

Development of the meshless finite volume particle method with exact and efficient calculation of interparticle area



Nathan J. Quinlan^{*}, Libor Lobovský¹, Ruairi M. Nestor²

Mechanical and Biomedical Engineering, National University of Ireland Galway, Ireland

ARTICLE INFO

Article history:

Received 26 July 2013

Received in revised form

13 January 2014

Accepted 10 February 2014

Available online 18 February 2014

Keywords:

Finite volume particle method

Meshless method

Taylor–Green flow

Shock tube

NACA 0012

Rayleigh–Taylor instability

ABSTRACT

The Finite Volume Particle Method (FVPM) is a meshless method based on a definition of interparticle area which is closely analogous to cell face area in the classical finite volume method. In previous work, the interparticle area has been computed by numerical integration, which is a source of error and is extremely expensive. We show that if the particle weight or kernel function is defined as a discontinuous top-hat function, the particle interaction vectors may be evaluated exactly and efficiently. The new formulation reduces overall computational time by a factor between 6.4 and 8.2. In numerical experiments on a viscous flow with an analytical solution, the method converges under all conditions. Significantly, in contrast with standard FVPM and SPH, error depends on particle size but not on particle overlap (as long as the computational domain is completely covered by particles). The new method is shown to be superior to standard FVPM for shock tube flow and inviscid steady transonic flow. In benchmarking on a viscous multiphase flow application, FVPM with exact interparticle area is shown to be competitive with a mesh-based volume-of-fluid solver in terms of computational time required to resolve the structure of an interface.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Meshless particle methods have been successful in a diverse range of fluid dynamics applications [1,2]. The meshless approach is beneficial not only because it enables mesh generation to be bypassed, but because it permits Lagrangian or arbitrary Lagrange Euler (ALE) discretisation without the risk of excessive mesh deformation. In particular, smoothed particle hydrodynamics (SPH) is widely and successfully used to simulate gravity-driven free-surface flow [3]. The finite volume particle method (FVPM), formulated by Hietel et al. [4] and independently by Ismagilov [5,6], is a relatively new meshless approach based on a generalisation of the classical finite volume method (FVM) [7]. It provides the flexibility of a particle method while retaining the strict conservation properties of the FVM and accommodates standard FVM algorithms such

as numerical flux functions in its meshless framework. The present work is concerned with a method to enhance accuracy and reduce computational cost of the finite volume particle method.

In contrast with SPH, FVPM ensures conservation in particle–particle exchanges regardless of any variation in particle size and interparticle distance. It incorporates boundary conditions through the straightforward prescription of flux on any boundary segment that lies inside a particle support; most variants of SPH require fictitious particles.

In contrast with the classical finite volume method, on the other hand, FVPM is based on overlapping finite volume “particles” in place of contiguous non-overlapping cells. Since particles have no explicit connectivity, they may move according to an arbitrary particle transport velocity field (typically, but not necessarily, Lagrangian) without the need for remeshing or management of mesh deformation. This enables simple treatment of interfaces and moving boundaries. In the mesh-based FVM, Lagrangian capability can be achieved through remeshing operations (e.g. Voronoi tessellation [8,9]). However, such an approach involves discontinuous changes in the topology of the numerical stencil, even in the limit of zero timestep. In FVPM, in contrast, the particle overlap changes in a continuous manner.

In FVPM, in regions of particle overlap, the volume is distributed among the overlapping particles according to their

^{*} Corresponding author. Tel.: +353 91492726.

E-mail addresses: nathan.quinlan@nuigalway.ie (N.J. Quinlan), lobo@kme.zcu.cz (L. Lobovský), ruairi.nestor@mcskenny.com (R.M. Nestor).

¹ Present address: NTIS—New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 22, 30614 Plzeň, Czech Republic.

² Present address: MCS Kenny, Parkmore, Galway, Ireland.

respective weighting functions. The exchange of conserved quantities between each pair of particles is weighted by an interparticle area vector (previously referred to as the geometric coefficient [4] or the particle interaction vector [10]) which is closely analogous to the discrete cell face area in the mesh finite volume method. Interparticle area is defined in terms of an integral of weight functions of all overlapping particles. Previous implementations of FVPM involved a numerical evaluation of this integral, which is a source of numerical error and is by far the most computationally costly part of the algorithm. This quadrature operation makes FVPM considerably more expensive than SPH, which in turn is more expensive than mesh-based methods.

Quinlan and Nestor [11] proposed an alternative formulation which allows interparticle area to be determined both exactly and quickly. In the present work, we describe further development of the new formulation. Extensive investigation of its convergence properties is presented, and its performance is characterised in a variety of applications. The new variant of FVPM is compared with an established mesh-based method in terms of both accuracy and computational cost. In Section 2, FVPM is briefly introduced. The new formulation is presented in Section 3. Some implementation aspects are presented in Section 4, along with results for computational cost. Performance in various test problems is demonstrated and assessed in Section 5.

2. The finite volume particle method

FVPM is described briefly here for application to the Navier–Stokes equations. Detailed derivations and analysis are given by Keck and Hietel [12], Teleaga and Struckmeier [13], and Junk [7]. The governing equations to be solved are of the form

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = 0, \quad (1)$$

where \mathbf{U} is the vector of conserved variables and \mathbf{F} is the flux function. To derive the FVPM, the system (1) is multiplied by a test function $\psi_i(\mathbf{x}, t)$ and then integrated over the domain Ω to yield the following weak formulation:

$$\begin{aligned} \int_{\Omega} \psi_i(\mathbf{x}, t) \frac{\partial \mathbf{U}}{\partial t} d\mathbf{x} - \int_{\Omega} \nabla \psi_i(\mathbf{x}, t) \cdot \mathbf{F}(\mathbf{U}) d\mathbf{x} \\ + \int_{\partial\Omega} \psi_i(\mathbf{x}, t) \mathbf{F}(\mathbf{U}) \cdot \mathbf{n} dS = 0. \end{aligned} \quad (2)$$

The test function $\psi_i(\mathbf{x}, t)$ is non-zero only in the region Ω_i , which is called the support of particle i . The spatial domain is fully covered by particle supports, which may overlap, i.e. $\Omega = \bigcup_i \Omega_i$. The final integral above is a boundary term in which \mathbf{n} is an outward normal on the boundary $\partial\Omega$, and S is boundary surface area.

The test function $\psi_i(\mathbf{x}, t)$ is constructed as

$$\psi_i(\mathbf{x}, t) = \frac{W_i(\mathbf{x})}{\sum_j W_j(\mathbf{x})}, \quad (3)$$

where the particle weight function $W_i(\mathbf{x}, t)$ is zero outside Ω_i .

Eq. (2) can be written for each particle i as

$$\begin{aligned} \frac{d}{dt} \int_{\Omega} \psi_i(\mathbf{x}, t) \mathbf{U} d\mathbf{x} - \int_{\Omega} \left(\nabla \psi_i(\mathbf{x}, t) \cdot \mathbf{F}(\mathbf{U}) + \mathbf{U} \frac{\partial \psi_i}{\partial t} \right) d\mathbf{x} \\ + \int_{\partial\Omega} \psi_i(\mathbf{x}, t) \mathbf{F}(\mathbf{U}) \cdot \mathbf{n} dS = 0. \end{aligned} \quad (4)$$

At this stage, as in the classical finite volume method, the approximation is introduced that \mathbf{U} varies little within the overlap of particles i and j , and can be represented by a single value. The flux \mathbf{F} is approximated with a numerical flux function $\mathcal{F}(\mathbf{U}_i, \mathbf{U}_j)$.

Furthermore, to account for flux due to particle motion, an overall numerical flux is defined as $\mathcal{G}(\mathbf{U}_i, \mathbf{U}_j, \dot{\mathbf{x}}_i, \dot{\mathbf{x}}_j) = \mathcal{F}(\mathbf{U}_i, \mathbf{U}_j) - \dot{\mathbf{x}}_{ij} \mathbf{U}_{ij}$, where $\dot{\mathbf{x}}_i$ is the arbitrary particle transport velocity and $\dot{\mathbf{x}}_{ij}$ and \mathbf{U}_{ij} are simple averages over the two particles.

