Implementation of CHyQMOM in OpenFOAM for the simulation of non-equilibrium gas-particle flows under one-way and two-way coupling

Implementation of CHyQMOM in OpenFOAM for the simulation of non-equilibrium gas-particle flows under one-way and two-way coupling

Dongyue Li[a,b], Daniele Marchisio[c,*]

[a]*State Key Laboratory of Advanced Metallurgy, University of Science and Technology Beijing, Beijing, China*
[b]*DYFLUID Ltd, Beijing, China*
[c]*DISAT, Institute of Chemical Engineering, Politecnico di Torino, Torino, Italy*

Abstract

The modeling of dilute gas-particle flows is challenging due to particle-trajectory crossing (PTC). Lagrangian particle tracking can be used, but requires a large number of parcels resulting in high computational costs. A less costly method is the Eulerian number density function (NDF) approach, based on the Boltzmann equation, often solved in terms of lower-order moments of the NDF. In this context the conditional hyperbolic quadrature method of moments (CHyQMOM) was developed and is here implemented for the first time in the OpenFOAM-7, together with high-order advection schemes and a new operator splitting procedure. The resulting solver is used to simulate different test cases: phase segregation problems, collision-less and weakly-collisional PTC flows, asymmetric and symmetric Taylor-Green vortex flow and a dilute gas-particle riser. Results, validated against analytical solutions and predictions obtained with Lagrangian particle tracking, show that the implemented CHyQMOM can be used to handle highly non-equilibrium flows.

Keywords: Gas-particle flow, E-QBMM, HyQMOM, OpenFOAM, Multiphase systems

1. Introduction

Computational fluid dynamics (CFD) has been used to simulate gas-particle flows that find application in many different industries, resulting in the development of a plethora of models and methods with their own pros and cons.

One very popular methodology is the use of Lagrangian particle tracking, that can provide high-fidelity solutions for dilute flows. Since each particle is tracked separately, it can successfully capture particle trajectory crossing (PTC) without any numerical problem. One disadvantage is that it scales poorly for large problems and the statistical error can only be reduced by employing a large number of particles, typically resulting in very high computational costs. Moreover, for two-way coupling problems, the particle diameter should be much smaller than the cell size. Last but not least, the

*Corresponding author, Tel: +39 0110904622, Fax: +39 0110904699, Email: daniele.marchisio@polito.it

Lagrangian particle tracking method is not efficient for highly-collisional flows due to the large amount of collisions that slows down the algorithm, due to the extremely small collision time-scales.

An interesting alternative is the use of the so-called two-fluid model (TFM), in which the particle phase is treated as continuous and it interacts with the gas via momentum coupling, written in terms of phase volume fractions and mean phase velocities. Each phase is governed by its own momentum balance Navier-Stokes (NS) equation. The TFM is computationally affordable and has been used extensively for simulating large-scale industrially-relevant flows [47, 38, 11]. However, it is well known that the NS equations are applicable only for collision-dominated flows. For rarefied non-equilibrium flows, the Navier-Stokes equations are invalid. It was also shown that the TFM fails in predicting PTC [8] and predicts nonphysical delta shocks (i.e. regions characterised by high particle concentrations) [54]. This results from the fact that in the TFM there exists only one average value of the particle velocity at each spatial location. For dilute flows however, particles tend to cross each other, often resulting in complex velocity distributions with zero mean. One possible solution is to allow the particle phase to assume more than one single characteristic velocity, as for example in the multi-fluid model [48].

In order to model non-equilibrium particle flows, a kinetic equation for the solid particles can be used, coupled with the NS equation for the gas phase. Theoretically, the discrete particles and gas molecules can be treated with a similar approach, but in this work we focus on the solid particles only, characterized by the number density function (NDF) for the particle velocity, which is in turn governed by the well-known kinetic equation or Boltzmann equation.

However, the direct numerical solution of the kinetic equation is intractable for most applications, due to the large number of independent variables. Different solution schemes were developed to handle the closure problem. In the unified gas-kinetic scheme (UGKS) family [57] particle transport and collision are closely coupled for both flux transport and inner cell evolution by the Bhatnagar-Gross-Krook (BGK) approximation, in turn based on the Chapman–Enskog expansion and the scheme is applicable from rarefied to continuous flows. The UGKS was used to simulate gas-particle flows [39] and rarefied flows [56, 27]. One can also use the moments of the NDF to solve the kinetic equation, as in the pioneering work of Grad [20] later extended by Struchtrup and Torrilhon [50] and employed to simulate specific problems [23, 24]. Recently the Grad method was extended to 29 equations [22] and coupled with the maximum-entropy method [49, 41, 42, 21, 16].

