View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Loughborough University Institutional Repository

A Robust Two-Node, 13 Moment Quadrature Method of Moments for Dilute Particle Flows Including Wall Bouncing

Dan Sun, Andrew Garmory[1] & Gary J. Page

Department of Aeronautical and Automotive Engineering, Loughborough University, Loughborough, LE11 3TU, UK

Abstract

For flows where the particle number density is low and the Stokes number is relatively high, as found when sand or ice is ingested into aircraft gas turbine engines, streams of particles can cross each other's path or bounce from a solid surface without being influenced by inter-particle collisions. The aim of this work is to develop an Eulerian method to simulate these types of flow. To this end, a two-node quadrature-based moment method using 13 moments is proposed. In the proposed algorithm thirteen moments of particle velocity, including cross-moments of second order, are used to determine the weights and abscissas of the two nodes and to set up the association between the velocity components in each node. Previous Quadrature Method of Moments (QMOM) algorithms either use more than two nodes, leading to increased computational expense, or are shown here to give incorrect results under some circumstances. This method gives the computational efficiency advantages of only needing two particle phase velocity fields whilst ensuring that a correct combination of weights and abscissas are returned for any arbitrary combination of particle trajectories without the need for any further assumptions. Particle crossing and wall bouncing with arbitrary combinations of angles are demonstrated using the method in a two-dimensional scheme. The ability of the scheme to include the presence of drag from a carrier phase is also demonstrated, as is bouncing off surfaces with inelastic collisions. The method is also applied to the Taylor-Green vortex flow test case and is found to give results superior to the existing two-node QMOM method and in good agreement with results from Lagrangian modelling of this case.

Key words: Multiphase flow; Eulerian method; QMOM; particle bouncing.

1. Introduction

The physical phenomenon of particle wall-bouncing is important for dilute, dispersed multiphase flows in a variety of applications for which numerical simulations are required. For example, when turbomachinery operates in sandy or dusty environments it is important to be able to predict accurately the rate at which particles impact on components in order to predict the rate of erosion or deposition [1] [2]. Another example is the build-up of ice both inside and outside of aircraft engines which is a matter of concern to the aviation industry. Predictions of ice accretion in such cases rely

[1] Email: a.garmory@lboro.ac.uk, Tel: +44 1509 227231.

on accurate calculations of the mass flux of ice crystals hitting a particular surface [3]. These ice crystals, unlike super-cooled liquid droplets [4], will bounce and travel further into the engine. For these cases it may be necessary to predict the motion of particles through the compressor including several rotating and stationary frames of reference for the rotors and stators, hence it is important that the method is numerically efficient when particles transfer across the interfaces. The prediction of bouncing particles is also important in processes involving powder technology, see [5] [6] [7] for examples of this.

The confinement by a solid surface changes the particle motion significantly as a result of particle-wall collisions. This is particularly true for high inertia particles which maintain their direction of motion for a long time after bouncing off a wall instead of following the carrier fluid flow path, and this can result in a subsequent collision with another wall [5]. The importance of particle inertia can be evaluated by the Stokes number which is defined as the ratio of the characteristic time of a particle to the characteristic time of the flow [8]. If the Stokes number is high, as it can be for sand particles, then wall bouncing and trajectory crossing will be significant [9, 10]. The examples given above require accurate and efficient simulation on complex geometries. It is also true in many engineering applications that the total mass flux of particles is of more practical interest than the behaviour of individual particles, making the use of Eulerian methods attractive. Hence, it is the aim of this work to provide an improved method of modelling flows involving particle bouncing and crossing in an Eulerian CFD framework.

To simulate wall-bouncing flows, the Lagrangian method is generally used to trace the flow path of particles in a dilute gas particle flow. The effects of wall roughness and inter-particle collisions can be included in a straightforward manner in this way. For example, Sommerfeld et al. [11] [12] calculated the particle bouncing trajectories in horizontal channels with a Lagrangian method. In comparison with experimental results, the Lagrangian calculation method accurately modelled the transport phenomena of wall collisions and inter-particle collisions. However, Lagrangian based methods have drawbacks compared to Eulerian methods which make them less suitable in practical CFD calculations. For reasonably high particle loadings the computational cost of tracking enough particles to model accurately the true population can be very high. This is exacerbated in unsteady or two-way coupled flows when a ‘frozen’ continuous flow-field cannot be employed and the tracking must take place at each timestep rather than as a single post-processing step. An Eulerian dispersed phase calculation can be carried out on the same CFD grid as the continuous phase which means it can take advantage of existing flow solvers, efficient parallelisation schemes, interfaces for rotating reference frames as well as pre and post-processing. Finally, the Eulerian method gives directly quantities of engineering interest, such as mass fluxes, which in a Lagrangian method will be dependent on the number of particles and the sampling time. For a thorough discussion of the merits of Lagrangian and Eulerian multiphase methods see Fox [13].

With Eulerian methods, if a single velocity field is used to describe the motion of a dispersed phase then this will be incapable of correctly modelling conditions where dilute streams of particles pass through one another, as is the case when bouncing off a surface. By definition, for these cases there will be particles within the same computational cell with different velocities and so a single velocity field is not capable of correctly representing such a velocity distribution. The Quadrature Method of Moments (QMOM) allows for the simulation of a particle velocity distribution within an Eulerian framework. Fox proposed a third order QMOM [14], which successfully predicted particle-trajectory crossing. With this third-order QMOM, further testing was carried out for Riemann type shock problems, impinging jets, a vertical channel flow and crossing trajectories of particles [15]. Besides dilute particle flow, this high-order quadrature-based moment method was also applied to moderately dilute [16] and dense fluid-particle flows [17] [18]. The model was also extended for poly-disperse sprays [19] and the n3-node QMOM derived to achieve a higher accuracy [20].

While the third-order QMOM of Fox [14] is capable of predicting crossing trajectories and bouncing, the set of 20 moments and 8 weighted velocity fields has a relatively high computational overhead. For dilute gas-particle flows with high Reynolds numbers in complex geometries, for example of the type found in turbomachinery, a computationally less intense method would be desirable. Desjardins et al [21] proposed a two-node QMOM method for dilute fluid-particle flows and predicted crossing particle jets and rebounding off the wall with this model. This approach reduces the number of nodes from $2d$ for third-order QMOM to 2, where d is the number of dimensions. Meanwhile, the number of moments is reduced from 20 for third-order QMOM system to 8 for a three-dimensional scheme with an associated reduction in the computational costs. Therefore, the two-node quadrature moment model is attractive as a computationally efficient model for particle bouncing flows. However, as we show in this paper, the method in [21] does not give the correct dilute particle flow behaviour in some cases. Jets of particles should, in the dilute limit, pass through each other without influencing the other jet. However, for some combinations of jet velocity the two-node eight-moment method does not reproduce this - instead the jets erroneously appear to collide with each other. In this paper we demonstrate this problem, and its underlying cause, before presenting an improved two-node 13-moment method which overcomes this limitation.

In the next section we first describe the quadrature-based method of moments followed by the existing two-node, eight-moment method and demonstrate simple test cases for which it does not give the correct behaviour. We then introduce the new two-node 13-moment formulation and demonstrate that it gives the correct behaviour where the eight-moment method does not. We then further demonstrate results using the new method for cases where the particles undergo bouncing and cases with drag due to a carrier phase, including the Taylor-Green vortex test case.

2. Quadrature-based method of moments for particles

The method of moments (MOM) was originally used to determine the evolution of the lower-order moments of the particle size distribution function of an aerosol population [22] [23]. Because of the unknown distribution function of particle size McGraw [24] introduced the Gaussian quadrature method to MOM by approximating the moments of an aerosol distribution as

$$M^s = \sum_{\alpha=1}^{\beta} w_\alpha r_\alpha^s \tag{1}$$

The principle of quadrature based method of moments (QMOM) lies in how to determine the abscissas w_α and weights r_α of the β nodes. This can be done using the product difference (PD) algorithm [25] as applied by McGraw [24] in QMOM. The QMOM method has been extended to moments of the velocity distribution [21] [14]. The velocity distribution is represented by a finite number of weighted velocity fields or nodes. Hence *s*-order moments of the velocity distribution function *f* can be denoted by summing over the nodes.

$$M_{i\ldots k}^s = \int U_i \cdots U_k f d\mathbf{v} \tag{2}$$

The weights and abscissas (or velocity field) of each node (the weighting and velocity components of node *α* can be written as [n_α,$U_{\alpha i}$,$U_{\alpha j}$,$U_{\alpha k}$]) are determined by forcing them to agree with the set of moments employed. In practice QMOM CFD algorithms work by updating a set of moments using the fluxes calculated from the velocity fields in the nodes. The updated moments must then be used to find a new set of velocity fields and their weights. Hence, a two-way method of converting between weighted velocity fields and moments is central to QMOM. In the next section the existing two-node, eight moment QMOM [21] is presented in order to better explain the development of the method in the current paper.

2.1. Existing Two-Node Eight Moment QMOM

Here we consider only the mono-disperse case where all particles are assumed to be uniformly sized spherical particles with no growth or nucleation processes present. The transport equations are established by the kinetic theory of gases [26],

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \frac{\partial f}{\partial \mathrm{x}} + \frac{\partial f}{\partial \mathrm{u}}\left(f \frac{\mathbf{F}}{m_p}\right) = \mathbb{C} \tag{3}$$

where *f* is the velocity distribution function, and **u** is the particle velocity vector. $\mathbb{C}$ is the collision term, representing the property change due to inter-particle collisions. If the particle flow is far from equilibrium, when the Knudsen number is quite large, the equation is controlled by the terms on the left-hand side (LHS) in eq. (3) and the collision term is neglected for a dilute fluid–particle flow. **F** in the last term on the LHS is the external force, including gravity and the drag force, on the particles. In this study, the drag force and gravity are expressed as

$$\mathbf{F}(\mathbf{U_f},\mathbf{u}) = \frac{m_p}{\tau_p}(\mathbf{U_f} - \mathbf{u}) + m_p\mathbf{g} \tag{4}$$

where m_p is the mass of a single particle, $\mathbf{g}$ is the gravity vector and τ_p is the characteristic timescale for particle drag, expressed as

$$\frac{1}{\tau_p} = \frac{3\rho_f}{4d_p\rho_p}C_D|\mathbf{U_f} - \mathbf{u}| \tag{5}$$

where ρ_f is the fluid density, ρ_p is the particle density, d_p is the particle diameter, and $\mathbf{U_f}$ is the carrier fluid velocity vector. The particle drag coefficient C_D can be modelled for $1 \leq Re_P \leq 1000$ as [27],

$$C_D = \frac{24}{Re_p}\left(1 + 0.15\left(Re_p\right)^{0.687}\right) \tag{6}$$

where $Re_P = \rho_f d_p|\mathbf{U_f} - \mathbf{u}|/\mu_f$ is the particle Reynolds number, and μ_f is the dynamic viscosity of the fluid phase.