The vectors γ_{ij} , β_{ij} and β_i^b are now defined as follows:

$$\gamma_{ij} = \int_{\Omega} \frac{W_i(\mathbf{x}) \nabla W_j(\mathbf{x})}{\left(\sum_j W_j(\mathbf{x}) \right)^2} d\mathbf{x} \quad (5)$$

$$\beta_{ij} = \gamma_{ij} - \gamma_{ji} \quad (6)$$

$$\beta_i^b = \int_{\partial\Omega} \frac{W_i(\mathbf{x})}{\sum_j W_j(\mathbf{x})} \mathbf{n} dS. \quad (7)$$

The integrand in Eq. (5) is non-zero only in the region where both $W_i(\mathbf{x})$ and $\nabla W_j(\mathbf{x})$ are non-zero, that is, in the overlap of particles i and j .

Manipulation of Eq. (4) now yields the defining equation of the semi-discrete FVPM [12]:

$$\frac{d}{dt} \int_{\Omega} \psi_i(\mathbf{x}, t) \mathbf{U} d\mathbf{x} + \sum_j \beta_{ij} \cdot \mathcal{G}(\mathbf{U}_i, \mathbf{U}_j, \dot{\mathbf{x}}_i, \dot{\mathbf{x}}_j) + \beta_i^b \cdot \mathcal{G}_i^b = 0. \quad (8)$$

The only differences between Eq. (8) and the mesh-based finite volume method are the vectors $\sum_j \beta_{ij}$ and $\sum_j \beta_i^b$, which take the place of cell–cell and cell–boundary interface areas, respectively, in the classical method. Eqs. (5) and (6) imply that $\sum_j \beta_{ij} = 0$ (analogous to the requirement for faces of a cell to form a closed surface), and $\beta_{ij} = -\beta_{ji}$, which ensures that interparticle exchanges are exactly conservative. These properties are exploited in the derivation of Eq. (8) above. The vector β_{ij} can therefore be interpreted as an interface area between overlapping volumes Ω_i and Ω_j , in precise analogy with the interface area between contiguous but non-overlapping volume cells in a mesh. In the present work β_{ij} will be referred to as the interparticle area. Monotonicity [14], stability [14] and Lax–Wendroff consistency [15] have been proven for FVPM. Junk [7] has shown that FVPM reduces exactly to the classical finite volume method in the special case where particles are contiguous but do not overlap.

In previous work, cubic [4], quadratic [10], piecewise quadratic [13] and piecewise linear [7] functions have been used for the weight function $W_i(\mathbf{x})$. With polynomial $W_i(\mathbf{x})$, the integral in Eq. (5) cannot be evaluated analytically, and has invariably been approximated by numerical quadrature. This has important consequences, as discussed in detail in later sections. Two techniques have been proposed to reduce the impact of quadrature-related numerical error in β_{ij} [16,13].

The scheme used in the present work is similar in detail to that of Nestor et al. [10], except where otherwise stated. A linear MUSCL procedure [17], in some cases with the Barth–Jespersen slope limiter [18], is used to reconstruct data to particle interfaces. Inviscid fluxes are computed using the AUSM⁺ [19] or AUSM⁺-up [20] scheme, and viscous stresses are based on velocity gradients calculated using the first-order consistent corrected SPH formulation of Bonet and Lok [21]. Particles may be advected with a Lagrangian or ALE velocity field, or may be stationary, according to the requirements of the application. For example, Lagrangian particles may be chosen to model a phase interface or free surface. Eulerian particles may be appropriate if the motivation for a meshless approach is simply to avoid mesh generation, as in meshless methods for external aerodynamics [22]. In some cases, particles are transported in a corrected Lagrangian velocity field, with a small correction velocity added to maintain smooth particle distribution [23,10]. A second-order Runge–Kutta time stepping

scheme is used, with time step determined by a CFL criterion

$$\Delta t = C \min_i \frac{2V_i}{\sum_j ((\mathbf{u}_i + \mathbf{u}_j) \cdot \boldsymbol{\beta}_{ij} + (a_i + a_j)|\boldsymbol{\beta}_{ij}|)}, \quad (9)$$

where C is the Courant number (usually 0.9), a is the local speed of sound and V is the particle volume.

When interparticle area $\boldsymbol{\beta}_{ij}$ is evaluated by numerical quadrature, the following weight function is used:

$$W(\mathbf{x} - \mathbf{x}_i, h) = \begin{cases} 4 - (\|\mathbf{x} - \mathbf{x}_i\|/h)^2 & \|\mathbf{x} - \mathbf{x}_i\| \leq 2h \\ 0 & \text{otherwise,} \end{cases} \quad (10)$$

where h is the smoothing length, defined (following the SPH convention) as one quarter of the width of the particle support. The numerical integration is performed by Gaussian quadrature with (unless otherwise stated) an array of 6×6 integration points in the particle overlap region. An alternative method for the evaluation of $\boldsymbol{\beta}_{ij}$ is the main subject of this article, and is presented in Section 3.

3. Exact evaluation of interparticle area

A novel method for the determination of the interparticle area $\boldsymbol{\beta}_{ij}$ was proposed by Quinlan and Nestor [11]. The motivation is to reduce both the error and computational time requirements of quadrature. First, we summarise the earlier work for the reader's convenience. A new particle weight function $W_i(\mathbf{x})$ is defined as

$$W_i(\mathbf{x}) = \begin{cases} 1 & (\mathbf{x} \in \Omega_i) \\ 0 & \text{otherwise.} \end{cases} \quad (11)$$

The resulting normalisation function $\sum_k W_k(\mathbf{x})$, denoted by $\sigma(\mathbf{x})$, is simply equal to the number of particles covering the point \mathbf{x} . The FVPM, as formulated with this new top-hat weight function, can be regarded as a hybrid of the original FVPM (with smooth weight functions on overlapping particles) and the mesh-based finite volume method (which has constant-valued weight functions on non-overlapping cells).

When the top-hat weight function (11) is used, $W_i(\mathbf{x})\nabla W_j(\mathbf{x})$ is non-zero only on the subset of the boundary of Ω_j which lies inside Ω_i (denoted as $\partial\Omega_j \cap \Omega_i$). Although $\nabla W_j(\mathbf{x})$ is undefined on $\partial\Omega_j$, the integral can be evaluated by considering the limit of a weight function which is equal to 1 over most of the particle support, and varies between 0 and 1 within an edge region of finite thickness. In one dimension, the integral γ_{ij} is then

$$\gamma_{ij} = \frac{1}{\sigma^-} - \frac{1}{\sigma^- + 1}, \quad (12)$$

where σ^- is the number of particles, excluding particle j , that cover a point on the boundary of the support of particle j . (The treatment of coincident particle boundaries is explained in Section 4.)

