{M}_i^{-1} \mathbf{C}_i \Delta p. \tag{C.12}$$

Once these new tentative results for $\mathbf{u}_i^{n+1}$ and p^{n+1} were found, the Picard iteration was complete. Any tracer fields (including the phase volume fraction fields) were then advected using the new tentative velocities $\mathbf{u}_i^{\text{tent}}$, thus providing the best available solutions for all fields in the next iteration.

It is important to note that although the corrected velocities $\mathbf{u}_i^{\text{tent}}$ satisfied the continuity equation, they only satisfied the following version of the momentum equation:

$$\mathbf{M}_i \frac{\mathbf{u}_i^{\text{tent}} - \mathbf{u}_i^n}{\Delta t} + \mathbf{A}_i \mathbf{u}_i^* + \mathbf{K}_i \mathbf{u}_i^* + \mathbf{F}_{\text{left},i} \mathbf{u}_i^* = \mathbf{C}_i p^{\text{tent}} + \mathbf{b}_i + \mathbf{f}_{\text{right},i}. \tag{C.13}$$

because the advection, stress and drag terms were not taken into account in the correction from $\mathbf{u}_i^*$ to $\mathbf{u}_i^{\text{tent}}$. This momentum equation is not quite the same as (C.2), which is why Picard iterations are required to converge to a set of velocities that satisfy both (C.2) and (C.3).

Once a desired Picard iteration limit or convergence was reached, $\mathbf{u}_i^{n+1}$ and p^{n+1} took the values of the final tentative solution and the time-step was deemed complete. The above solution method was then repeated until a desired time limit or steady state was attained.