The moments of the velocity distribution are defined in Eq (2). The transport equations for moments up to third order can be obtained from Eq (3), neglecting inter-particle collisions as,

$$\begin{aligned}
&\frac{\partial M^0}{\partial t} + \frac{\partial M_k^1}{\partial x_k} = 0\\
&\frac{\partial M_i^1}{\partial t} + \frac{\partial M_{ik}^2}{\partial x_k} = D_i\\
&\frac{\partial M_{ij}^2}{\partial t} + \frac{\partial M_{ijk}^3}{\partial x_k} = D_{ij}\\
&\frac{\partial M_{iii}^3}{\partial t} + \frac{\partial M_{iiik}^4}{\partial x_k} = D_{iii}
\end{aligned} \tag{7}$$

The notation $\partial/\partial x_k$ implies summation over the three spatial coordinates. The velocity distribution can be represented as the sum of a number of delta functions

$$f(\mathbf{u}) = \sum_{\alpha=1}^{\beta} n_\alpha \delta(\mathbf{u}-\mathbf{U}_\alpha) \tag{8}$$

where β is the number of nodes. n_α is the number density representing the weights of the nodes, $\alpha = 1, \cdots, \beta$. $\mathbf{U}_\alpha$ is the particle velocity vector of node α, representing the abscissas ($\mathbf{U}_\alpha = \left[U_{\alpha i}, U_{\alpha j}, U_{\alpha k}\right]^T$) in the QMOM system. Accordingly, using a two-node quadrature the moments can be expressed as

$$M^0 = n_1 + n_2$$

$$
\begin{aligned}
&M_i^1 = n_1 U_{1i} + n_2 U_{2i} \\
&M_{ij}^2 = n_1 U_{1i} U_{1j} + n_2 U_{2i} U_{2j} \\
&M_{ijk}^3 = n_1 U_{1i} U_{1j} U_{1k} + n_2 U_{2i} U_{2j} U_{2k} \\
&M_{ij\ldots k}^s = n_1 U_{1i} U_{1j} \cdots U_{1\mathrm{k}} + n_2 U_{2i} U_{2j} \cdots U_{2k}
\end{aligned} \tag{9}
$$

Note that any number of moments can be generated from the two nodes. The unclosed drag force term in the moment transport equations (7) can be expressed using a two quadrature node as,

$$
\begin{aligned}
&D_i = \frac{n_1}{m_p} F_{1i} + \frac{n_2}{m_p} F_{2i} \\
&D_{ij} = \frac{n_1}{m_p} U_{1i} F_{1j} + \frac{n_1}{m_p} U_{1j} F_{1i} + \frac{n_2}{m_p} U_{2i} F_{2j} + \frac{n_2}{m_p} U_{2j} F_{2i} \\
&D_{iii} = 3\frac{n_1}{m_p} (U_{1i})^2 F_{1i} + 3\frac{n_2}{m_p} (U_{2i})^2 F_{2i}
\end{aligned} \tag{10}
$$

The force components for the two fields are found from the vector **F** calculated using equation (4). Thus, the moment transport equations are set up by equation (7) and the transport of the moments can be calculated from the velocity nodes using Equations (9) and (10). The remaining problem is how to calculate the weights and abscissas from the set of moments, and this is considered for the eight moment case in the following section.

2.1.1. Calculation of velocity nodes using eight-moments

In this section the moment to velocity field conversion method employed in the eight moment QMOM model of Desjardins [21] is first presented in order to explain the motivation behind increasing the number of moments to thirteen in the present work. In addition to the zeroth moment, first and second moments in each direction are used, together with a single third order moment to represent the sum of the third moment in each direction.

$$\frac{\partial M^0}{\partial t} + \frac{\partial M_k^1}{\partial x_k} = 0 \tag{11}$$

$$\frac{\partial M_i^1}{\partial t} + \frac{\partial M_{ik}^2}{\partial x_k} = D_i \tag{12}$$

$$\frac{\partial M_{ii}^2}{\partial t} + \frac{\partial M_{iik}^3}{\partial x_k} = D_{ii} \tag{13}$$

$$\frac{\partial \mathrm{Q}}{\partial t} + \frac{\partial \mathrm{R}_k}{\partial x_k} = 3\sum_{i=1}^{d}(D_{iii}) \tag{14}$$

where the moments are defined as in equation (9), and D_i, D_{ii}, D_{iii} as in equation (10). In the third moment transport equation,

$$Q = \sum_{i=1}^{d} M_{iii}^3 \tag{15}$$

$$\mathrm{R}_k = n_1 \left(\sum_{j=1}^{d} \left(U_{1j}\right)^3\right) U_{1k} + n_2 \left(\sum_{j=1}^{d} \left(U_{2j}\right)^3\right) U_{2k} \tag{16}$$

In order to calculate the equivalent velocity fields a variance x is calculated by

$$\mathrm{x} = \frac{q_p}{2\sqrt{\left(q_p\right)^2 + 4\left(\sum_{i=1}^{d} a_{ii}\right)^2}} \tag{17}$$

where

$$q_p = \frac{1}{M^0}\left(Q - M^0 \sum_{i=1}^{d} \left(U_{pi}\right)^3 - 3M^0 \sum_{i=1}^{d} a_{ii} U_{pi}\right) \tag{18}$$

$$a_{ii} = \frac{M_{ii}^2}{M^0} - U_{pi} U_{pi} \tag{19}$$

and

$$U_{pi} = M_i^1 / M^0 \tag{20}$$

The number density and velocity of each node are then found by

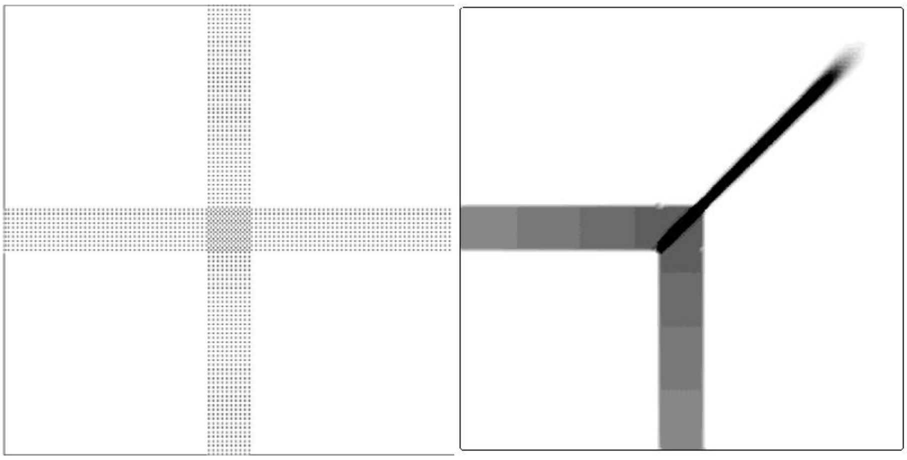
$$n_1 = \left(\frac{1}{2} + x\right) M^0 \,, \quad n_2 = \left(\frac{1}{2} - x\right) M^0 \tag{21}$$

$$U_{1i} = U_{\mathrm{p}i} - \sqrt{a_{ii}} \sqrt{\frac{1/2 - \mathrm{x}}{1/2 + x}} \tag{22}$$

$$U_{2i} = U_{\mathrm{p}i} + \sqrt{a_{ii}} \sqrt{\frac{1/2 + x}{1/2 - x}} \tag{23}$$

There are two deficiencies with this method. Firstly, for each velocity component the value in the second node will always be greater than the first assuming that positive roots are chosen or smaller with the negative root. This ordering of the velocity components is not physical; it is quite possible to find a situation where $U_{1j} > U_{2\mathrm{j}}$ and $U_{2i} > U_{1\mathrm{i}}$ and by being unable to generate such a combination the existing method will fail for some combinations of particle trajectory. For example, Figure 1 shows the case of two dilute streams of particles approaching at right angles to each other with conditions [n_1,U_{1i},U_{1j}] = [1,0,1] and [n_2,U_{2i},U_{2j}] = [1,1,0]. It is important to realise that the equations have been derived assuming dilute particle flow and hence the streams should cross each other unhindered as seen in Figure 1a. However, the existing QMOM method incorrectly shows the streams merging. The correct result requires $U_{1j} > U_{2\mathrm{j}}$ and $U_{2i} > U_{1\mathrm{i}}$ which cannot be obtained using the method above. The actual nodes obtained in the regions where the streams meet are [n_1,U_{1i},U_{1j}] = [1,0,0] and [n_2,U_{2i},U_{2j}] = [1,1,1] which, crucially, satisfy the original eight moments but do not give the desired crossing trajectories.

A second deficiency is in the calculation of the node weighting x from q_p which is in turn calculated in equation (18). The method should be capable of recovering the original two velocity fields used to create a two-node quadrature velocity distribution. However, in some cases it will incorrectly return a pair of nodes of equal weight. For example, if we start from the two node distribution [n_1,U_{1i},U_{1j}] = [1.5,0,1] and [n_2,U_{2i},U_{2j}] = [0.5,1,0], then the terms in equation (18) will be $M_0 = 2, Q = 2, U_{pi} = 0.25, U_{pj} = 0.75, a_{ii} = 0.1875$ and $a_{jj} = 0.1875$. This will lead to q_p becoming zero which would lead to the two node weights incorrectly being recovered as equal. The effect of this will be demonstrated in Section 4.1. Both of these deficiencies are essentially the result of the moment set containing insufficient information to reproduce a unique combination of nodes. In the next section we present a new method which uses 13 moments with two nodes to ensure that the correct, unique, combination of velocities is returned in each node for all cases.