In two dimensions, as illustrated in Fig. 1, the edge $\partial\Omega_j \cap \Omega_i$ is partitioned into segments $k = \{1 \dots m\}$ by intersections with boundaries of other particles at $m + 1$ points $\mathbf{x}_k^* = (x_{1,k}^*, x_{2,k}^*)$. Component α of γ_{ij} , for $\alpha = \{1, 2\}$, is given by integrating the 1D form (12) to obtain

$$\gamma_{ij,\alpha} = \sum_{k=1}^m \int_{x_{3-\alpha,k}^*}^{x_{3-\alpha,k+1}^*} \left(\frac{1}{\sigma_k^-} - \frac{1}{\sigma_k^- + 1} \right) dx_{3-\alpha}. \quad (13)$$

Within each segment k , the integrand in Eq. (13) is independent of $x_{3-\alpha}$. The equation therefore simplifies to

$$\gamma_{ij,\alpha} = \sum_{k=1}^m (x_{3-\alpha,k+1}^* - x_{3-\alpha,k}^*) \left(\frac{1}{\sigma_k^-} - \frac{1}{\sigma_k^- + 1} \right). \quad (14)$$

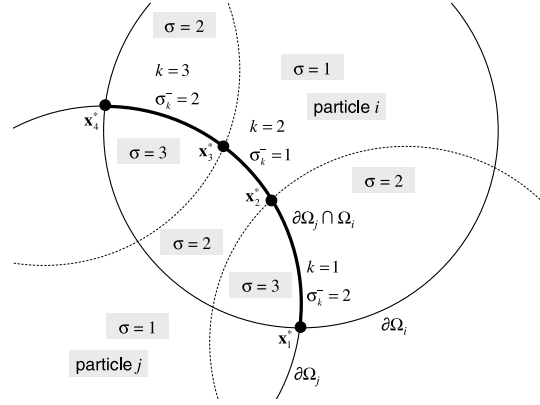


Fig. 1. Schematic diagram of the overlap of particles i and j and two neighbours in 2D, illustrating the notation for the calculation of γ_{ij} . The highlighted arc is $\partial\Omega_j \cap \Omega_i$, on which $W_j \nabla W_i \neq 0$. Labels on grey backgrounds refer to two-dimensional regions, and other labels refer to points or edge segments.

The integration in Eq. (7) can be treated in a similar manner to evaluate the particle-boundary area β_i^b .

Eq. (14) provides evaluation of interparticle area $\boldsymbol{\beta}_{ij}$ without the need for numerical approximations. The established method, quadrature evaluation of Eq. (5), entails evaluations of $W(\mathbf{x})$ for all covering particles at all quadrature points. The only information required for the new method is \mathbf{x}_k^* , the intersections of particle edge supports, and σ_k^- , the number of neighbour particles covering each edge segment. The computation of this geometric data is relatively inexpensive.

4. Implementation and performance

The original implementation of the new method for exact interparticle area [11] was validated by simulating a 2D Taylor–Green flow [24]. This flow consists of a spatially periodic decaying vortex described by the following exact solution of the Navier–Stokes equations:

$$u(x, y, t) = -u_0 \cos(2\pi x/L) \sin(2\pi y/L) e^{-\frac{8\nu\pi^2}{L^2}t} \quad (15)$$

$$v(x, y, t) = u_0 \sin(2\pi x/L) \cos(2\pi y/L) e^{-\frac{8\nu\pi^2}{L^2}t} \quad (16)$$

$$p(x, y, t) = -\frac{1}{4} [\cos(4\pi x) + \cos(4\pi y)] e^{-16\nu\pi^2 t}, \quad (17)$$

where u and v are the x and y components of velocity, p is pressure, u_0 is the maximum initial speed, ν is kinematic viscosity, and the domain has periodic boundaries at $x = \pm L/2$, $y = \pm L/2$. Total computational time was found to be faster with exact evaluation than with quadrature by a factor of 2.7–3.5.

In the present work, the implementation of the exact integration scheme has been refined to further reduce computational cost. As each particle boundary segment of particle j (as shown in Fig. 1) is covered by multiple particles i , its contribution to $\gamma_{ij,\alpha}$ can be computed just once and stored. In the Taylor–Green flow case, this results in a speedup of 6.4–8.2 with respect to quadrature-based FVPM for the same number of particles. A comparison of error as a function of computational time (determined by the number of particles) for exact and quadrature-based $\boldsymbol{\beta}_{ij}$ is shown in Fig. 2. As in the previous work, both the exact and quadrature-based methods display convergence close to third order in particle size over most of the range investigated. However, the quadrature-based solution fails to converge for the finest discretisations, while solutions with exact $\boldsymbol{\beta}_{ij}$ converge in all cases.

Although the evaluation of $\boldsymbol{\beta}_{ij}$ has been illustrated above with reference to circular particles, the procedure is applicable

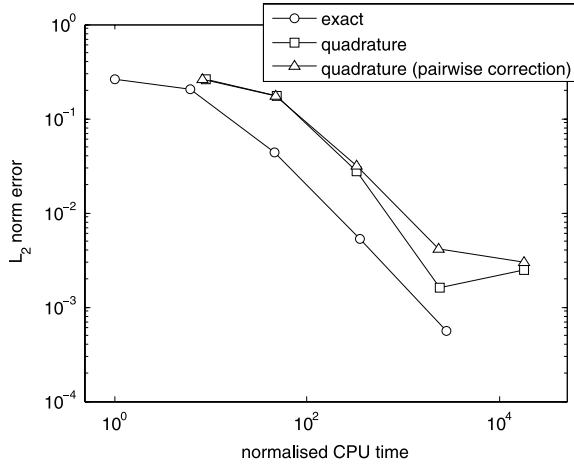


Fig. 2. Variation of L_2 norm velocity error with computational time in FVPM simulations of a Taylor–Green flow at various spatial resolutions, for three approaches to computation of interparticle area. The Reynolds number is 100 and error is evaluated at time $t = L^2/(8\nu\pi^2)$ (one time constant of velocity decay), with interparticle area computed by the new exact evaluation and by quadrature. Pairwise correction of interparticle area is by the method of Hietel and Keck [16], which restores exact conservation. Particle spacing $\Delta x/L$ ranges from 0.00625 to 0.1 for all three methods.

to any particle shape. The method has been implemented for square particles, which have the advantage over circular particles that the domain can be covered with fewer particles of a given width. Extension to square particles requires some care to deal with degenerate cases in which the edges of two neighbouring particles coincide exactly. In that case, the intersection of the particle boundaries cannot be determined. Such events are unlikely in the test cases presented in the present work, but in future large simulations, they are highly probable. For example, for 10^6 particles in 3D with 50 neighbours each, 10^5 time steps, and double precision calculation, the probability is approximately 0.1 that some pair of overlapping square particles will have one equal coordinate at some time (assuming randomly distributed particles within each neighbourhood). A consistent and robust method is required to detect exact coincidences of particle boundaries and deal with them when they arise. Therefore, to avoid ambiguity due to finite-precision arithmetic, particle boundary coordinates are computed once only and stored. If two overlapping particles have edges at equal coordinates, the intersection points are then computed as if the particle with the higher index (an arbitrary but unique identifier) has an infinitesimally higher coordinate. This provides a unique pair of intersections (or no intersection) between the particles. With this approach, it is possible to handle particle distributions constructed such that every neighbour pair has degenerate intersections.

The calculation of particle–particle intersections requires fewer floating-point operations for squares than for circles, but this is offset by the increased number of logical tests required to deal with corners and safeguard against degenerate cases. Consequently, the reduction in computation time for square particles is insignificant.

5. Results

5.1. Convergence of the spatial discretisation for Taylor–Green flow

Convergence of FVPM with exact interparticle area in Taylor–Green flow over long times has previously been shown ([11] and Section 4 above). In the present work, to evaluate the FVPM spatial discretisation independently of time integration, the Taylor–Green flow was simulated for one time step, for exact and quadrature-based interparticle area, circular and square particles,

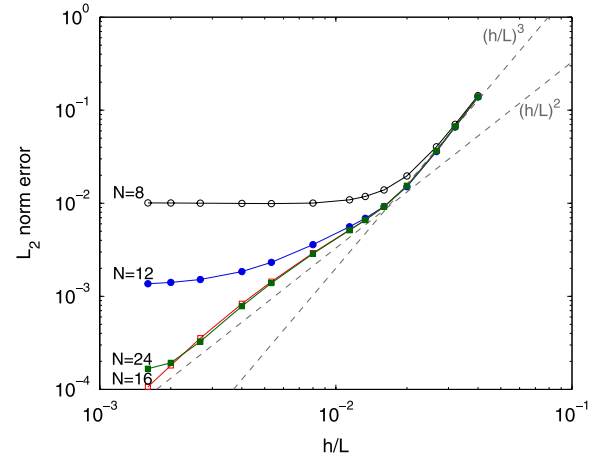


Fig. 3. Variation of L_2 norm acceleration error with particle size in FVPM simulations of Taylor–Green flow at $Re = 100$, with interparticle area based on quadrature, $N \times N$ integration points, circular particles, and $h/\Delta x = 0.8$.

various values of particle overlap, and various values of Reynolds number. Results are presented in terms of the computed local acceleration, based on the left-hand side of Eq. (8). Particles are distributed on a regular Cartesian grid at initial spacing Δx and initialised according to Eqs. (15)–(17) at $t = 0$ with Reynolds number $u_0 L/\nu = 100$.