Besides the methods mentioned above, one can also use quadrature-based moments methods (QBMM) [40] in which the Boltzmann equation is solved in terms of the moments of the NDF and the closure problem is overcome by assuming a functional form for the NDF. This presumed NDF reconstructs the population of particles with a finite set of "particle classes" corresponding to nodes or

abscissas of a quadrature approximation, as detailed in the first formulation of the quadrature method of moments (QMOM). The weights and nodes/abscissas are determined by forcing them to agree with as many moments of the NDF as possible [17]. For univariate problems (i.e., one-dimensional velocity space), the product-difference algorithm can be used [43]. For multi-dimensional problems (i.e., two- or three-dimensional velocity space) different methods were proposed. Fox [17, 18] firstly developed the so-called tensor-product (TP) algorithm in the context of gas-particle flows, although several years earlier TP was also developed by Yoon and McGraw for aerosol modeling [58]. However, for highly multi-dimensional distributions, since only pure moments are used and the nodes/abscissas are fixed, sometimes the weights can be negative. Cheng and Fox [9] and Cheng et al. [10] developed the conditional quadrature method of moments (CQMOM) which overcomes this problem. CQMOM was later applied to one-way coupled gas-particle flows [59]. In order to address degenerate cases, the adaptive Wheeler algorithm was also developed [59]. Unlike TP, the weights predicted by CQMOM are always non-negative since the nodes/abscissas can move in the velocity space without rotating the reference frame, but the moment transport equation system is only weakly hyperbolic, which implies it may produce local delta-shocks. The multi-Gaussian extended quadrature method of moments (EQMOM) was developed and applied to kinetic equations in the context of large-eddy simulations [5] and applied also in other contexts [7, 52]. Since EQMOM is based on a continuous reconstruction of the NDF it can predict both the large-scale PTC (primary nodes) and the small-scale dispersion (secondary nodes). Meanwhile, it was proven that EQMOM is strictly hyperbolic [5, 26]. On the other hand, the anisotropic Gaussian (AG) method was also developed and it seeks the best compromise in terms of number of moments versus reproduction of the physics of the particulate flow [52]. Due to the hyperbolic nature of AG, Godunov solvers can be employed to obtain stable and realizable results. It was also proven that AG can be used to predict cluster-induced turbulence in gas-particle flows [30, 28]. The only drawback of AG is that it cannot be used to predict large-scale PTC.

Recently, the maximum-entropy approach was coupled with QMOM. In this manner, the discrete velocity abscissas/nodes can be reconstructed and the maximum-entropy principle can be also ensured [4]. Apart from that, Kong and Fox developed a size-conditioned algorithm for generalized population balance equations which can be used to describe poly-disperse gas-particle systems [29]. Meanwhile, since the realizability of the moments is very critical, many numerical techniques were developed to ensure realizability when high-order spatial discretization schemes are employed. Readers are suggested to other works for a thorough discussion [53, 44, 51, 31].

In the work of Fox et al. [19], the hyperbolic quadrature method of moments (HyQMOM) was developed for kinetic equations. Similar with the fourth-order MG method previously mentioned, the aim of HyQMOM is to achieve hyperbolicity. Compared with CQMOM, HyQMOM does not generate

non-physical delta shocks, which refer to rapid accumulation of particles when multiple PTC occurs. This is because HyQMOM fixes the last central moment to calculate the weights and nodes/abscissas to ensure the transported moment system to be hyperbolic. Moreover, for two-node and three-node HyQMOM, the weights and nodes/abscissas can be found analytically. Compared with MG, it is more computationally affordable: only five moments and ten moments are necessary for one- or two-dimensional test cases. Eventually, it should be noted here that for multi-dimensional problems a modified CQMOM was combined with HyQMOM, resulting in the so-called conditional hyperbolic quadrature method of moments (CHyQMOM).

To the best of our knowledge, after CHyQMOM was developed [19, 46], it was not further investigated for the simulation of gas-particle flows nor implemented in a popular CFD package. The goal of the current study is to address this gap by implementing CHyQMOM in OpenFOAM-7 and by investigating its accuracy for a number of gas-particle flows. This follows our previous works in which QBMM were implemented in OpenFOAM and employed to investigate liquid-liquid and gas-liquid flows [33, 35, 34, 32] as well as solid-liquid flows [2, 14, 13, 1, 3]. Several test cases are explored here by including gas-particle flows, characterised by one-way and two-way coupling. Results are compared against analytical solutions, when available, or against predictions obtained with Lagrangian particle tracking. Advantages and disadvantages of CHyQMOM for gas-particle flows are discussed together with possible numerical improvements.