(a) Expected result (b) two-node QMOM (6 moments for 2D)

Figure 1. 2D contours of particle number densities for crossing dilute jets with boundary conditions ([n_1,U_{1i},U_{1j}] = [1,0,1] and [n_2,U_{2i},U_{2j}] = [1,1,0] Results (a) show ideal results as from a Lagrangian method and (b) shows results obtained using QMOM method of [21] .

2.2. Proposed Two-Node, 13 Moment QMOM

Given any two weighted velocity fields it is possible to generate any number of moments from them. It is desired to use the fewest number of moments needed in order to unambiguously recover the velocity fields. In this work a set of 13 moments are used in a three dimensional system. The method of converting from a set of 13 moments to two, unique, weighted velocity fields is presented in this section. To obtain the weights and abscissas in our method, two steps are included. The first step is the calculation of abscissas and weights with the product-difference (PD) algorithm. The second step is to use additional cross moments to correctly associate the weights and the velocity components within the fields. The set of moments employed is

$$W^3 = \left(M^0, M_i^1, M_j^1, M_k^1, M_{ii}^2, M_{jj}^2, M_{kk}^2, M_{ij}^2, M_{jk}^2, M_{ik}^2, M_{iii}^3, M_{jjj}^3, M_{kkk}^3\right) \in \mathbb{R}^{13} \tag{24}$$

In comparison to the two-node, eight-moment system of [21] the extra moments here are the cross moments M_{ij}^2, M_{jk}^2 and M_{ik}^2 as well as all three of the third order moments M_{iii}^3, M_{jjj}^3 and M_{kkk}^3 instead of just their sum Q. We discuss below why these are needed.

The *i*th component of the mean velocity is defined as $U_{pi} = M_i^1/M^0$, and then from this each component of the velocity deviation of each node can be defined

$$\begin{aligned} c_{1i} &= U_{1i} - U_{pi} \\ c_{2i} &= U_{2i} - U_{pi} \end{aligned} \tag{25}$$

Accordingly, the moment of velocity deviation for an arbitrary moment can be defined in terms of the two nodes as

$$a_{i,j\cdots k} = \frac{1}{M^0}\left(n_1 c_{1i} c_{1j} \cdots c_{1\mathrm{k}} + n_2 c_{2i} c_{2j} \cdots c_{2\mathrm{k}}\right) \tag{26}$$

These deviation moments can be found from the velocity moments. We know, by definition, that

$$a_0 = 1,\, a_i = 0. \tag{27}$$

The second-order moment can be calculated by the velocity covariance matrix

$$[a_{ij}] = \begin{bmatrix} M_{ii}^2/M^0 - U_{pi}U_{pi} & M_{ij}^2/M^0 - U_{pi}U_{pj} & M_{ik}^2/M^0 - U_{pi}U_{pk} \\ M_{ij}^2/M^0 - U_{pi}U_{pj} & M_{jj}^2/M^0 - U_{pj}U_{pj} & M_{jk}^2/M^0 - U_{pj}U_{pk} \\ M_{ik}^2/M^0 - U_{pi}U_{pk} & M_{jk}^2/M^0 - U_{pj}U_{pk} & M_{kk}^2/M^0 - U_{pk}U_{pk} \end{bmatrix} \tag{28}$$

The third-order velocity variance is represented by the third central moment

$$[a_{iii}] = \begin{bmatrix} \frac{1}{M^0} M_{iii}^3 - \left(U_{pi}\right)^3 - 3a_{ii}U_{pi} \\ \frac{1}{M^0} M_{jjj}^3 - \left(U_{pj}\right)^3 - 3a_{jj}U_{pj} \\ \frac{1}{M^0} M_{kkk}^3 - \left(U_{pk}\right)^3 - 3a_{kk}U_{pk} \end{bmatrix}. \tag{29}$$

All of these calculations are based on the proposition that (i) $M^0 > 0$, and (ii) $\forall i = 1, \cdots, d,\ M^0 M_{ii}^2 \geq (M_i^1)^2$. Where (i) means the number density is non-negative which must be true physically and (ii) is easily proved according to the definition of moments.

In this method the two-node quadrature closure is used to calculate the weight and abscissa from each Cartesian direction. Only one pair of weights, n_1 and n_2, is needed which will be the same for all three directions. However, they are calculated from each direction to avoid situations, discussed below, where one or more of the directions cannot yield the weights. For example in the *i* direction the moments $(M^0, M_i^1, M_{ii}^2, M_{iii}^3)$ and the velocity deviation moments derived from these $(a_0, a_i, a_{ii}, a_{iii})$ are used. Introducing a variance x as the deviation of weights, the definition equations and moments can be summarized as

$$n_1 = \left(\frac{1}{2} + x\right) M^0 \ , \quad n_2 = \left(\frac{1}{2} - x\right) M^0 \tag{30}$$

$$\left(\frac{1}{2} + x\right)(c_{1i})^2 + \left(\frac{1}{2} - x\right)(c_{2i})^2 = a_{ii} \tag{31}$$

$$\left(\frac{1}{2} + x\right)(c_{1i})^3 + \left(\frac{1}{2} - x\right)(c_{2i})^3 = a_{iii} \tag{32}$$

From the expression in equations (30-31), in this method values for the variance, x can be calculated in each direction by

$$x = \frac{a_{iii}}{2\sqrt{(a_{iii})^2 + 4(a_{ii})^3}}, \quad a_{ii} \neq 0. \tag{33}$$

This is in contrast to Eqn (17) of the eight moment method where summations from all directions are used to find x. The reason for this is discussed below and demonstrated in Section 4.1. While the same weighting value, x, should be obtained from all three directions, there will be cases where one or more direction does not give a correct weighting. These cases are discussed below. The velocity of each node in the *i*-direction is found by,

$$\begin{aligned} U_{1i} &= U_{pi} - \sqrt{a_{ii}}\sqrt{\frac{1/2-\mathrm{x}}{1/2+\mathrm{x}}} \\ U_{2i} &= U_{pi} + \sqrt{a_{ii}}\sqrt{\frac{1/2+\mathrm{x}}{1/2-\mathrm{x}}} \end{aligned} \tag{34}$$

In a three dimensional system, the weights and abscissas in the other two dimensions can be calculated by moment sequences $(a_0, a_j, a_{jj}, a_{jjj})$ and $(a_0, a_k, a_{kk}, a_{kkk})$ as above. As the moments in all directions are derived from a two-node quadrature, the moments $(M^0, M_i^1, M_{ii}^2, M_{iii}^3)$ for all *i* will have the same relationship leading to the same value of *x* being generated in all three directions which can then be used in Eqn (30) to find the weights. There are two exceptions to this:

1. If $a_{ii} = 0$ in one or two directions. Under these circumstances the value of x is undefined in that direction as the velocity component in each field is equal and hence any combination of weights satisfies the moments in that direction. In this situation the variance x used to generate the weights in Eqn (30) is set equal to the non-zero values.

2. If $a_{ii} = 0$ for all directions then this means the two fields are identical and x is set to zero to give equal weights.

The eight moment approach of [21] uses summations of moments over all direction to find x as individual third order moments for each dimension are not available. As will be discussed further below, this can lead to numerical problems when $\sum_{i=1}^{d} a_{iii} = 0$ but $\mathrm{a}_{iii} \neq 0$ and node weights are incorrectly set to be equal.

Using Eqns. (30-34) node weights and the velocity components in all directions of the abscissas can be calculated. However, the association of the velocity components in each node is not fully defined at this point. This stems from $\sqrt{a_{ii}}$ having both positive and negative roots with insufficient information available to choose which root to take. If the assumption is made, for example, to use the positive roots of $\sqrt{a_{ii}}, \sqrt{a_{jj}}$ and $\sqrt{a_{kk}}$ then this restricts the velocity components in the nodes to a particular sequence. For example as $(U_{1i} - U_{2i})(U_{1j} - U_{2j}) \geq 0$ in 2-D cases. With different assumptions it would be possible to recreate stream crossing, but at the expense of obtaining erroneous results for other combinations of stream velocities. Hence, the next step is to associate the correct combination of velocity components with each weight. This will eliminate the failed trajectory crossing seen in Figure 1. To achieve this we use the cross moments M_{ij}^2 and the cross moment of velocity difference a_{ij}. The cross moment of velocity difference is defined by

$$\left(\frac{1}{2} + x\right) c_{1i} c_{1j} + \left(\frac{1}{2} - x\right) c_{2i} c_{2j} = a_{ij}, \ (i \neq j) \tag{35}$$

If ε_i represents the sign of the roots used in equation (34) then inserting expressions (30, 31) into equation (35) yields

$$\varepsilon_i \varepsilon_j = \frac{a_{ij}}{\sqrt{a_{ii}}\sqrt{a_{jj}}} \tag{36}$$

which means if the assumption is made that $\varepsilon_i = 1$, then the signs of ε_j and ε_k can be obtained by expression (36). A unique pair of weighted velocity fields can then be obtained from the set of moments. Therefore, the value of the remaining four velocities can be recalculated using the second order cross moment of velocity derivation.

$$\begin{aligned} U_{1j} &= U_{pj} - \left(\frac{a_{ij}}{\sqrt{a_{ii}}}\right)\sqrt{\frac{n_2}{n_1}} \\ U_{2j} &= U_{pj} + \left(\frac{a_{ij}}{\sqrt{a_{ii}}}\right)\sqrt{\frac{n_1}{n_2}} \\ U_{1k} &= U_{pk} - \left(\frac{a_{ik}}{\sqrt{a_{ii}}}\right)\sqrt{\frac{n_2}{n_1}} \\ U_{2k} &= U_{pk} + \left(\frac{a_{ik}}{\sqrt{a_{ii}}}\right)\sqrt{\frac{n_1}{n_2}} \end{aligned} \tag{37}$$

A unique pair of weighted velocity fields is thus obtained from the set of moments. The velocities of the two nodes are re-associated by the cross-moments of second-order.