The non-dimensional L_2 norm error in flow acceleration magnitude is evaluated at time $t = 0$ as

$$L_2 = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(\frac{a_i - \tilde{a}_i}{v_0^2/L} \right)^2} \quad (18)$$

where n is the number of particles, $a_i = \|\partial \mathbf{v}_i / \partial t\|$ is the computed magnitude of the local acceleration at particle i , and \tilde{a}_i is the exact value of acceleration magnitude computed from the time derivatives of the right hand sides of Eq. (15)–(16). Error is characterised for both numerical and exact integration in determination of β_{ij} , using weight functions (10) and (11) respectively. Results are also presented for circular and square particle supports, various values of non-dimensionalised particle size h/L , and various values of particle overlap parameter $h/\Delta x$. The fluid is modelled as weakly compressible, using the modified Tait equation of state $p = \rho_0 a_0^2/\gamma [(\rho/\rho_0)^\gamma - 1]$ where a_0 and ρ_0 are the speed of sound and density at $p = 0$, respectively, and $\gamma = 7$ [25]. To ensure that effects of compressibility are negligible in comparison with the discretisation errors of interest, the speed of sound a_0 is selected to give a Mach number u_0/a_0 of 0.01. It has been found that decreases in Mach number below this level have a negligible influence on solutions over the range of discretisations investigated.

5.1.1. Gaussian quadrature for interparticle area

The Taylor–Green flow was simulated using Gaussian quadrature to determine β_{ij} with the quadratic weight function defined in Eq. (10). The resulting L_2 norm error in acceleration is shown in Fig. 3 for Gaussian quadrature with circular particles. There is a component of error which depends on the number of quadrature points. When h/L is small, this integration error dominates the overall error, and is independent of particle size h . Over the range tested, it appears to be negligible for 16×16 quadrature points or more. With square particles (Fig. 4), however, the error displays sensitivity to numerical quadrature at $h/L \lesssim 5 \times 10^{-3}$ for all numbers of integration points. It should be noted that square particles may have more neighbours than circular particles of the

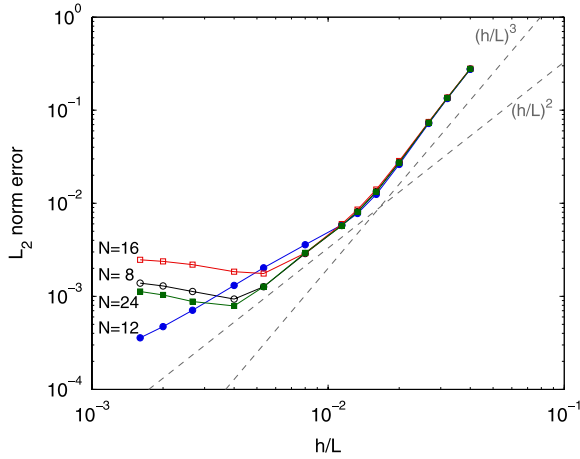


Fig. 4. Variation of L_2 norm acceleration error with particle size in FVPM simulations of Taylor–Green flow at $Re = 100$, with interparticle area based on quadrature, $N \times N$ integrating points, square particles, and $h/\Delta x = 0.8$.

same $h/\Delta x$, with a different spatial distribution of neighbouring particles and larger overlap regions.

The above result can be investigated theoretically using a simple model for the variation of quadrature-related error while particle size and particle distribution are scaled (i.e. h/L is varied while holding $h/\Delta x$ constant). The domain of integration in Eq. (5) and the distribution of integration points both scale with the particle size. Thus, the quadrature approximation to the integral scales in the same manner as the exact integral. For constant $h/\Delta x$ and constant shape of the local particle distribution, therefore, there is a constant ratio (i.e. independent of h and Δx) between the approximated β_{ij} and the exact value. Denoting this ratio as $1 + \epsilon_{ij}$, the semi-discrete FVPM (8) becomes (omitting boundary terms)

$$\frac{d}{dt} \int_{\Omega} \psi_i(\mathbf{x}, t) \mathbf{U} d\mathbf{x} + \sum_j \beta_{ij} \cdot \mathbf{g}_{ij} + \sum_j \epsilon_{ij} \beta_{ij} \cdot \mathbf{g}_{ij} = 0. \quad (19)$$

The final summation term contributes an error which is independent of particle size as long as $h/\Delta x$ and the quadrature scheme are held constant. As other error sources decay with reduction of h , this term comes to dominate, as observed in the numerical experiments described above.

The curves in Fig. 5 show that the FVPM solution with Gaussian quadrature on 16×16 points is highly sensitive to $h/\Delta x$. For small $h/\Delta x$ (i.e. small particle overlap and small number of neighbours), the error falls to a minimum which cannot be further reduced by reducing h/L . In other words, the numerical solution does not converge to the analytical solution for fixed $h/\Delta x$.

5.1.2. Exact interparticle area

Results are shown in Fig. 6 for circular particles with exact evaluation of β_{ij} as described in Section 3. Convergence is apparent over the whole tested range of $h/\Delta x$ values and error is nearly independent of $h/\Delta x$. The minimum overlap tested is $h/\Delta x = 0.3540$, which is slightly greater than the minimum required to cover 2D space with circular particles without gaps ($h/\Delta x = 1/\sqrt{8} \approx 0.3536$). This result suggests that minimum particle overlap is sufficient in FVPM with exact interparticle area, unlike both FVPM with Gaussian quadrature (Fig. 5) and SPH [26].

These observations have also been confirmed for square particle supports. A direct comparison of results for circular and square supports is presented in Fig. 7, showing that the convergence trends of FVPM with exact β_{ij} are independent of particle support shape and overlap (or number of neighbours).

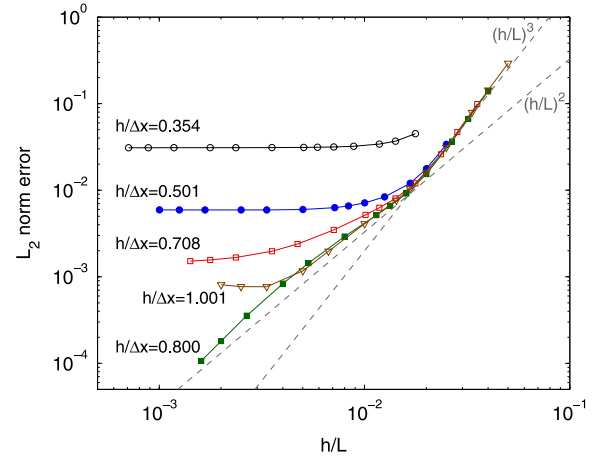


Fig. 5. Variation of L_2 norm acceleration error with particle size in FVPM simulations of Taylor–Green flow at $Re = 100$, with interparticle area based on quadrature, 16×16 integrating points, circular particles, and a range of $h/\Delta x$.