2. Basic theory and governing equations

In this section, we revisit the theory behind QMOM, HyQMOM and CHyQMOM, which was originally developed in previous works [43, 19] and we discuss the governing equations pertaining both the gas and solid particulate phases.

2.1. Gas-phase governing equations

As it is well-known the gas-phase governing equations, for a gas-solid multiphase system, are the continuity and momentum balance equations, which read as follows:

$$\frac{\partial \alpha_{\mathrm{c}}}{\partial t} + \nabla \cdot (\alpha_{\mathrm{c}} \mathbf{U}_{\mathrm{c}}) = 0, \tag{1}$$

and:

$$\frac{\partial \alpha_{\mathrm{c}} \mathbf{U}_{\mathrm{c}}}{\partial t} + \nabla \cdot (\alpha_{\mathrm{c}} \mathbf{U}_{\mathrm{c}} \mathbf{U}_{\mathrm{c}}) = \alpha_{\mathrm{c}} \left(-\nabla p + \nabla \cdot \boldsymbol{\tau}_{\mathrm{c}} + \mathbf{g}\right) - \mathbf{M}_{\mathrm{drag}}, \tag{2}$$

where the operator ∇ implicitly refers to the spatial coordinates: $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)$, α_{c} is the gas-phase volume fraction, which sums to unity with the solid-phase volume fraction α_{d}, $\mathbf{U}_{\mathrm{c}}$ is average

gas velocity, p is the average gas pressure, ρ_c is the gas density, $\mathbf{g}$ is the gravity acceleration vector and $\mathbf{M}_\mathrm{drag}$ represents the momentum exchange between the gas phase and the particle phase. In this work, only the drag force is included. The extension to other gas-particle forces, although not discussed here, is straightforward. $\boldsymbol{\tau}_\mathrm{c}$ is the sum of the viscous and turbulent stress tensors, which can be calculated by:

$$\boldsymbol{\tau}_\mathrm{c} = \mu_\mathrm{eff}\left[\nabla\mathbf{U}_\mathrm{c} + (\nabla\mathbf{U}_\mathrm{c})^\mathrm{T}\right] - \frac{2}{3}\left[\mu_\mathrm{eff}\left(\nabla\cdot\mathbf{U}_\mathrm{c}\right) + \rho k\right]\mathbf{I}, \tag{3}$$

where: $\mu_\mathrm{eff} = \mu_c + \mu_t$ is the effective gas viscosity, obtained by the summation of the molecular and turbulent gas-phase viscosity and k is the gas-phase turbulent kinetic energy. As well-known the gas-phase turbulent viscosity is calculated from the turbulent kinetic energy, k, and the turbulent energy dissipation rate, ε, by employing a turbulence model. The drag force can be instead calculated as follows:

$$\mathbf{M}_\mathrm{drag} = \frac{3}{4}\frac{\alpha_\mathrm{d} C_\mathrm{D}}{d}\left|\mathbf{U}_\mathrm{c} - \mathbf{U}_\mathrm{d}\right|\left(\mathbf{U}_\mathrm{c} - \mathbf{U}_\mathrm{d}\right), \tag{4}$$

where C_D is the drag coefficient, which is generally calculated from a correlation, such as for example the Wen & Yu model [55], d is the solid particle diameter, whereas $\mathbf{U}_\mathrm{d}$ is the average velocity for the solid particles.

2.2. Solid-phase governing equations

As well-known in gas-solid multiphase flows, solid particles are characterized via a NDF defined so that:

$$f(t,\mathbf{x},\mathbf{v})\mathrm{d}\mathbf{v}$$

represents the expected number of particles per unit volume with velocity between $\mathbf{v}$ and $\mathbf{v}+\mathrm{d}\mathbf{v}$. The average solid particle velocity can therefore be calculated as follows:

$$\mathbf{U}_\mathrm{d} = \frac{\iiint f(t,\mathbf{x},\mathbf{v})\mathbf{v}\mathrm{d}\mathbf{v}}{\iiint f(t,\mathbf{x},\mathbf{v})\mathrm{d}\mathbf{v}}, \tag{5}$$

whereas the solid particle volume fraction as follows:

$$\alpha_\mathrm{d} = V_\mathrm{d}\iiint f(t,\mathbf{x},\mathbf{v})\mathrm{d}\mathbf{v}, \tag{6}$$

where $V_\mathrm{d} = \frac{\pi}{6}d^3$ is the volume of one single particle, as the population is assumed to be monodisperse (i.e. all particles have the same size, volume and mass). The evolution of the NDF is dictated by the Boltzmann equation:

$$\frac{\partial f}{\partial t} + \nabla\cdot(\mathbf{v}f) + \nabla_\mathbf{v}\cdot(\mathbf{A}f) = \mathbb{C}, \tag{7}$$

where $\nabla_{\mathbf{v}} = (\partial/\partial u, \partial/\partial v, \partial/\partial w)$ and u, v and w are the three components of the particle velocity vector $\mathbf{v}$, $\mathbf{A}$ is the particle acceleration and $\mathbb{C}$ is the particle collision integral.