2.3. Comparison with higher node closures

If we compare the method proposed here to the eight-node QMOM model by Fox [14] we see that in both methods the weighting and two values for velocity are generated for each direction using Eqns. (30) and (34). In our method cross-moments are used to associate velocity components in each node whereas in the eight-node third-order QMOM [14], all combinations of the three sets of two velocities are taken into account which leads to the eight nodes. The higher node closure will lead to a moment set that can contain a greater amount of information on the particle velocity distribution and, therefore, a method capable of handling more complex multiphase flow fields. However, our aim in this paper is to produce an Eulerian method for the simulation of dilute particle flows where trajectory crossing or bouncing takes place. It is desirable for this to be as computationally inexpensive as possible hence why a two-node closure is employed here. As well as its relative computational costs, the two-node method here has some numerical advantages over the eight-node method. For the eight-node QMOM, no special rules are used to construct the node sequence, instead, all options of the velocity sequence, which determine the number of nodes, are considered in the moment-inversion approach and the weight of each node is determined by mathematics of the definition of the options. This can lead to negative values of weights being observed in the simulation of discontinuous particle flow. To avoid the negative value, the offending weights are set to zero, as mentioned in the equations (46-49) in Fox [14]. Thus, the accuracy of the quadrature-based moment solution is reduced for dilute particle flows, such as those found in the applications of interest in this work. The proposed two-node system will always return positive weights.

3. Numerical Scheme

The transport step of the QMOM requires the moments to be updated using the weighted velocity nodes. The transport equation for the general moment M^3_{abc} can be written as

$$\frac{\partial M^3_{abc}}{\partial t} + \frac{\partial M^4_{abck}}{\partial x_k} = S \qquad (a,b,c=i,j,k) \tag{38}$$

where S represents source terms such as those due to drag or gravity. As the moment can be found from the two velocity nodes as

$$M^4_{abck} = n_1 U_{1a} U_{1b} U_{1c} U_{1\mathrm{k}} + n_2 U_{2a} U_{2b} U_{2c} U_{2k} \tag{39}$$

The transport equation can be rewritten as

$$\frac{\partial M^3_{abc}}{\partial t} + \frac{\partial n_1 U_{1a} U_{1b} U_{1c} U_{1k}}{\partial x_k} + \frac{\partial n_2 U_{2a} U_{2b} U_{2c} U_{2k}}{\partial x_k} = S \tag{40}$$

This conservative transport equation can then be converted into finite volume form using the divergence theorem.

$$\frac{\partial M^3_{abc}}{\partial t} + \frac{1}{V}\sum n_1 U_{1a} U_{1b} U_{1c} \mathbf{U}_1 \cdot \mathbf{A} + \frac{1}{V}\sum n_2 U_{2a} U_{2b} U_{2c} \mathbf{U}_2 \cdot \mathbf{A} = S \tag{41}$$

where *V* is the cell volume and the summation is over all faces enclosing a cell, and **A** is the directed area vector. While the calculations in this paper are carried out on a Cartesian grid the above formulation is valid on any arbitrary unstructured mesh. Note that at each cell face there will be two fluxes, one due to velocity node one and the other due to velocity node two. The time step is set to a fraction of the smallest characteristic time among convection (Δt_{CFL}), drag (τ_p) and collisions (τ_{coll}).

3.1. Spatial Discretisation Schemes

As with any finite volume transport scheme the remaining stage is to find the face fluxes using an appropriate discretisation scheme. It would be possible to apply the discretisation scheme to the moments to find the set of face moments and then convert them to the velocities, but following [15] we apply the discretisation schemes to the velocity and weight fields; this is to avoid a problem if high-order schemes are used to discretise the moments that may produce non-realizable combinations of moments.

For first-order transport a simple upwind scheme is used where $\phi_{\alpha,f}$ represents any velocity component of node α on face *f*.

$$\phi_{\alpha,f} = \phi_{\alpha,UP} \tag{42}$$

and 'UP' represents the cell upwind of the face based on $\mathbf{U}_\alpha \cdot \mathbf{A}$**.** For second-order transport a TVD scheme is used

$$\phi_{\alpha,f} = \phi_{\alpha,UP} + \frac{1}{2}\psi(\rho)(\phi_{\alpha,DOWN} - \phi_{\alpha,UP}) \tag{43}$$

$$\rho = \frac{\phi_{\alpha,UP2} - \phi_{\alpha,UP}}{\phi_{\alpha,DOWN} - \phi_{\alpha,UP}} \tag{44}$$

'UP2' represents the cell upwind of the upwind cell. The limiter function $\psi(\rho)$ needs to be specified and the second-order numerical schemes here uses Superbee which is tested and recommended by Desjardins [21].

$$\psi(\rho) = \max\big(0, \min(2\rho, 1), \min(\rho, 2)\big) \tag{45}$$

3.2. Boundary Conditions

To give the desired surface bouncing behaviour it is necessary to specify appropriate boundary conditions for the dispersed phase. Here we achieve this by using ‘ghost cells’ at the edge of the domain (seen in Figure 2) for which the two weighted velocity fields are found from the values in the first real cell according to

$$\begin{bmatrix} n_\alpha \\ U_{\alpha i} \\ U_{\alpha j} \\ U_{\alpha k} \end{bmatrix}_{gc} = \begin{bmatrix} n_\alpha / e_T \\ e_i U_{\alpha i} \\ e_j U_{\alpha j} \\ e_k U_{\alpha k} \end{bmatrix}_1 \tag{46}$$

where e_i, e_j and e_k are the coefficients of restitution in the three directions between the particle and wall. They are specified separately here for completeness in cases, such as those involving erosion, where the normal and tangential restitution coefficients are important. The total velocity coefficient of restitution e_T is defined as

$$e_T = \frac{\sqrt{(e_i U_i)^2 + \left(e_j U_j\right)^2 + (e_k U_k)^2}}{\sqrt{(U_i)^2 + \left(U_j\right)^2 + (U_k)^2}} \tag{47}$$

In the case of perfectly elastic collisions off a surface normal to the *i*-direction it should be noted that $e_i = 1, e_j = -1$ and $e_k = 1$.

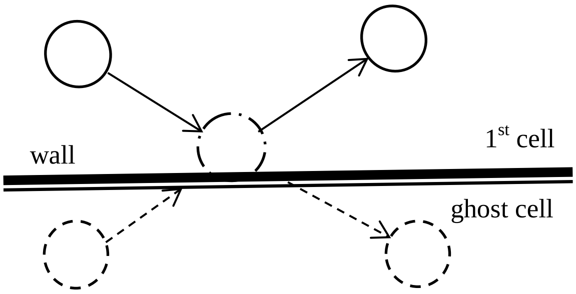


Figure 2. Scheme for boundary conditions

4. Test cases for crossing-trajectory and wall-rebounding particle flows

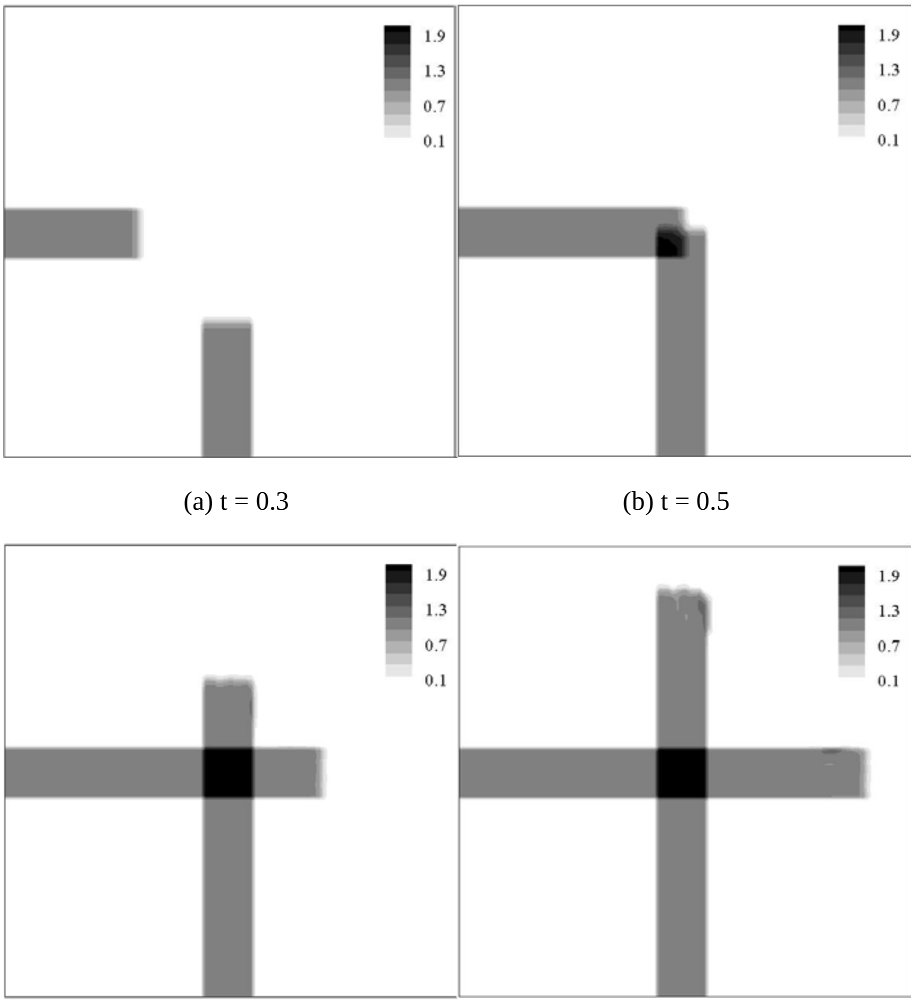
In this section, the two-node 13 moment QMOM model is used to simulate non-equilibrium dilute fluid-particle flows to test the general performance of the model in predicting trajectory crossing and wall-rebounding particle flows.

Initially this is done in the infinite Stokes number limit, to demonstrate the purely inertial behaviour of the particle streams, before moving on to demonstrate the capability to incorporate drag forces due to a carrier phase into the model.

4.1. Crossing Dilute, Infinite Stokes Number, Particle Jets

Here the improvement for crossing problems given by using cross-moments is demonstrated by a simple 2D two-stream problem. The flow domain has a non-dimensional size of 1×1. One particle stream is injected from the bottom wall of the domain, with the velocity at u_{1i} =0, u_{1j} =1, and the number density at n_1=1. The other particle stream is [n_2,u_{2i},u_{2j}] = [1,1,0].

The advantage of the 13-moment method proposed here (again, for this 2D case, only 8 moments are required) is that the correct combination of ε values can always be found. This is demonstrated in Figure 3 in which the two streams are seen to approach and then cross without interfering with each other. The result seen in Figure 1(b) where [n_1,u_{1i},u_{1j}] = [1,0,0] and [n_2,u_{2i},u_{2j}] = [1,1,1] in the crossing zone is a valid solution to the set of moments produced in that location. The desired crossing behaviour is *also* a valid solution to the eight moments but there is not enough information contained within the moments to choose between the solutions. By forcing the velocity nodes to conform to extra moments (particularly cross-moments) it is possible to obtain the correct crossing solution such as [n_1,u_{1i},u_{1j}] = [1,0,1] and [n_2,u_{2i},u_{2j}] = [1,1,0] without further assumption.