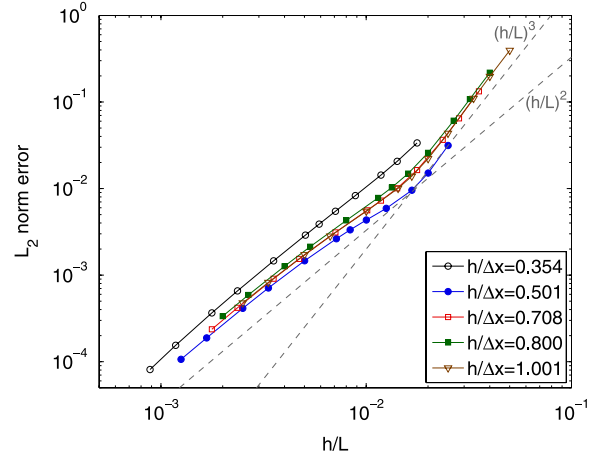


Fig. 6. Variation of L_2 norm velocity error with particle size in FVPM simulations of Taylor–Green flow at $Re = 100$, with interparticle area based on exact integration, circular particles, and a range of $h/\Delta x$.

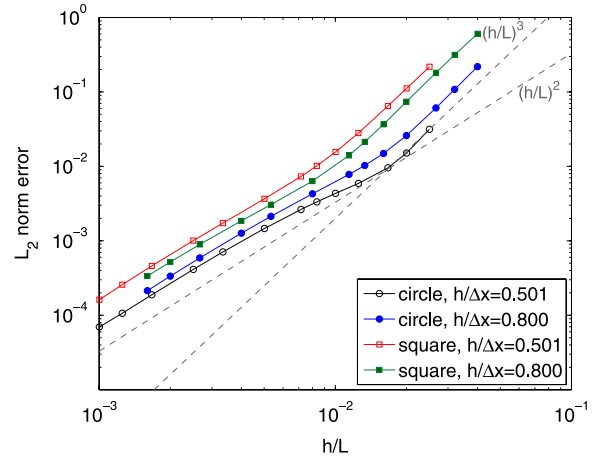


Fig. 7. Variation of L_2 norm acceleration error with particle size in FVPM simulations of Taylor–Green flow at $Re = 100$, with interparticle area based on exact integration, for circular and square particles, and two values of $h/\Delta x$.

FVPM solutions with exact interparticle area display third-order convergence in L_2 error with respect to h/L for coarse particle resolution, i.e. for large h/L . However, for small h/L , the FVPM results are only second-order convergent. The transition between second-

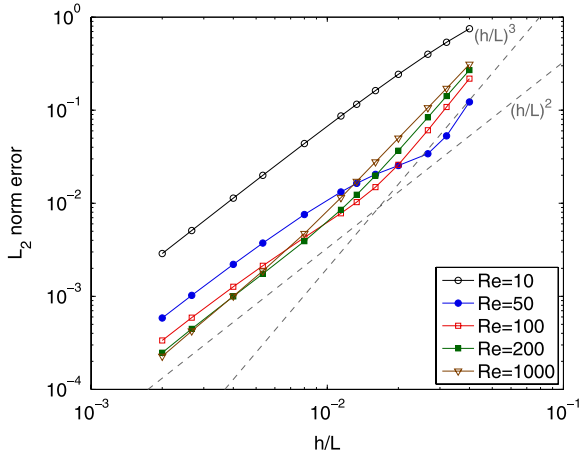


Fig. 8. Variation of L_2 norm acceleration error with particle size in FVPM simulations of Taylor–Green flow at various Reynolds numbers, with interparticle area based on exact integration and $h/\Delta x = 0.8$.

and third-order convergence is approximately independent of the ratio of particle size to spacing, $h/\Delta x$, as can be observed in Fig. 6. However, it is sensitive to Reynolds number, as shown in Fig. 8. The transition occurs at higher h/L as the Reynolds number is reduced. For $Re = 1000$, the observed convergence is third-order for $h/L \gtrsim 10^{-2}$ and second-order for finer discretisations. For $Re = 10$, on the other hand, the results display second-order convergence in the entire range of h/L tested. This behaviour suggests that the lower order of convergence is related to the calculation of viscous stress, which is based on SPH approximation of velocity gradients.

5.2. Shock tube

A one-dimensional shock tube flow of calorically perfect gas with $\gamma = 1.4$ has been simulated with FVPM. The initial pressure ratio p_4/p_1 is 4 and initial temperature is uniform. The shock tube is modelled in two dimensions with 10 particles across the width of the shock tube and 100 along the length. Although the analytical problem is strictly one-dimensional, two-dimensional discretisation allows observation of the numerical effects of walls and non-uniform particle distributions. Simulations were conducted both for regular and randomised initial particle distributions. The AUSM⁺-up scheme [20] is used along with the limiter of Barth and Jespersen [18]. To generate the randomised initial particle field, particles were initially placed on a regular cartesian grid of pitch Δx , then shifted by a random displacement, distributed with uniform probability between $-0.2\Delta x$ and $0.2\Delta x$ in both x and y directions. To ensure that the initial pressure discontinuity is planar and well-defined, regular distribution is maintained near the initial discontinuity in all cases.

The Courant number was 1.0 for all simulations, except those where quadrature was used with $N \times N = 2 \times 2$ quadrature points. Those simulations were found more susceptible to instability, and a lower Courant number of 0.5 was used in most cases. The case of $N = 2$ with pairwise β_{ij} correction and uniform particle distribution was a further exception, for which $C = 0.25$. In all cases, circular particles with $h/\Delta x = 0.7$ were used, with Lagrangian motion.

Results of the shock-tube simulations are shown in Fig. 9. In all conditions tested, results with exact β_{ij} (top-hat weight function (11)) are free of overshoot near the shock and show a sharply resolved contact surface. They recover nearly one-dimensional solutions, even when particle positions are randomised in transverse and longitudinal directions.

Results for regular initial particle distributions are shown in Fig. 9. Quadrature with 6×6 integration points (quadratic weight

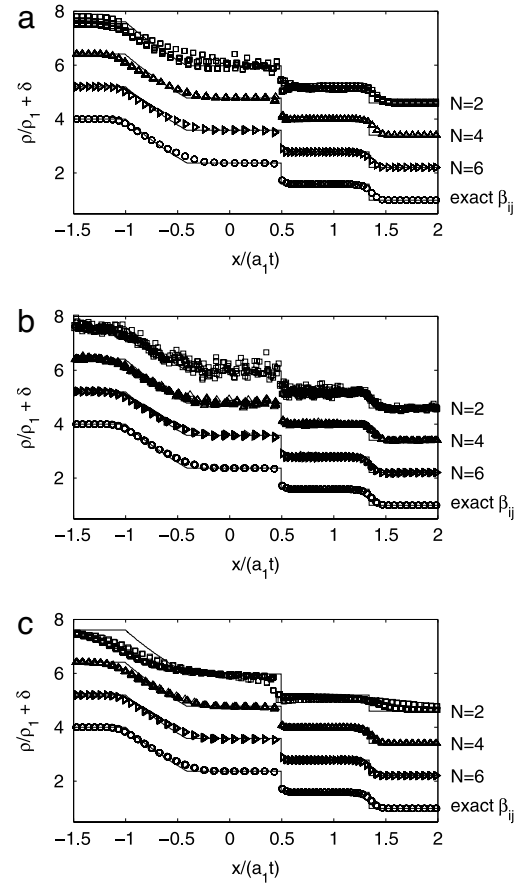


Fig. 9. FVPM simulations of nominally 1D flow in a 2D shock tube, with β_{ij} determined approximately by quadrature with $N \times N$ integration points, or exactly. Symbols denote numerical results and solid lines denote the analytical solution. Data are normalised with respect to initial density ρ_1 and sound speed a_1 in the low-pressure region, and shifted on the density axis by an arbitrary offset δ for clarity. (a) Regular initial particle distribution with no β_{ij} correction; (b) randomised initial particle distribution with no β_{ij} correction; (c) randomised initial particle distribution with pairwise β_{ij} correction in quadrature cases. The solutions for exact β_{ij} are the same in cases (b) and (c).

function (11)) yields results almost identical to those obtained with exact β_{ij} . With fewer integration points, quadrature gives poor results. Transverse density variations of over 10% are evident for $N = 2$, due to the effect of the side walls of the shock tube. The case of quadrature with $N = 2$ is included because its CPU time cost is only about 20% greater than that of exact β_{ij} . However, its accuracy is very poor.