If only the drag force and the gravity force are considered the acceleration term reads as follows:

$$\mathbf{A} = \frac{1}{\tau_{\mathrm{p}}}(\mathbf{v} - \mathbf{U}_{\mathrm{c}}) + \mathbf{g}, \tag{8}$$

where $\mathbf{v}$ is the particle velocity vector, $\mathbf{U}_{\mathrm{c}}$ is the gas-phase velocity, τ_{p} is the particle relaxation time and $\mathbf{g}$ is the gravity acceleration.

We employ the Bhatnagar-Gross-Krook (BGK) approximation to describe the collision term:

$$\mathbb{C} = \frac{1}{\tau_c}(f_{\mathrm{eq}} - f), \tag{9}$$

where f_{eq} is the equilibrium NDF.

As mentioned, Eq. (7) is often solved in terms of the moments of the NDF and the resulting closure problem is overcome with a quadrature approximation. In the following we discuss this closure by considering first a one-dimensional problem and then a two-dimensional problem, with u and v indicating the components of the particle velocity vector $\mathbf{v}$, with x and y representing the corresponding spatial directions.

2.2.1. Univariate QMOM and HyQMOM

Let us consider a univariate NDF, $f(u)$, where spatial and time dependecies are omitted. The k^{th} moment can be defined as follows:

$$m_k = \int f(u) u^k \mathrm{d}u. \tag{10}$$

For non-vacuum region ($m_0 > 0$), the central moment C_k is defined as follows:

$$C_k = \frac{1}{m_0} \int f(u) \left(u - \frac{m_1}{m_0} \right)^k \mathrm{d}u, \tag{11}$$

where m_1/m_0 is the average particle velocity along the x-direction. By definition the following relationships are valid:

$$\begin{aligned}
C_0 &= 1, \\
C_1 &= 0, \\
C_2 &= m_2 - \left(\frac{m_1}{m_0}\right)^2, \\
C_3 &= m_3 - 3m_2\left(\frac{m_1}{m_0}\right) + 2\left(\frac{m_1}{m_0}\right)^3, \\
C_4 &= m_4 - 4m_3\left(\frac{m_1}{m_0}\right) + 6\left(\frac{m_1}{m_0}\right)^2 - 3\left(\frac{m_1}{m_0}\right)^4.
\end{aligned} \tag{12}$$

While C_0 is always equal to unity and C_1 is always null, C_2 represents the non-negative normalized variance, C_3 is the skewness and C_4 is the kurtosis. The moments are strictly realizable if and only if $C_2 > 0$ and $C_4 \geqslant C_2^2 + C_3^2/C_2$. When using both QMOM or HyQMOM the NDF is implicitly reconstructed as follows:

$$f(u) = \sum_{\alpha=1}^{N} w_\alpha \delta\left[u - \left(u_\alpha + \frac{m_1}{m_0}\right)\right], \tag{13}$$

where w_α are the non-negative weights, u_α are the scaled velocity nodes, or abscissas, in the following referred to as nodes, and N is the number of nodes. Substituting Eq. (13) into Eq. (11) leads to:

$$C_k = \frac{1}{m_0}\sum_{\alpha=1}^{N} w_\alpha u_\alpha^k. \tag{14}$$

It can be also seen that the moments can be calculated as follows:

$$m_k = \sum_{\alpha=1}^{N} w_\alpha \left(u_\alpha + \frac{m_1}{m_0}\right)^k. \tag{15}$$

In QMOM, for any system of $2N$ moments, the central moments can be found and the nodes and weights can be calculated by the moment inversion procedure. However, it was shown that the moment system of QMOM is weakly hyperbolic [6]; hyperbolicity is ensured in HyQMOM by fixing the last central moment C_{2N-1} (and consequently also the moment m_{2N-1}) [19]. If three nodes are used ($N = 3$) with QMOM six moments are needed to evaluate the three nodes and weights, whereas with HyQMOM the last central moment C_5 is fixed as follows:

$$C_5 = \frac{C_3}{C_2^2}(2C_2C_4 - C_3^2), \tag{16}$$

and therefore only 5 moments are needed, resulting in the following analytical expressions:

$$
\begin{aligned}
u_1 &= \frac{1}{2}\sqrt{C_2}\left(q - \sqrt{4\eta - 3q^2}\right), \\
u_2 &= 0, \\
u_3 &= \frac{1}{2}\sqrt{C_2}\left(q + \sqrt{4\eta - 3q^2}\right), \\
w_1 &= \frac{m_0}{2(\eta - q^2)}\left(1 + \frac{q}{\sqrt{4\eta - 3q^2}}\right), \\
w_2 &= m_0\left(1 - \frac{1}{\eta - q^2}\right), \\
w_3 &= \frac{m_0}{2(\eta - q^2)}\left(1 - \frac{q}{\sqrt{4\eta - 3q^2}}\right),
\end{aligned}
\tag{17}
$$

where

$$
\eta = \frac{C_4}{C_2^2}, \quad q = \frac{C_3}{C_2^{3/2}}. \tag{18}
$$

As long as $C_2 > 0$ and $C_4 \geqslant C_2^2 + C_3^2/C_2$, we have that $\eta \geqslant 1 + q^2$ and all the weights reported in Eq. (17) are non-negative. Besides hyperbolicity HyQMOM has the additional advantage, with respect to QMOM, of transporting less moments, however the second node is always null, potentially causing numerical issues as explained below.

2.2.2. Bivariate CQMOM and CHyQMOM

QMOM and HyQMOM when applied to multidimensional problems need to be combined with the concept of conditional probability density function (PDF) and conditional moment, resulting in CQMOM and CHyQMOM. The derivation of the governing equations can be found in the cited literature, whereas a detailed recap for a bivariate problem can be found in Appendix A. In summary for a bivariate problem the moment of the NDF of order k with respect to u and of order l with respect to v is defined as follows:

$$
m_{k,l} = \iint f(u,v) u^k v^l \,\mathrm{d}u\,\mathrm{d}v. \tag{19}
$$

In CQMOM and CHyQMOM the NDF is written as the product of a univariate NDF and a conditional PDF:

$$
f(u,v) = f(u) f(v|u), \tag{20}
$$

where $f(u)$ is the already introduced univariate NDF and $f(v|u)$ is the conditional PDF for v when u is equal to a given value. It should be noted here that the NDF can also be written as follows:

$$f(u,v) = f(v)f(u|v) \tag{21}$$

resulting into another permutation.

With both CQMOM and CHyQMOM the NDF is reconstructed as follows:

$$f(u,v) = \sum_{\alpha=1}^{N_1}\sum_{\beta=1}^{N_2} w_\alpha w_{\alpha,\beta}\delta\left[u - \left(u_\alpha + \frac{m_{1,0}}{m_{0,0}}\right)\right]\delta\left[v - \left(v_{\alpha,\beta} + \frac{m_{0,1}}{m_{0,0}}\right)\right], \tag{22}$$

when the first permutation is employed, namely: $f(u,v) = f(u)f(v|u)$. The quadrature nodes u_α, with $\alpha = 1,\ldots,N_1$, are called primary nodes, whereas $v_{\alpha,\beta}$, with $\beta = 1,\ldots,N_2$, are called secondary nodes.

On the contrary, if the second permutation is employed: $f(u,v) = f(v)f(u|v)$, the following reconstruction is used:

$$f(u,v) = \sum_{\alpha=1}^{N_1}\sum_{\beta=1}^{N_2} w_\alpha w_{\alpha,\beta}\delta\left[u - \left(u_{\alpha,\beta} + \frac{m_{1,0}}{m_{0,0}}\right)\right]\delta\left[v - \left(v_\alpha + \frac{m_{0,1}}{m_{0,0}}\right)\right], \tag{23}$$

where v_α are now primary nodes and $u_{\alpha,\beta}$ are secondary nodes.

In CQMOM with the first permutation the primary nodes, u_α, and weights, w_α, are calculated from the pure moments, resulting in (for $N_1 = 3$):

$$m_{0,0}, m_{1,0}, m_{2,0}, m_{3,0}, m_{4,0}, m_{5,0} \rightarrow \begin{pmatrix} w_1 \\ u_1 \end{pmatrix}, \begin{pmatrix} w_2 \\ u_2 \end{pmatrix}, \begin{pmatrix} w_3 \\ u_3 \end{pmatrix}.$$