(a) t = 0.3

(b) t = 0.5

(c) t = 0.7

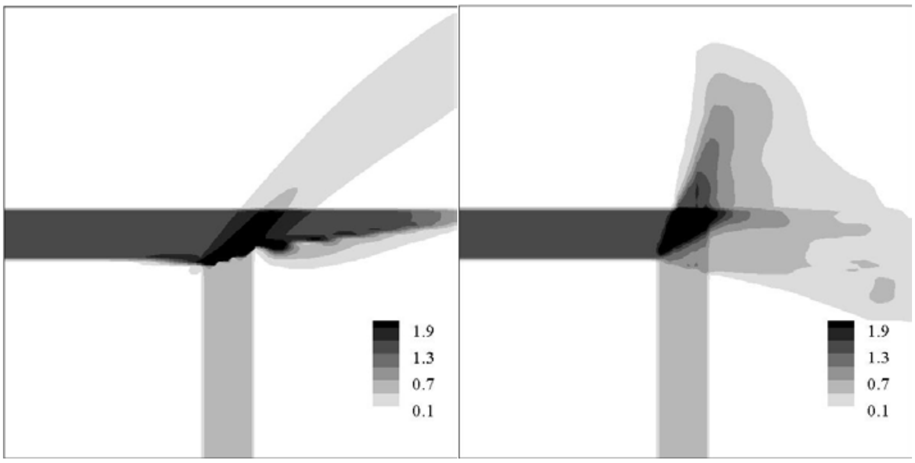
(d) t = 0.9

Figure 3. Temporal evolution of contours of particle number density for two dilute jets with boundary conditions [n_1,U_{1i},U_{1j}] = [1,0,1] and [n_2,U_{2i},U_{2j}] = [1,1,0] simulated using the two-node 13-moment method.

The addition of cross-moments alone does not guarantee correct recovery of the velocity fields and weights in all cases without the extra third moments, $M_{iii}^3, (i = i, j, k)$. Figure 5 shows the effect of using cross moments to find the ε values, but using the sum of the third order moments $Q = \sum_{i=1}^{d} M_{iii}^3$ rather than the individual values, $M_{iii}^3, (i = i, j, k)$. We see once again that in Figure 4(a) the original six moments fail to give the correct crossing behaviour, however, in Figure 4(b) we see that even with the cross moments used the correct behaviour is not recovered. The summation of the normal third order moment Q used in the two-node QMOM fails to identify some cases where $q_p = \sum_{i=1}^{d} a_{iii} = 0$ and $\mathrm{a}_{iii} \neq 0$. For example, in Figure 4, if the initial two node sequences are [n_1,U_{1i},U_{1j}] = [1.5,0,1] and [n_2,U_{2i},U_{2j}] = [0.5,1,0], then from the two-node closure approach $q_p = 0$. This means that the number density of the two nodes will be calculated to be equal. By calculating a_{iii} separately this condition can be avoided.

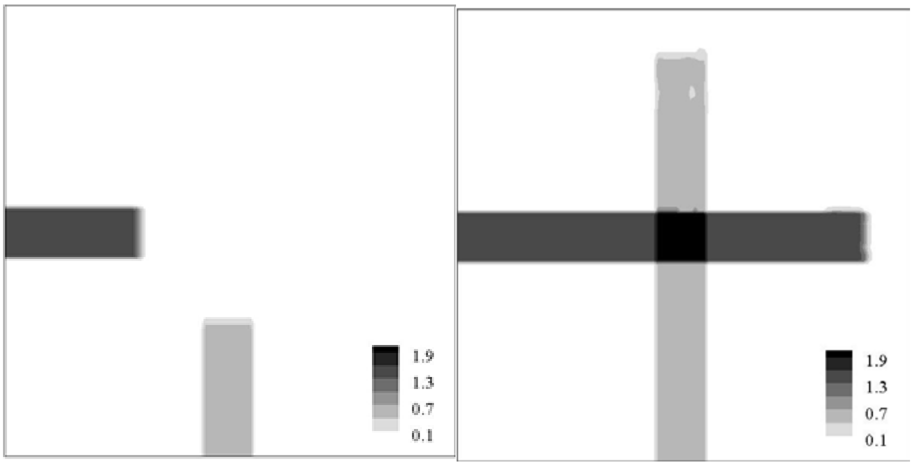
Figure 5 shows the result of crossing particle jets by the two-node, 13 moment QMOM when the boundary condition is [n_1,U_{1i},U_{1j}] = [1.5,0,1] and [n_2,U_{2i},U_{2j}] = [0.5,1,0]. The number densities at t = 0.3 before crossing and t =

0.9 after crossing are shown and the crossing trajectories of the two particle jets are correctly predicted. In the region that the two jets cross, the lateral third-order moment of velocity deviation is negative, a_{iii} =-0.08, while the vertical third-order moment is positive, a_{jjj} = 0.08. Using the two-node thirteen-moment quadrature closure, $|x| = 0.25$, from which the number density of two nodes can be calculated correctly as n_1 = 1.5 and n_2 = 0.5. If the summation of the third-order moment is used, as in [21], then $q = a_{iii} + a_{jjj} = 0$ and $x = 0$, which leads to the mistaken result that n_1= n_2 = 1.0 as seen in Figure 4b. Thus, three third-order normal moments of particle velocity are necessary for the crossing jets with any particle flow rates.



(a) Two-node, 8-moment QMOM (b) two-node, 11 moment QMOM

Figure 4. Particle number density for crossing trajectory jets ([n_1,U_{1i},U_{1j}] = [1.5,0,1] and [n_2,U_{2i},U_{2j}] = [0.5,1,0]), (a) shows result using original six moment set from [21] and (b) shows result using additional cross-moments but not individual third moments.

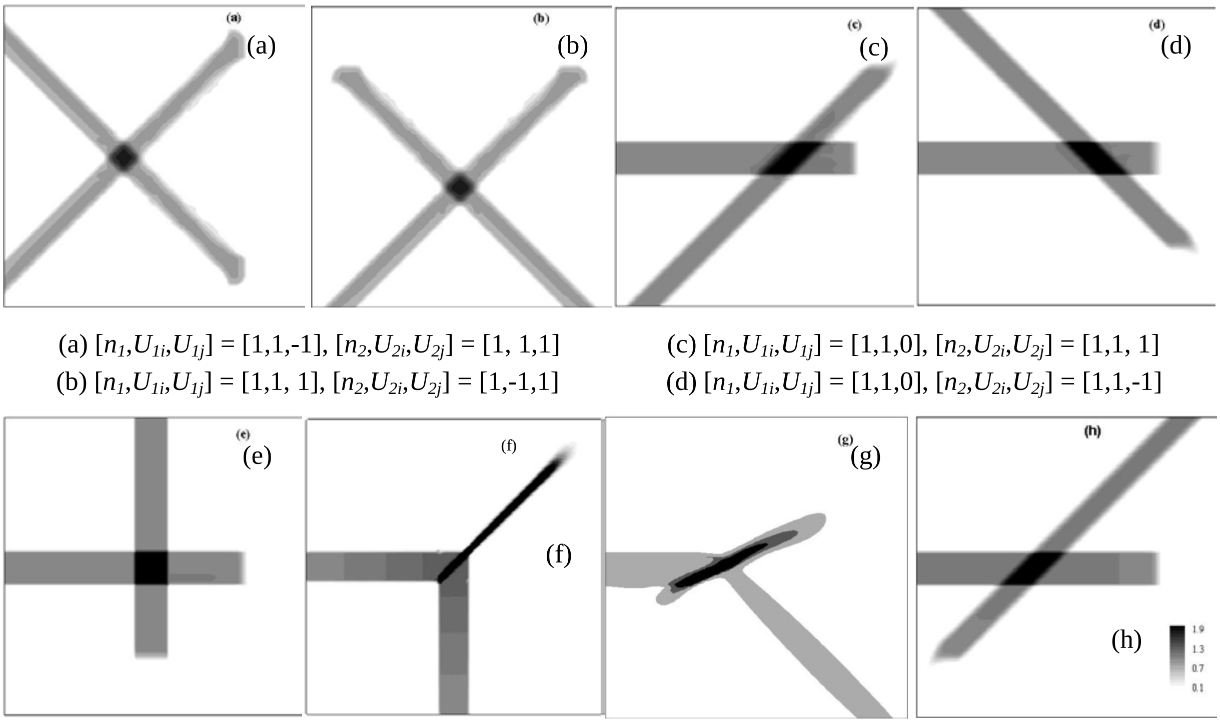


(a) M^0 at t = 0.3 (b) M^0 at t = 0.9

Figure 5. Crossing particle jets with boundary conditions [n_1,U_{1i},U_{1j}] = [1.5,0,1] and [n_2,U_{2i},U_{2j}] = [0.5,1,0]. Contour plots of total number density at t=0.3 (a) and t=0.9(b).