For randomised initial particle distributions (Fig. 9(b)), quadrature-based computations display spurious transverse variations of density ranging in magnitude from 3.8% ($N = 6$) to 43% ($N = 2$). Pairwise correction of β_{ij} [16] (Fig. 9(c)), intended to ensure conservation, restores the accuracy of results for $N = 6$ to a similar level as for regular initial particle distributions, and in most cases gives a uniform field in the transverse direction. However, the least accurate quadrature, $N = 2$, results in severe smoothing of all waves, with large transverse variation in density.

Simulations were also carried out with square particles aligned obliquely to the flow direction. The thickness of the computed shock was unaffected by the orientation of the particles.

5.3. Supercritical aerofoil

Inviscid transonic flow around a NACA 0012 aerofoil was simulated for a freestream Mach number of 0.8 and an angle of attack of 1.25° . The aerofoil is in a rectangular domain of streamwise

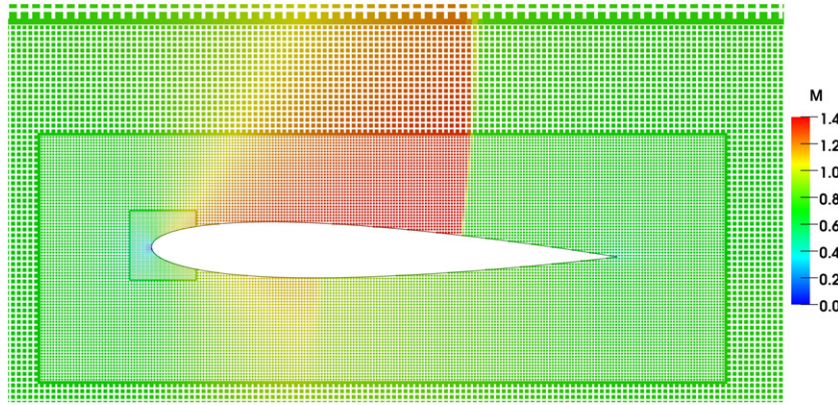


Fig. 10. Particles coloured by Mach number in FVPM computation of flow around a NACA 0012 aerofoil with $\Delta x_0/c = 1/320$, with exact determination of interparticle area. The size of dots is proportional to (but less than) the size of finite volume particles. Apparently solid lines at interfaces between regions of different particle size represent particles of intermediate size which are placed at the spacing of the smaller particles.

length $10c$ and width $10c$, where c is the nominal chord length, and the leading edge is located $2c$ from the upstream boundary. Following Vassberg and Jameson [27], the polynomial that defines the aerofoil surface is extended to $x/c \approx 1.0089$ to form a sharp trailing edge. The aerofoil surface and the lateral boundaries are free-slip walls. The upstream and downstream boundaries are represented by the non-reflecting boundary conditions of Giles [28], as adapted for SPH by Lastiwka et al. [29]. The AUSM⁺-up scheme [20] is used along with the limiter of Barth and Jespersen [18]. Eulerian particles with square supports are used, arranged in a locally refined piecewise uniform Cartesian distribution, as shown in Fig. 10. Particle spacing differs by a factor of 2 between contiguous zones. A single row of particles of intermediate size is placed at interfaces between zones of different particle size to reduce the number of overlapping neighbours in these areas. The smallest particle spacing is $\Delta x_0 = c/320$ and occurs in a region around the leading edge. Additional particles are distributed on the aerofoil surface at a spacing of approximately Δx_0 . The ratio $h/\Delta x$ is 0.45. Interparticle area was evaluated using both the new exact integration scheme and 6×6 quadrature with pairwise correction, which restores exact conservation [16]. Pairwise correction comes into effect at interfaces between zones of different particle spacing and at the aerofoil surface, where the particle distribution is not regular.

The particle distribution is illustrated in Fig. 10, along with the resulting Mach number field computed with exact determination of β_{ij} . Pressure distributions on the aerofoil surface are shown in Fig. 11 for both exact and approximate β_{ij} , based on the weight functions (11) and (10) respectively. The main difference between the results is in the position of the shock on the lower surface. For this feature, exact β_{ij} gives closer agreement with the high-resolution finite volume solution of Vassberg and Jameson [27]. Quadrature also results in local oscillations in the pressure distribution, perhaps because the solution is highly sensitive to area errors at speeds near Mach number 1. On the other hand, exact interparticle area β_{ij} yields a smooth pressure distribution.

5.4. Rayleigh–Taylor instability

The above numerical test cases were chosen to demonstrate the performance of the new exact formulation for β_{ij} , but do not benefit from the meshless approach. In the final test problem, the new FVPM formulation is used to model Rayleigh–Taylor instability in a multiphase flow, and is compared to a classical mesh-based method. In principle, the Lagrangian particle method may reduce numerical diffusion at the phase interface. This problem has previously been used as a benchmark for SPH by several authors [30–33]. The initial conditions are shown in Fig. 12. A

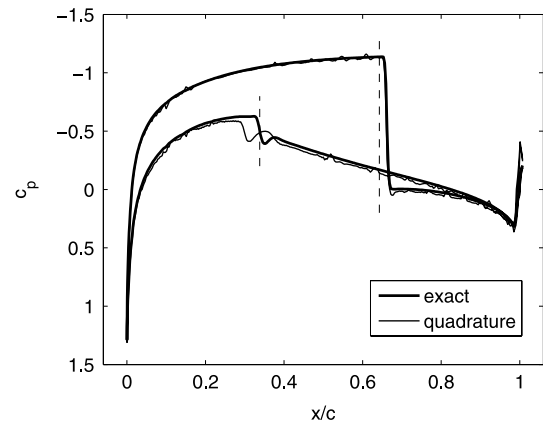


Fig. 11. Pressure distribution on a NACA 0012 aerofoil computed with particle interaction vectors determined exactly and by quadrature with pairwise correction. Vertical dashed lines indicate shock positions computed by Vassberg and Jameson [27].

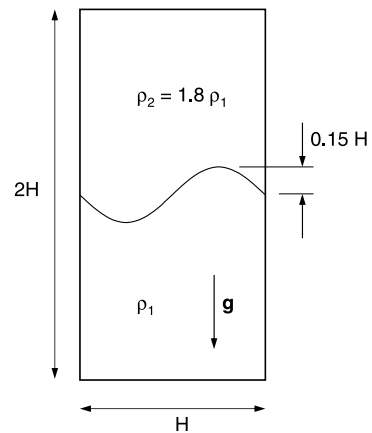


Fig. 12. Geometry and initial conditions for the Rayleigh–Taylor instability problem.

heavy fluid rests above a lighter fluid in a box with no-slip walls measuring $H \times 2H$. The interface between the fluids is at $y = -(0.15)\sin(2\pi x/H)$ and the density ratio ρ_2/ρ_1 is 1.8. Acceleration due to gravity g acts in the $-y$ direction. The two fluids have equal kinematic viscosity and the Reynolds number is 420, based on H and characteristic velocity \sqrt{gH} . Both fluids are governed by the modified Tait equation of state and their speed of sound a_0 is selected for a characteristic Mach number \sqrt{gH}/a_0 of 0.0626.