Secondary nodes $v_{\alpha,\beta}$ and weights $w_{\alpha,\beta}$ can instead be calculated from the mixed moments, resulting in (for $N_2 = 3$):

$$\begin{matrix} m_{2,0} & m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} \\ m_{1,0} & m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} \\ m_{0,0} & m_{0,1} & m_{0,2} & m_{0,3} & m_{0,4} & m_{0,5} \end{matrix} \rightarrow \begin{pmatrix} w_{1,1} \\ v_{1,1} \\ w_{1,2} \\ v_{1,2} \\ w_{1,3} \\ v_{1,3} \end{pmatrix}, \begin{pmatrix} w_{2,1} \\ v_{2,1} \\ w_{2,2} \\ v_{2,2} \\ w_{2,3} \\ v_{2,3} \end{pmatrix}, \begin{pmatrix} w_{3,1} \\ v_{3,1} \\ w_{3,2} \\ v_{3,2} \\ w_{3,3} \\ v_{3,3} \end{pmatrix},$$

and analogously when one makes use of the second permutation. A similar procedure is used when

CHyQMOM is employed. A sketch of the positions of the nodes of the reconstructed NDF in the (u, v)-plane is reported in Fig. 1, for the first and second permutation. As it can be seen their positions depend on the permutation, a feature which needs specific techniques, while dealing with practical problems characterized by two-way coupling.

Equation (7) needs to be transformed into a set of transport equations for the moments of the NDF and to be solved under the finite-volume framework. This is usually done by using the operator splitting procedure described below.

2.2.3. Operator splitting procedure

The operator splitting procedure [59] consists in the following steps: firstly the moments transport equations are updated only considering physical advection, without considering the other source terms. Then the acceleration term is considered, the moments reconstructed and at last the collision flux is accounted for, resulting in the final moments values. The operator splitting procedure was extensively employed in other related works [59, 45, 30].

Spatial fluxes evaluation. Let us start with the first step illustrated below for a two-dimensional problem. For three-node CHyQMOM, by taking the moments of Eq. (7), we consider the following ten moments transport equations:

$$
\begin{aligned}
&\frac{\partial m_{0,0}}{\partial t} + \frac{\partial m_{1,0}}{\partial x} + \frac{\partial m_{0,1}}{\partial y} = 0,\\
&\frac{\partial m_{1,0}}{\partial t} + \frac{\partial m_{2,0}}{\partial x} + \frac{\partial m_{1,1}}{\partial y} = 0,\\
&\frac{\partial m_{2,0}}{\partial t} + \frac{\partial m_{3,0}}{\partial x} + \frac{\partial m_{2,1}}{\partial y} = 0,\\
&\frac{\partial m_{3,0}}{\partial t} + \frac{\partial m_{4,0}}{\partial x} + \frac{\partial m_{3,1}}{\partial y} = 0,\\
&\frac{\partial m_{4,0}}{\partial t} + \frac{\partial m_{5,0}}{\partial x} + \frac{\partial m_{4,1}}{\partial y} = 0,\\
&\frac{\partial m_{0,1}}{\partial t} + \frac{\partial m_{1,1}}{\partial x} + \frac{\partial m_{0,2}}{\partial y} = 0,\\
&\frac{\partial m_{0,2}}{\partial t} + \frac{\partial m_{1,2}}{\partial x} + \frac{\partial m_{0,3}}{\partial y} = 0,\\
&\frac{\partial m_{0,3}}{\partial t} + \frac{\partial m_{1,3}}{\partial x} + \frac{\partial m_{0,4}}{\partial y} = 0,\\
&\frac{\partial m_{0,4}}{\partial t} + \frac{\partial m_{1,4}}{\partial x} + \frac{\partial m_{0,5}}{\partial y} = 0,\\
&\frac{\partial m_{1,1}}{\partial t} + \frac{\partial m_{2,1}}{\partial x} + \frac{\partial m_{1,2}}{\partial y} = 0.
\end{aligned}
\tag{24}
$$

This is the stage of flux evaluation, in which only advection is included. This system is unclosed due to the unknown third-order moments $m_{2,1}, m_{1,2}$, fourth-order moments $m_{3,1}, m_{1,3}$ and fifth-order

moments $m_{5,0}, m_{4,1}, m_{1,4}$. In QBMM, these unknown moments can be reconstructed by using the nodes and weights as reported in Eq. (A.4).

The general solution for $m_{k,l}$ can be obtained once the fluxes at the cell interfaces are evaluated. The flux should ensure that the updated moments are realizable. In this work, we implemented the higher-order realizable scheme developed by Vikas et al. [53]. The implementation procedure in OpenFOAM is omitted here and readers are suggested to consult our previous work for more information [35, 37, 36].