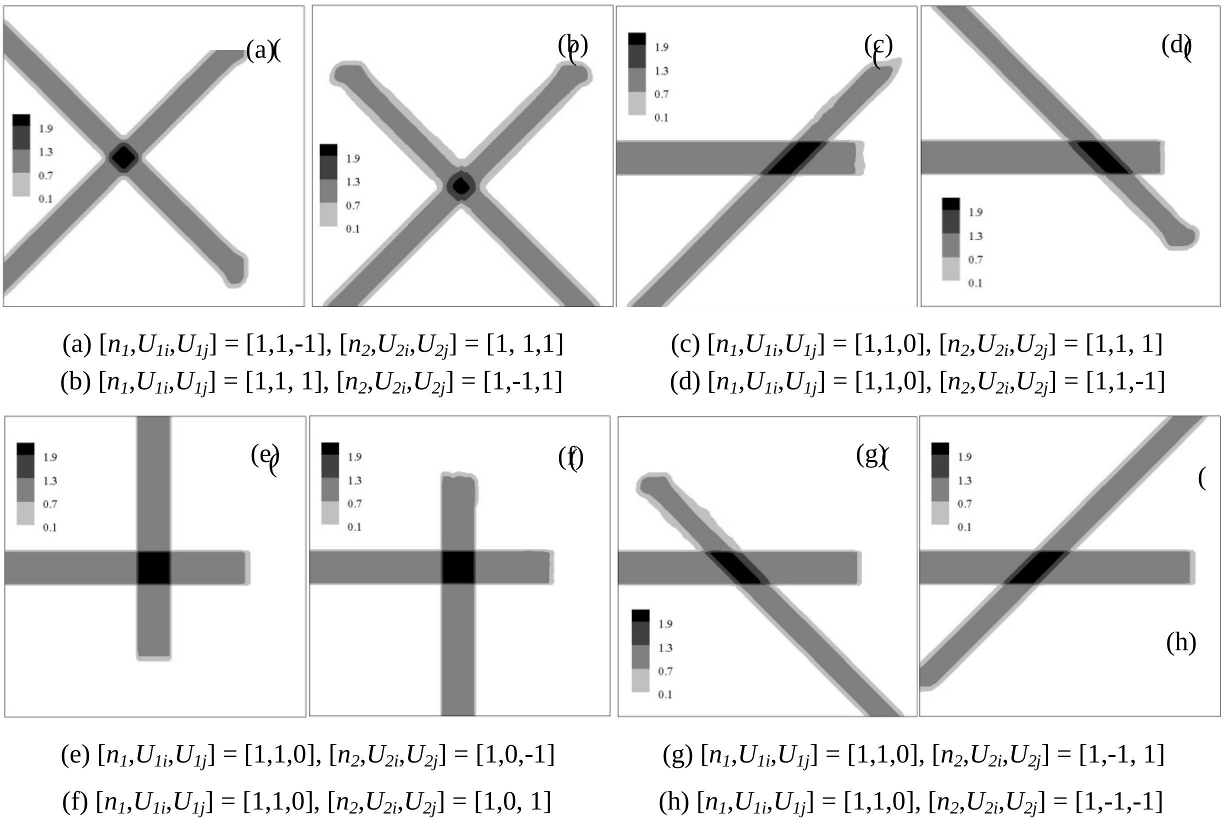
4.2. Crossing Dilute, Infinite Stokes Number, Particle Jets at Arbitrary Angles

In this section, the general validity of the two-node, 13-moment is tested. Arbitrary angles between walls and oncoming streams can be represented by various injection angles and crossing angles of particle jets. The results obtained using the two-node eight-moment method [21] are shown in Figure 6. Note, in particular the failed cases observed in Figure 6(f) & (g) for which the assumption that $U_{2j} > U_{1\mathrm{j}}$ and $U_{2i} > U_{1\mathrm{i}}$ in the conversion from moments to velocities means that the correct velocity combination is not associated with each node. The same tests with the two-node, 13-moment QMOM are shown in Figure 7. It can be seen that the correct behaviour, with dilute jets crossing without interference, can be seen for all eight cases.



(a) $[n_1,U_{1i},U_{1j}] = [1,1,-1]$, $[n_2,U_{2i},U_{2j}] = [1, 1,1]$
(b) $[n_1,U_{1i},U_{1j}] = [1,1, 1]$, $[n_2,U_{2i},U_{2j}] = [1,-1,1]$
(c) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,1, 1]$
(d) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,1,-1]$
(e) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,0,-1]$
(f) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,0, 1]$
(g) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,-1, 1]$
(h) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,-1,-1]$

Figure 6. Contour plots showing particle number densities for crossing dilute jets with QMOM method of [21].



(a) $[n_1,U_{1i},U_{1j}] = [1,1,-1]$, $[n_2,U_{2i},U_{2j}] = [1, 1,1]$
(b) $[n_1,U_{1i},U_{1j}] = [1,1, 1]$, $[n_2,U_{2i},U_{2j}] = [1,-1,1]$

(c) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,1, 1]$
(d) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,1,-1]$

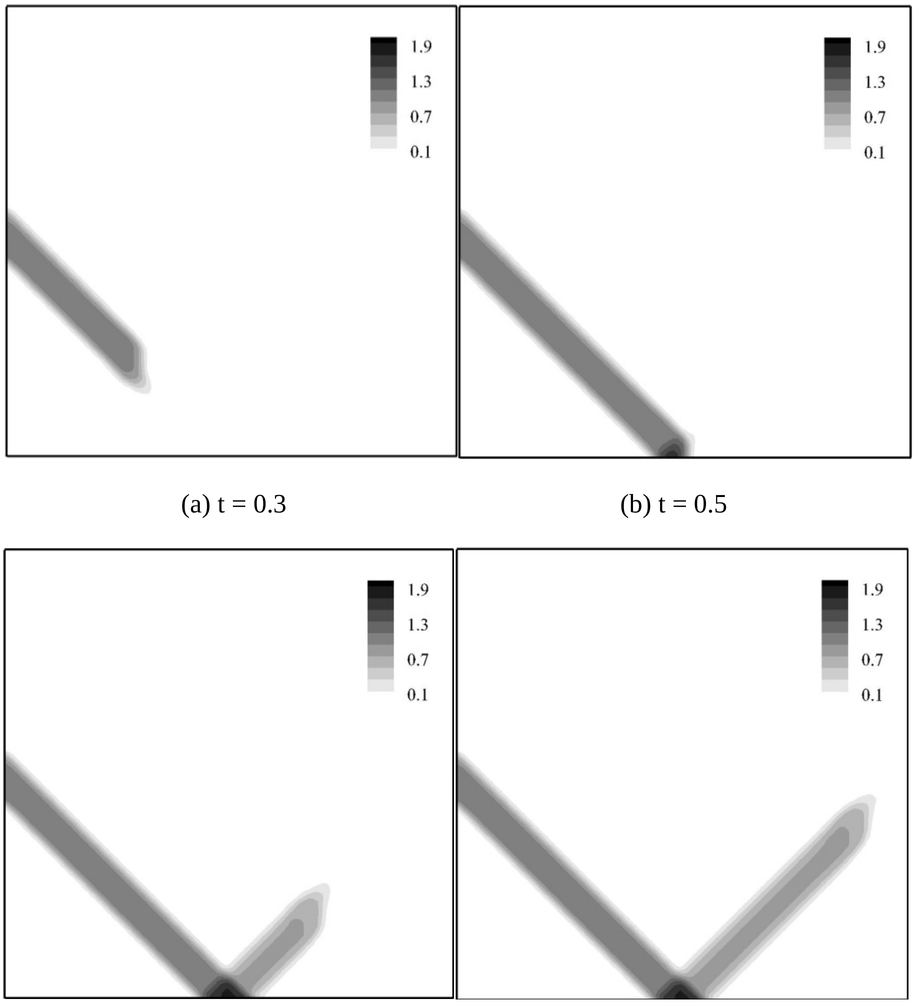
(e) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,0,-1]$
(f) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,0, 1]$

(g) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,-1, 1]$
(h) $[n_1,U_{1i},U_{1j}] = [1,1,0]$, $[n_2,U_{2i},U_{2j}] = [1,-1,-1]$

Figure 7. Contour plots showing particle number densities for crossing dilute jets with new two-node, 13-moment QMOM presented in this work.

4.3. Wall Bouncing of Dilute, Infinite Stokes Number, Particle Jets

Figure 8 shows the prediction of particle wall rebounding. One particle flow stream ($[n_1,U_{1i},U_{1j}] = [1,1,-1]$) is injected into the flow field towards the bottom wall (note that the left hand boundary condition is n_2=0). At t=0.5, the particles collide with the wall, and then bounce off the wall. Near the wall, the number density of particles is doubled resulting from the overlap of injecting and rebounding particles, which means that two flow streams are crossing at the wall. For this elastic and smooth wall where $e_j = -1$, the two node values become $[n_1,U_{1i},U_{1j}] = [1,1,-1]$ and $[n_2,U_{2i},U_{2j}] = [1,1,1]$ at the collision point.



(a) t = 0.3

(b) t = 0.5

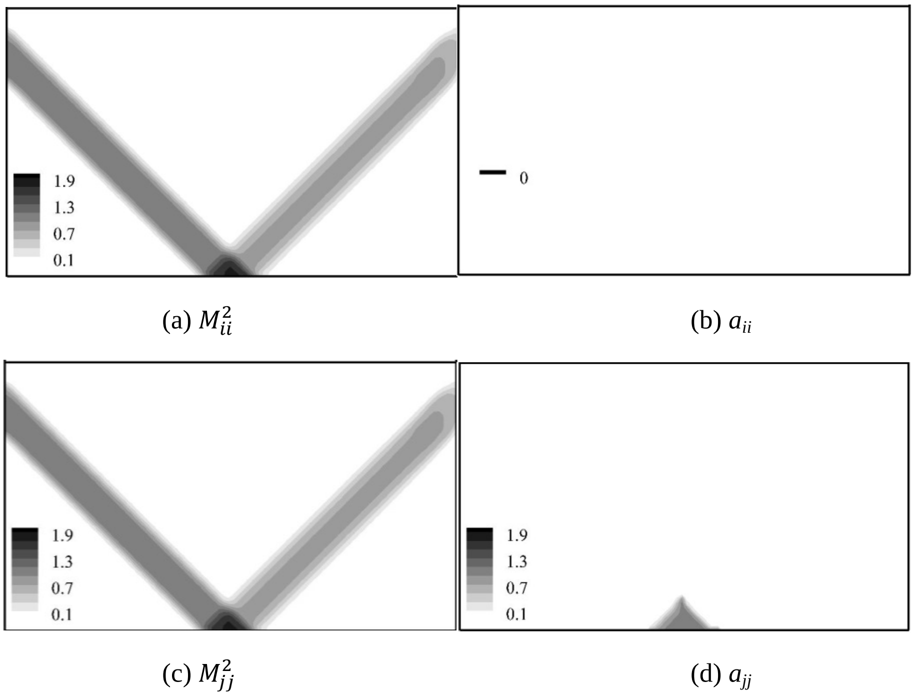
(c) t = 0.7

(d) t = 0.9

Figure 8.Time evolution of particle number density for wall rebounded particle flow.