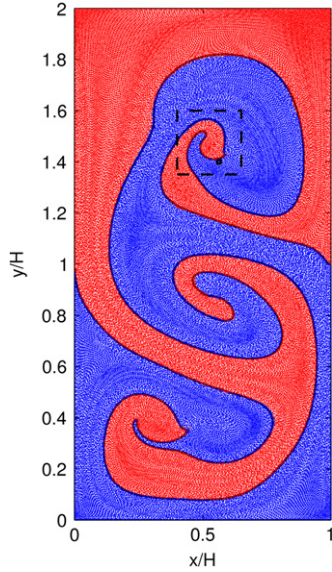


Fig. 13. Particle distribution at $t\sqrt{g/H} = 5$ in the Rayleigh–Taylor instability with $\Delta x/H = 1/300$ (blue and red dots denote particles of light and heavy fluid, respectively) and the interface computed using VoF with mesh cell size $\Delta x/H = 1/600$ (black curve). The square indicates the field of view in plots below, and the black circle shows the size of the particle support. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

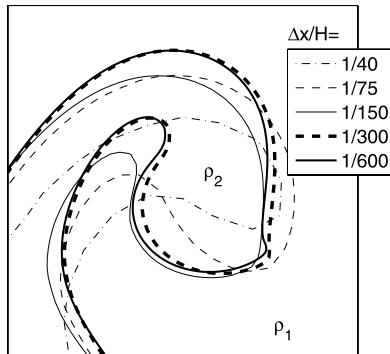


Fig. 14. Interface at $t\sqrt{g/H} = 5$ in the Rayleigh–Taylor instability problem, computed using VoF with various mesh cell size Δx .

Circular Lagrangian particles are distributed at initial spacings in the range $1/80 \leq \Delta x/H \leq 1/300$, with $h/\Delta x = 0.6$, giving 18 neighbours per particle. Interparticle area is calculated exactly, based on the weight function (11). No surface tension or physical diffusion is modelled and no special numerical treatment is used at the phase interface.

For comparison, the flow has been simulated using the mesh-based finite volume solver *interFoam*, a part of the *OpenFOAM*

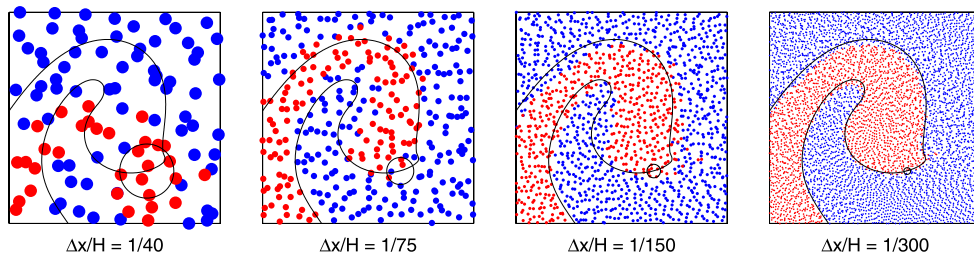


Fig. 15. Particle distribution at $t\sqrt{g/H} = 5$ in the Rayleigh–Taylor instability, computed with the FVPM with various initial particle spacings Δx . Also shown in each plot is the interface contour computed using VoF on the finest mesh, 600×1200 cells. Circles indicate the size of the particle support in each case. The field of view in these plots is marked by a dashed line in Fig. 13.

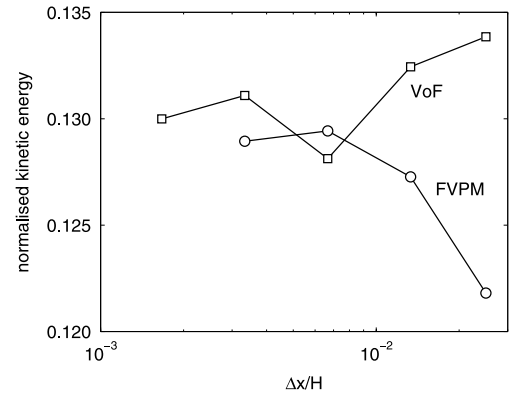


Fig. 16. Total kinetic energy (normalised to $\rho_1 g H^2$) at $t\sqrt{g/H} = 5$ in the Rayleigh–Taylor instability problem, computed using FVPM and VoF with various initial particle spacing and mesh cell size Δx .

open-source software suite [34–36]. This solver employs a volume-of-fluid (VoF) method for multiphase flow with the PISO algorithm for incompressible flow. The VoF solver was used without interface compression, which was sometimes found to cause unphysical effects, and did not affect the rate of convergence, according to any of the measures used here. Finite volume solutions were obtained on Cartesian meshes of cell size equal to the particle spacing in the various FVPM tests, as well as a finer 600×1200 mesh, maintaining a Courant number of approximately 0.5 based on instantaneous maximum material velocity. As the VoF-computed interface is smeared over several cells, it is defined for visualisation purposes as the density contour $\rho(\mathbf{x}) = (\rho_1 + \rho_2)/2$.

A sample result from the finest FVPM and VoF discretisations at dimensionless time $t\sqrt{g/H} = 5$ is shown in Fig. 13. There is close agreement for the computed interface structure. To evaluate convergence and agreement in more detail, we consider the region highlighted in Fig. 13, which is the most sensitive to computational resolution. The interface computed by VoF for various resolutions is shown in Fig. 14. In terms of the qualitative structure of the interface, the solution is converged. On coarser meshes, the interface geometry is significantly affected by mesh cell size. Mesh sensitivity is particularly evident in resolution of the penetration of a jet of light fluid into the heavy fluid.

Results of FVPM for the same region are visualised in Fig. 15. The VoF interface computed on the finest mesh is also shown. At this scale, good agreement is evident between the finest FVPM and VoF computations. The major features of the interface in this region are well predicted by FVPM for all discretisations with $\Delta x/H \leq 1/75$. There are small variations in the position of the roll-up structure, with deviations from the fine-mesh VoF result on the order of one particle radius.

Dependence of total kinetic energy on spatial resolution is illustrated in Fig. 16. At the finest discretisations, FVPM and VoF agree within 2%. Exact agreement of converged results is not to be expected, since a weakly compressible fluid model (FVPM) is being

Table 1

Normalised computational time for FVPM and VoF solutions of the Rayleigh–Taylor instability problem.

$\Delta x/H$	VoF	FVPM
1/40	1.00	89.5
1/75	9.81	620
1/150	145	5060
1/300	2620	43 200
1/600	12 200	–

compared with an incompressible fluid model (VoF). At coarser resolution, FVPM displays less sensitivity than VoF to Δx , with approximately half the range of kinetic energy for all cases but $\Delta x/H = 1/40$.

Computational time requirements are shown in Table 1. FVPM is more than an order of magnitude more costly than VoF for equal particle spacing and mesh cell size. However, computational cost should be compared for equivalent quality of results, and not simply on number of discretisation points. The results above suggest that coarser discretisation may be sufficient in FVPM when the objective is to resolve interface kinematics. For example, the interface structure computed by mesh-converged VoF at $\Delta x/H = 1/300$ is qualitatively the same as that computed by FVPM on a coarser discretisation of $\Delta x/H = 1/150$ (less than twice as slow as VoF at $\Delta x/H = 1/300$), or arguably for $\Delta x/H = 1/75$ (more than 4 times faster than VoF at $\Delta x/H = 1/300$). These results suggest that the meshless method may be competitive in terms of the computational effort required to resolve interface flow features.

6. Conclusions

The finite volume particle method (FVPM) has the potential to combine the rigorous properties of the classical finite volume method with the advantages of a meshless framework similar to that of SPH. To date, its computational cost has been prohibitive. However, with a particle weight function which is uniform over the particle support, the interparticle area in FVPM can be evaluated exactly and much more quickly than by the established method of numerical quadrature. In tests on 2D flow problems, the method is faster than FVPM with quadrature by a factor of 6.4–8.2. It also reduces numerical error, displaying improved convergence, better resolution of shocks, better tolerance of irregular particle distribution, and smoother pressure distribution in a transonic flow. In contrast with standard FVPM and SPH, error appears to be independent of particle overlap (or number of neighbours); it is only necessary that there are no gaps between particles. Locally refined spatial resolution has been exploited without any special treatment (this is routine in mesh-based techniques, but rare in meshless methods). In a multiphase flow, the new scheme is competitive with a mesh-based volume-of-fluid finite volume method, in terms of computational time requirements and resolution of the interface geometry.