It should be noted here that when the modified CQMOM is used for spatial transport care must be taken when choosing the permutation. Such issue does not exist in our previous work in which a size-dependent velocity model was used [37, 36]. For example, for a two-dimensional velocity space, there are two permutations of CQMOM, but only one of them give the correct spatial flux in each direction. In order to calculate the correct flux component in the x direction (e.g., $\frac{\partial m_{1,0}}{\partial x}$), u should be selected as the primary node and v as the secondary node. Likewise, v should be the primary node and u should be the secondary node for the flux calculation in y direction (e.g., $\frac{\partial m_{0,1}}{\partial y}$).

In summary, the first step of operator splitting follows this procedure:

1. Given the initial ten moments, calculate the primary nodes u_α, secondary nodes $v_{\alpha,\beta}$ and corresponding weights with the first permutation.
2. Calculate the primary nodes v_α, secondary nodes $u_{\alpha,\beta}$ and corresponding weights with the second permutation.
3. Interpolate the weights defined at cell centers to cell faces by using a high-order scheme. Interpolate the nodes to cell faces by using a first-order scheme [53].
4. Use $u_\alpha/v_{\alpha,\beta}$ and weights defined at cell faces to update fluxes in x direction, use $v_\alpha/u_{\alpha,\beta}$ and weights to update fluxes in y direction.
5. Evolve Eq. (24) by using an explicit time scheme and obtain the final moments.

Acceleration evaluation. After the moments are advected, their values should be updated to account for the acceleration and collision terms. In the work of Patel et al. [46], the acceleration terms are transformed to mixed moments and central moments. Such procedure can only handle simple cases. In order to account for complex drag terms, a supplementary quadrature step is envisaged. Since the acceleration has no effects on the weights, one can work directly with the quadrature constructing an ordinary differential equation for each node:

$$\begin{aligned} \frac{\mathrm{d}u_\alpha}{\mathrm{d}t} &= -\frac{u_\alpha - U_c}{\tau_\mathrm{p}} + g_x, \quad \frac{\mathrm{d}v_{\alpha,\beta}}{\mathrm{d}t} = -\frac{v_{\alpha,\beta} - V_c}{\tau_\mathrm{p}} + g_y, \\ \frac{\mathrm{d}v_\alpha}{\mathrm{d}t} &= -\frac{v_\alpha - V_c}{\tau_\mathrm{p}} + g_y, \quad \frac{\mathrm{d}u_{\alpha,\beta}}{\mathrm{d}t} = -\frac{u_{\alpha,\beta} - U_c}{\tau_\mathrm{p}} + g_x, \end{aligned} \tag{25}$$

where τ_{p} is the already introduced particle relaxation time and U_c and V_c are the x and y components of the gas velocity.

For the two-dimensional three-node CHyQMOM 24 ODEs need to be solved. After these nodes are updated by selecting a suitable time integration scheme, the final moments can be reconstructed as reported in Eq. (A.4). Such procedure is extensively discussed in other similar works [5, 7, 15, 45]. However, we need to stress here that care must be also taken since different moments permutations produce different reconstructed velocities. In order to accommodate different permutations, the moments updating procedure discussed above is modified as follows:

$$m_{k,l}^{*} \rightarrow \begin{cases} \xrightarrow{u_\alpha, v_{\alpha,\beta}} m_{k,l}^{(1),**} \\ \xrightarrow{v_\alpha, u_{\alpha,\beta}} m_{k,l}^{(2),**} \end{cases} \rightarrow m_{k,l}^{**} = \frac{1}{2}\left(m_{k,l}^{(1),**} + m_{k,l}^{(2),**}\right), \tag{26}$$

where $m_{k,l}$ can represent any order moment. Only in this manner, the effect of permutation can be addressed.

Collision flux evaluation. Although in this work we consider dilute gas-particle flows, it is still necessary to include the collision flux to prevent local particle accumulation and relevant results will be discussed in later sections. We employ the Bhatnagar-Gross-Krook (BGK) approximation to describe the collision term:

$$\mathbb{C} = \frac{1}{\tau_c}(f_{\mathrm{eq}}(u,v) - f(u,v)), \tag{27}$$

where f_{eq} is the equilibrium distribution function, which can be written as the following multivariate Gaussian distribution:

$$f_{\mathrm{eq}}(u,v) = \frac{m_{0,0}}{\sqrt{|2\pi\boldsymbol{\lambda}|}} \exp\left[-\frac{1}{2}\left(\begin{pmatrix} u \\ v \end{pmatrix} - \begin{pmatrix} \frac{m_{1,0}}{m_{0,0}} \\ \frac{m_{0,1}}{m_{0,0}} \end{pmatrix}\right)^{\mathrm{T}} \cdot \boldsymbol{\lambda}^{-1} \cdot \left(\begin{pmatrix} u \\ v \end{pmatrix} - \begin{pmatrix} \frac{m_{1,0}}{m_{0,0}} \\ \frac{m_{0,1}}{m_{0,0}} \end{pmatrix}\right)\right], \tag{28}$$

where $\boldsymbol{\lambda}$ is defined by

$$\boldsymbol{\lambda} = \gamma\omega^2\Theta\mathbf{I} + (\gamma\omega^2 - 2\gamma\omega + 1)\boldsymbol{\sigma}; \tag{29}$$