It can be seen that the wall rebounding problem can be considered equivalent to a crossing-trajectory problem. Taking the injecting flow as one stream, the rebounding flow can be taken as the other. The collision point at the wall is the pseudo trajectory-crossing point. Figure 2 illustrates the relationship between the injecting jet and the rebounding jet visually. The particles below the wall are considered as the pseudo particles, which can supply the particle source of the rebounding jet and absorb the injecting particles.

The strategy of the two-node QMOM is to separate the main flow into two streams by means of the reassociated velocity fields. The distribution of moment of particle velocity and velocity deviation are plotted in Figure 8. At the collision point, n_1=n_2, u_1=u_2, and v_1= - v_2. Normal moments of particle velocities are calculated as $M_{ii}^2 = n_1(U_{1i})^2 + n_2(U_{2i})^2$, hence the two normal second-order moments are the same. Whereas the moments of velocity deviation, calculated by $a_{ii} = M_{ii}^2/M^0 - \left(U_{\mathrm{p}i}\right)^2$, represent the magnitude of the deviation of particle velocities between the two nodes. In the *i* direction, velocity deviation c_i = 0, then a_{ii} =0, as shown in Figure 8(a). But in the *j* direction, $a_{jj} \neq 0$, then $\mathbf{c_1}$ and $\mathbf{c_2}$ can be calculated from a_{jj} with the quadrature approach mentioned in section 2. Even though, at the particle-wall collision point, the averaged normal velocity in the normal direction j is zero, the velocity deviation is non-zero calculated by the velocity moment and mean velocity. In this way, the two stream flows are separated by the QMOM method.



(a) M_{ii}^2 (b) a_{ii}

(c) M_{jj}^2 (d) a_{jj}

Figure 2. Distributions of moments of particle velocity and velocity deviation

5. Particle Motion For Finite Stokes Number

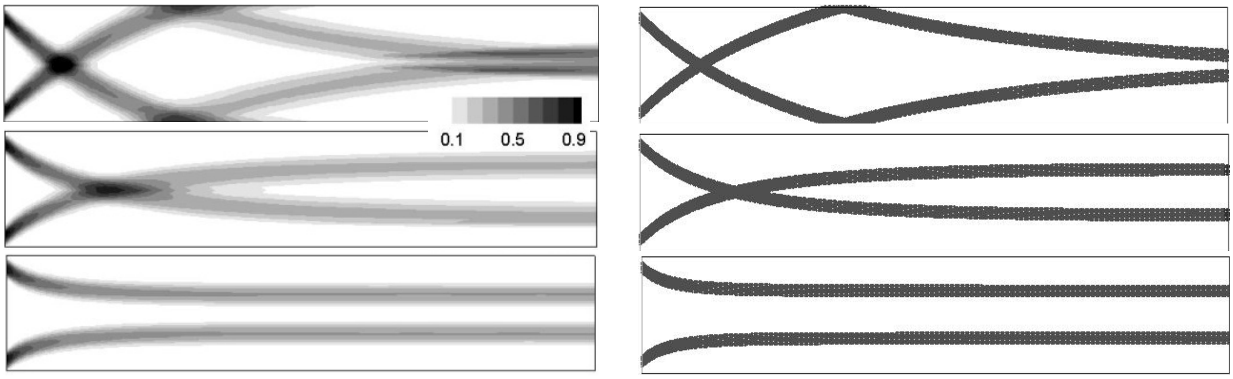
In this section we demonstrate the ability of the two-node, 13-moment QMOM to predict behaviour seen in real world dilute gas-particle flows. We consider particle streams in a continuous gas carrier phase using a range of Stokes numbers, this demonstrates that the drag terms are included correctly in the moment equations, including the cross-moment terms, before moving on to the more complex Taylor-Green test case.

5.1. Effect of Varying Stokes Number

Depending on its inertia a particle's trajectory will be influenced by the drag force imparted by the carrier fluid. For Stokes number St << 1, particles follow the streamlines closely, while for St >>1 the particle response time is larger than the smallest time scale of the flow and the particle will follow a ballistic trajectory. We have already seen simulations of particles with infinite Stokes number in the previous sections. The effect of different Stokes number is tested in this study, which is the ratio of the characteristic time of a particle to the characteristic time of the flow,

$$St = \frac{\rho_p (d_p)^2}{18\mu_f D} |\mathbf{U_f} - \mathbf{u}| \tag{48}$$

Simulations are carried out in a 2-D straight channel, where the flow domain is 0.2m×1 m. The grid cell size is 0.01m×0.01 m. The carrier gas enters the pipe from the left inlet and leaves via the right outlet boundary. The top and bottom walls both give elastic collisions for the particles. Two streams of particles are injected from the left inlet of the pipe. The inlet conditions for the two particle jets are $[n_1,u_{1i},u_{1j}] = [1,1,1]$ and $[n_2,u_{2i},u_{2j}] = [1,1,-1]$. The particle density is 2400kg/m^3, and three types of particle diameters are tested 0.08mm, 0.15mm and 0.25mm. One way coupling of fluid and particle momentum transportation is used in this simulation. Air at 1 atm and ambient temperature is used as the fluid phase with uniform velocity $[u_{gi},u_{gj}] = [1.2, 0]$ in the flow domain.



(a) QMOM

(b) Lagrangian model

Figure 9. Dilute particle jets under the influence of drag from a carrier phase at different Stokes number. (a) Contour plots of particle number density using the proposed QMOM method. (b) Particle tracks from Lagrangian DPM model [28]. (St ≈ 2.78 (top), 1.00 (middle) and 0.28 (bottom))

Figure 9 shows the number densities of particles for the three Stokes number cases. To provide validation data the same case is recreated using a Lagrangian method. The Lagrangian results were produced using the DPM solver in ANSYS Fluent [28]. The DPM model is used to trace the path of the particles, without collisions between particles and without two-way coupling between phases. The mass flow rates of inlet particles are calculated from the volume fractions and particle density as mentioned above. Near the particle jet inlets, the initial Stokes numbers of the three particle diameters are approximately 2.78 for the 0.25mm particle, 1.00 for 0.15mm and 0.28 for 0.08mm. In all three cases the drag force causes the particle streams to be swept downstream. For the largest particles (and hence highest inertia) we see that the particle streams cross before colliding with the opposite wall and being reflected. As particle size is reduced, the particles are swept more strongly downstream with the 0.15 mm particles still crossing each other but not reaching the opposite wall. For the 0.08 mm particles their inertia is so small that the streams never meet each other as they are swept very quickly downstream. We have already seen particle flows with infinite Stokes number and for Stokes numbers approaching zero the particle will be coupled strongly with the carrier flow. Hence, we can be confident that the method used here is capable of predicting flow under a range of Stokes number in flows found in chemical engineering or turbomachinery applications.

5.2. Taylor-Green Flow

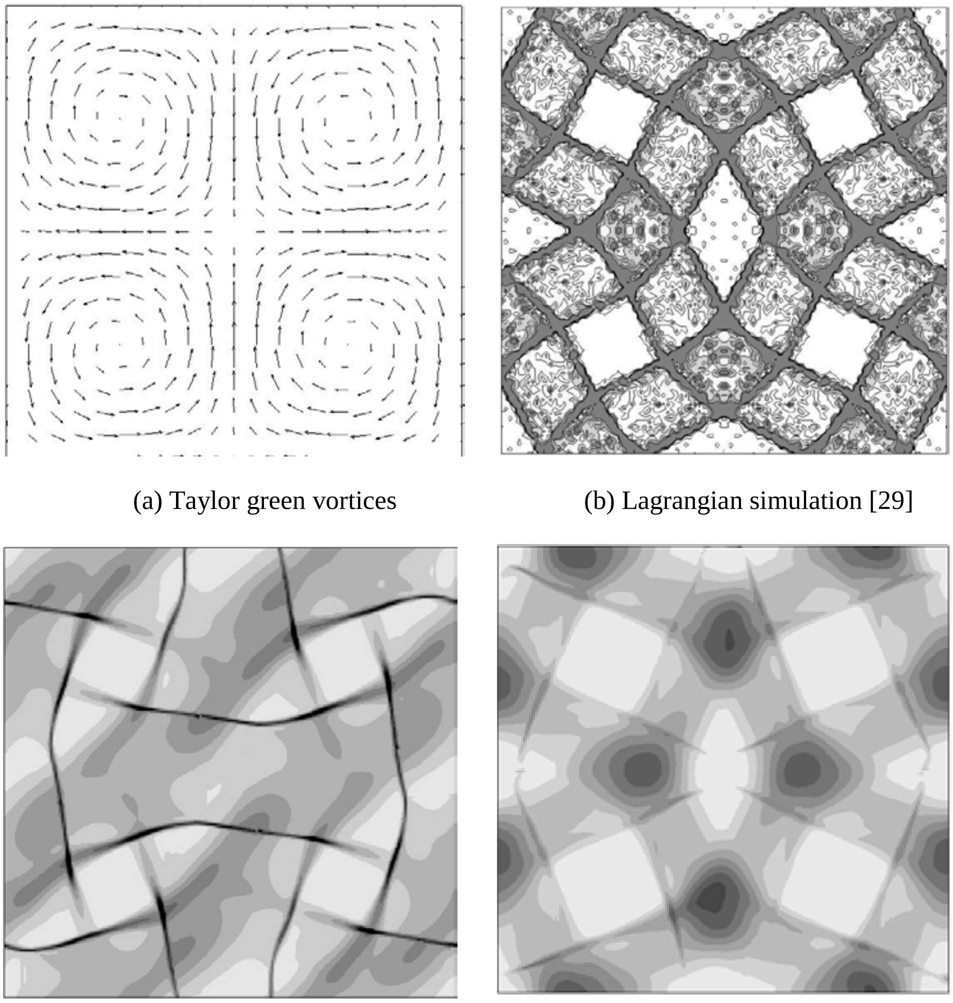
Results using the 2D Taylor-Green vortex flow test are presented here to test the capability of our method to predict the development of a more complex dilute particle flow field. This test case has previously been used in [15] to demonstrate the ability of a four-node QMOM method to predict the development of particle concentration in a fixed continuous phase flow field. This flow consists of four periodic vortices, defined as,

$$u_{gi} = \quad \sin(2\pi x)\cos(2\pi y) \tag{49}$$

$$u_{gj} = -\sin(2\pi y)\cos(2\pi x) \tag{50}$$

where the subscript g indicates that this is the gas flow field. The non-dimensional gas maximum velocity is 1, and the vector field is shown in Figure 10(a). The non-dimensional 2-D flow domain size is 1 × 1 with 200 × 200 grid cells, and periodic boundaries are used at the edge of the domain. The initial number density of particles is uniform in the flow domain. As a benchmark Figure 10(b) shows the particle number densities predicted by Le Lostec et al [29] with the Lagrangian method, where St=1.0 at non-dimensional time t=4. For comparison Figure 10(c) shows the results with the original two-node eight-moment QMOM by Desjardins [21]. It can be seen that the pattern seen in the Lagrangian results is not reproduced as the 2-node 8-moment QMOM fails to produce the rotationally repeating characteristic of the particle distribution. This is due to the assumption made in the quadrature approach that $(u_{1i}\text{-}u_{2i})(u_{1j}\text{-}u_{2j})<0$. Streaks of high particle density similar to those seen in the failed verification tests of Figure 6(f) & (g) can be observed.