Acknowledgements

The research leading to these results has been supported by funding from the European Community's Seventh Framework Programme (FP7/2007–2013) under grant agreement 225967 “NextMuSE”, the Irish Research Council for Science, Engineering and Technology, and the European Regional Development Fund (ERDF) project “NTIS – New Technologies for the Information Society”, European Centre of Excellence, CZ.1.05/1.1.00/02.0090.

References

- [1] P. Koumoutsakos, Multiscale flow simulations using particles, *Annu. Rev. Fluid Mech.* 37 (2005) 457–487.

- [2] J.J. Monaghan, Smoothed particle hydrodynamics and its diverse applications, *Annu. Rev. Fluid Mech.* 44 (1) (2012) 323–346. <http://dx.doi.org/10.1146/annurev-fluid-120710-101220>.
- [3] D. Violeau, *Fluid Mechanics and the SPH Method*, Oxford University Press, Oxford, UK, 2012. URL http://ukcatalogue.oup.com/product/9780199655526.do#Uclzdpdx7_xE.
- [4] D. Hietel, K. Steiner, J. Struckmeier, A finite volume particle method for compressible flows, *Math. Models Methods Appl. Sci.* 10 (2000) 1363–1382.
- [5] T. Ismagilov, Smooth volume integral conservation law and method for problems in Lagrangian coordinates, *Comput. Math. Math. Phys.* 46 (2006) 453–464.
- [6] T.Z. Ismagilov, On a smooth volume approach, integral conservation law, and upwind scheme with monotonic reconstruction, *Sib. J. Numer. Math.* 9 (4) (2006) 345–352.
- [7] M. Junk, Do finite volume methods need a mesh? in: M. Griebel (Ed.), *Meshfree Methods for Partial Differential Equations*, Springer, 2003, pp. 223–238.
- [8] M. Serrano, P. Español, I. Zúñiga, Voronoi fluid particle model for Euler equations, *J. Stat. Phys.* 121 (1–2) (2005) 133–147. <http://dx.doi.org/10.1007/s10955-005-8414-y>.
- [9] V. Springel, E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh, *Mon. Not. R. Astron. Soc.* 401 (2) (2010) 791–851. <http://dx.doi.org/10.1111/j.1365-2966.2009.15715.x>.
- [10] R. Nestor, M. Basa, M. Lastiwka, N. Quinlan, Extension of the finite volume particle method to viscous flow, *J. Comput. Phys.* 228 (2009) 1733–1749.
- [11] N.J. Quinlan, R.M. Nestor, Fast exact evaluation of particle interaction vectors in the finite volume particle method, in: M. Griebel, M.A. Schweitzer (Eds.), *Meshfree Methods for Partial Differential Equations V*, in: *Lecture Notes in Computational Science and Engineering*, vol. 79, Springer, Berlin, Heidelberg, 2011, pp. 219–234. http://dx.doi.org/10.1007/978-3-642-16229-9_14.
- [12] R. Keck, D. Hietel, A projection technique for incompressible flow in the meshless finite volume particle method, *Adv. Comput. Math.* 23 (2005) 143–169.
- [13] D. Teleaga, J. Struckmeier, A finite-volume particle method for conservation laws on moving domains, *Internat. J. Numer. Methods Fluids* 58 (2008) 945–967.
- [14] D. Teleaga, A finite volume particle method for conservation laws, Ph.D. Thesis, University of Kaiserslautern, 2005.
- [15] M. Junk, J. Struckmeier, Consistency analysis of meshfree methods for conservation laws, *Mitt. Ges. Angew. Math. Mech.* 24 (2001) 99–126.
- [16] D. Hietel, R. Keck, Consistency by coefficient correction in the finite volume particle method, in: M. Griebel (Ed.), *Meshfree Methods for Partial Differential Equations*, in: *Lecture Notes in Computational Science and Engineering*, Springer, Berlin, 2003, pp. 211–221.
- [17] B. van Leer, Towards the ultimate conservative difference scheme. V—a second-order sequel to Godunov's method, *J. Comput. Phys.* 32 (1979) 101–136.
- [18] T. Barth, D. Jespersen, The design and application of upwind schemes on unstructured meshes, in: *AIAA 27th Aerospace Sciences Meeting*, AIAA, Reno, 1989.
- [19] M.-S. Liou, A sequel to AUSM: AUSM⁺, *J. Comput. Phys.* 129 (1996) 364–382.
- [20] M.-S. Liou, A sequel to AUSM, part II: AUSM⁺-up for all speeds, *J. Comput. Phys.* 214 (2006) 137–170.
- [21] J. Bonet, T.-S.L. Lok, Variational and momentum preservation aspects of smooth particle hydrodynamic formulations, *Comput. Methods Appl. Mech. Engrg.* 180 (1999) 97–115.
- [22] A. Katz, A. Jameson, A comparison of various meshless schemes within a unified algorithm, in: *47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, American Institute of Aeronautics and Astronautics, 2009. <http://dx.doi.org/10.2514/6.2009-596>.
- [23] C. Schick, Adaptivity for particle methods in fluid dynamics, Master's Thesis, University of Kaiserslautern, 2000.
- [24] G.I. Taylor, A.E. Green, Mechanism of the production of small eddies from large ones, *Proc. R. Soc. Lond. Ser. A Math. Phys. Sci.* 158 (895) (1937) 499–521. <http://dx.doi.org/10.1098/rspa.1937.0036>. [arXiv:http://rspa.royalsocietypublishing.org/content/158/895/499.full.pdf+html](http://rspa.royalsocietypublishing.org/content/158/895/499.full.pdf+html). URL <http://rspa.royalsocietypublishing.org/content/158/895/499.short>.
- [25] R.H. Cole, *Underwater Explosions*, Princeton University Press, Princeton, 1948.
- [26] N. Quinlan, M. Lastiwka, M. Basa, Truncation error in mesh-free particle methods, *Int. J. Numer. Methods Eng.* 66 (2006) 2064–2085.
- [27] J.C. Vassberg, A. Jameson, In pursuit of grid convergence, part 1: two-dimensional Euler solutions, in: *27th AIAA Applied Aerodynamics Conference*, no. AIAA-2009-4114, San Antonio, Texas, 2009.
- [28] M.B. Giles, Non-reflecting boundary conditions for Euler equation calculations, *AIAA J.* 28 (12) (1992) 2050–2058.
- [29] M. Lastiwka, M. Basa, N. Quinlan, Permeable and non-reflecting boundary conditions in SPH, *Internat. J. Numer. Methods Fluids* 61 (7) (2009) 709–724.
- [30] S. Cummins, M. Rudman, An SPH projection method, *J. Comput. Phys.* 152 (1999) 584–607.
- [31] X.Y. Hu, N.A. Adams, An incompressible multi-phase SPH method, *J. Comput. Phys.* 227 (1) (2007) 264–278. <http://dx.doi.org/10.1016/j.jcp.2007.07.013>.
- [32] X.Y. Hu, N.A. Adams, A constant-density approach for incompressible multi-phase SPH, *J. Comput. Phys.* 228 (6) (2009) 2082–2091. <http://dx.doi.org/10.1016/j.jcp.2008.11.027>.

- [33] N. Grenier, M. Antuono, A. Colagrossi, D. Le Touzé, B. Alessandrini, An hamiltonian interface SPH formulation for multi-fluid and free surface flows, *J. Comput. Phys.* 228 (22) (2009) 8380–8393.
<http://dx.doi.org/10.1016/j.jcp.2009.08.009>.
- [34] OpenFOAM Foundation, OpenFOAM User Guide, <http://www.openfoam.org/>.
- [35] H. Rusche, Computational fluid dynamics of dispersed two-phase flows at high phase fractions, Ph.D. Thesis, Imperial College London, 2003.
- [36] S.S. Deshpande, L. Anumolu, M.F. Trujillo, Evaluating the performance of the two-phase flow solver interFoam, *Comput. Sci. Discov.* 5 (1) (2012) 014016+.
<http://dx.doi.org/10.1088/1749-4699/5/1/014016>.