The particle crossing trajectories in the vortices in Figure 10(d) are predicted with the new two-node QMOM model. The high density particle belts are successfully predicted. Low particle density in the centre of the domain and near the centre of the edges of the domain are observed in Figure 10 (d), as also seen in Figure 10(b) simulated by the Lagrangian method. The result seen in Figure 10(d) is also very similar to that observed using a higher node method in [15]. Therefore, the method presented here has been demonstrated to predict the correct development of a dilute particle flow field under a complex 2D flow field with flow curvature and finite Stokes number. This is an important test as these conditions will be encountered in real applications, such as those found in gas turbines for example. To investigate the computational advantage of using a two-node over a higher order method we also implemented the method used in [15] into our own code which was found to give similar results to theose reported in [15]. The relative run times of the orginal method of [21], our proposed method and the higher order method from [15] were then compared using a single processor. Relative to the run times for our proposed method the method of [21] was found to take 0.69 of our method and the higher order method was found to take 1.52. A 33% reduction in computing time compared to the high order method could be significant in a large calculation and it should be kept in mind that the difference in nodes and moments (and hence cost) would be even greater for a three dimensional case. This gives confidence in our method's capabilities for future practical engineering test cases.



(a) Taylor green vortices

(b) Lagrangian simulation [29]

(c) 2-node 8-moment QMOM [21]

(d) 2-node, 13-moment QMOM

Figure 30. Particle number densities in Taylor-Green flow where St =1, *t*=4

6. Conclusions

A new two-node, 13-moment quadrature method of moments QMOM has been developed to allow robust calculation of crossing and bouncing trajectory dilute gas-particle flows at high Knudsen number with a relatively low computational cost. The method builds on the existing two-node, eight-moment QMOM [21] but uses additional third-order moments and second-order cross-moments. Without the extra moments it was necessary to make choices between equally valid solutions of the moment to velocity conversion step, based arbitrarily on the reference frame. By requiring the two weighted velocity fields to conform to these extra moments it is possible to ensure that the velocities recovered from the set of moments are those that give the correct stream crossing behaviour for any crossing angle without any arbitrary decisions. The extra moments, particularly the cross moments, are needed to ensure that sufficient physical information is contained in the set of moments. While any moment of velocity can be found from two or more weighted velocity fields, it is important to make sure that the set of moments chosen can be converted back to the velocity fields in all cases. This also highlights the importance of using a thorough set of verification tests, for which a solution is known, when constructing QMOM schemes.

For cases with large or complex geometries, such as those found in turbomachinery, the full detail of the existing third-order QMOM [14] may not be required and solving for 20 moments and eight weighted velocity fields (in 3D) will be undesirably expensive. The method proposed here only needs 13 moments and two weighted velocity fields. This reduces both computation and memory storage overheads.

To demonstrate the method, 2-D particle flows have been simulated using the new model. Arbitrary crossing-trajectory and wall rebounding particle flows have been tested. The results show the correct behaviour for all cases tested, and, the model is shown to be able to handle non-equilibrium particle flows. The new method has been extended to include drag terms to demonstrate that it is capable of producing results with a range of Stokes numbers. The method has also been tested for a complex 2D dilute particle flow by simulating the development of particle number density in Taylor-Green vortex flow. The results are seen to be in good agreement with those found using a Lagrangian method and higher node QMOM methods. The results are superior to those produced with the pre-existing two-node method. The excellent performance in the Taylor-Green test case gives confidence that the method can be applied to cases with finite Stokes numbers and flow curvature as would be found in practical engineering applications. Future work will be aimed at applying the method to cases such as the ingestion of particles into the inlet of gas turbine aircraft engines.

Acknowledgement

This work has been funded by the Innovate UK (formerly the United Kingdom Technology Strategy Board, TSB) under SILOETII, in conjunction with Rolls-Royce PLC.

References

[1] A. Hamed and W. Tabakoff, “Erosion and deposition in turbomachinery,” *Journal of Propulsion and Power,* vol. 22, pp. 350-360, 2006.

[2] J. A. Laitone, “Characterization of particle rebound phenomena in the erosion of turbomachinery,” *Journal of Aircraft,* vol. 20, pp. 275-281, 1983.

[3] S. Nilamdeen and W. G. Habashi, “Multiphase Approach Toward Simulating Ice Crystal Ingestion,” *Journal of Propulsion and Power,* vol. 27, pp. 959-969, 2011.

[4] H. Beaugendre, F. Morency and W. G. Habashi, “ICE3D, FENSAPICE’s 3D In-Flight Ice Accretion Module,” *Journal of Aircraft,* vol. 40, pp. 239-247, 2003.

[5] M. Sommerfeld, “Modelling of particle-wall collisions in confined gas-particle flows,” *International Journal of Mulitphase Flow,* vol. 18, pp. 905-926, 1992.

[6] M. Sommerfeld and C. A. Ho, “Numerical calculation of particle transport in turbulent wall bounded flows,” *Powder Technology,* vol. 131, pp. 1-6, 2003.

[7] K. D. Squires and O. Simonin, “LES-DPS of the effect of wall roughness on dispersed-phase transport in particle laden turbulent channel flow,” *International Journal of Heat and Fluid Flow,* vol. 27, pp. 619-626, 2006.

[8] C. Tropea, A. Yarin and J. Foss, Springer handbook of experimental fluid mechanics, New York: Springer, 2007.

[9] C. K. K. Lun and H. S. Liu, “Numerical simulation of dilute turbulent gas-solid flows in horizontal channels,” *International Journal of Multiphase Flow,* vol. 23, pp. 575-605, 1997.

[10] S. K. Pawar, J. T. Padding, N. G. Deen, A. Jongsma, F. Innings and J. A. M. Kuipers, “Lagrangian modelling of dilute granular flow - modified stochastic DSMC versus deterministic DPM,” *Chemical Engineering Science,* vol. 105, pp. 132-142, 2014.

[11] M. Sommerfeld, “Analysis of collision effects for turbulent gas-particle flow in a horizontal channel: Part I. Particle transport,” *International Journal of Multiphase Flow,* vol. 29, pp. 675-699, 2003.

[12] M. Sommerfeld and J. Kussin, “Analysis of collision effects for turbulent gas-particle flow in a horizontal channel. Part II. Integral properties and validation,” *International Journal of Multiphase Flow,* vol. 29, pp. 701-718, 2003.

[13] R. O. Fox, “Large-eddy simulation tools for multiphase flows,” *Annual Review of Fluid Mechanics,* vol. 44, pp. 47-76, 2012.

[14] R. O. Fox, “A quadrature-based third-order moment method for dilute gas-particle flows,” *Journal of Computational Physics,* vol. 227, pp. 6313-6350, 2008.

[15] V. Vikas, Z. J. Wang, A. Passalacqua and R. O. Fox, “Realizable high-order finite-volume schemes for quadrature-based moment methods,” *Journal of Computational Physics,* vol. 230, pp. 5328-5352, 2011.

[16] A. Passalacqua, R. O. Fox, R. Garg and S. Subramaniam, “A fully coupled quadrature-based moment method for dilute to moderately dilute fluid-particle flows,” *Chemical Engineering Science,* vol. 65, pp. 2267-2283, 2010.

[17] A. Passalacqua and R. O. Fox, “Advanced continuum modelling of gas-particle flows beyond the hydrodynamic limit,” *Applied Mathematical Modelling,* vol. 35, pp. 1616-1627, 2011.

[18] V. Vikas, C. Yuan, Z. J. Wang and R. O. Fox, “Modelling of bubble column flows with quadrature-based moment methods,” *Chemical Engineering Science,* vol. 66, pp. 3058-3070, 2011.

[19] L. Schneider, N. Le Lostec, P. Villedieu and A. Sadiki, “A moment method for splashing and evaporation processes of polydisperse sprays,” *International Journal of Multiphase Flow,* vol. 36, pp. 261-272, 2010.

[20] R. O. Fox, “Higher-order quadrature-based moment methods for kinetic equations,” *Journal of Computational Physics,* vol. 228, pp. 7771-7791, 2009.

[21] O. Desjardins, R. O. Fox and P. Villedieu, “A quadrature-based moment method for dilute fluid-particle flows,” *Journal of Computational Physics,* vol. 227, pp. 2514-2539, 2008.

[22] H. M. Hulburt and S. Katz, “Some problems in particle technology: A statistical mechanical formulation,” *Chemical Engineering Science,* vol. 19, pp. 555-574, 1964.

[23] S. E. Pratsinis, “Simultaneous nucleation, condensation and coagulation in aerosol reactors,” *Journal of Colloid and Interface Science,* vol. 124, pp. 416-427, 1988.

[24] R. McGraw, “Description of aerosol dynamics by the quadrature method of moments,” *Aerosol Science and*

Technology, vol. 27, pp. 255-265, 1997.

[25] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, Numerical Recipes, Cambridge, 1990.

[26] H. Grad, “On the kinetic theory of rarefied gases,” *Communications on Pure and Applied Mathematics,* vol. 2, pp. 331-407, 1949.

[27] L. Schiller and A. Naumann, “Uber die grundlegenden berechnungen bei der schwerkraftaufbereitung,” *Zeitung des vereins deutscher ingenieure,* pp. 318-320, 1933.

[28] ANSYS, “Fluent 14.0 User Manual,” 2011.

[29] N. Le Lostec, R. O. Fox, O. Simonin and P. Villedieu, “Numerical description of dilute particle-laden flows by a quadrature-based moment method,” in *Proceedings of the Summer Program*, Stanford University, 2008.