$\boldsymbol{\sigma}$ is the velocity covariance matrix which is defined as follows [17]:

$$\boldsymbol{\sigma} = \begin{pmatrix} \frac{m_{2,0}}{m_{0,0}} - u^2, & \frac{m_{1,1}}{m_{0,0}} - uv \\ \frac{m_{1,1}}{m_{0,0}} - uv, & \frac{m_{2,0}}{m_{0,0}} - v^2 \end{pmatrix}; \tag{30}$$

Θ is the granular temperature defined by $\Theta = 1/2(\boldsymbol{\sigma}_{11} + \boldsymbol{\sigma}_{12})$, $\gamma = 1/\mathrm{Pr}$, $\omega = (1+e)/2$, e is the restitution coefficient, Pr is the Prandtl number. In this work, we degenerate the multivariate equilibrium Gaussian distribution to the Maxwellian distribution by setting $\mathrm{Pr} = 1$ and $e = 1$. Thus,

$\boldsymbol{\lambda}$ can be simplified to

$$\boldsymbol{\lambda} = \Theta \mathbf{I}. \tag{31}$$

Once $\boldsymbol{\lambda}$ is calculated, it can be used to calculate the equilibrium moments:

$$m_{0,0}^{\mathrm{eq}} = m_{0,0}, \tag{32}$$

$$m_{1,0}^{\mathrm{eq}} = m_{0,0}u, m_{0,1}^{\mathrm{eq}} = m_{0,0}v, \tag{33}$$

$$m_{2,0}^{\mathrm{eq}} = m_{0,0}(\boldsymbol{\lambda}_{11} + u^2), m_{0,2}^{\mathrm{eq}} = m_{0,0}(\boldsymbol{\lambda}_{22} + v^2), \tag{34}$$

$$m_{3,0}^{\mathrm{eq}} = m_{0,0}(3\boldsymbol{\lambda}_{11}u + u^3), m_{0,3}^{\mathrm{eq}} = m_{0,0}(3\boldsymbol{\lambda}_{22}v + v^3), \tag{35}$$

$$m_{4,0}^{\mathrm{eq}} = m_{0,0}(6\boldsymbol{\lambda}_{11}u^2 + 3\boldsymbol{\lambda}_{11}^2 + u^4), m_{4,0}^{\mathrm{eq}} = m_{0,0}(6\boldsymbol{\lambda}_{22}v^2 + 3\boldsymbol{\lambda}_{22}^2 + v^4), \tag{36}$$

$$m_{1,1}^{\mathrm{eq}} = m_{0,0}(\boldsymbol{\lambda}_{12} + uv) = m_{0,0}uv. \tag{37}$$

The final moments m^{***} can be updated by solving the following ODEs:

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \frac{1}{\tau_c}(m^{\mathrm{eq}} - m^{**}). \tag{38}$$

2.2.4. Two-way coupling

Under the two-way coupling regime attention must also be paid to the evolution of the continuous gas phase. In this work, the procedure discussed in our previous work which was used to simulate monokinetic gas-liquid flows is employed [37]. For CHyQMOM the key point is how to handle the two-way coupling for different permutations, which will be discussed in this section. We recall that the particle phase fraction is: $\alpha_{\mathrm{d}} = V_{\mathrm{d}} m_{0,0}$. In order to improve the numerical stability, Eq. (4) is reformulated as follows:

$$\mathbf{M}_{\mathrm{drag}} = \frac{3}{4}\alpha_{\mathrm{d}} C_{\mathrm{D}} \mathrm{Re} \frac{\mu_{\mathrm{c}}}{d^2} (\mathbf{U}_{\mathrm{c}} - \mathbf{U}_{\mathrm{d}}) = \frac{\rho_{\mathrm{d}}\alpha_{\mathrm{d}}}{\frac{4}{3}\frac{d^2\rho_{\mathrm{d}}}{C_{\mathrm{D}}\mathrm{Re}\mu_{\mathrm{c}}}} (\mathbf{U}_{\mathrm{c}} - \mathbf{U}_{\mathrm{d}}) = \rho_{\mathrm{d}}\alpha_{\mathrm{d}} \frac{1}{\tau_{\mathrm{p}}} (\mathbf{U}_{\mathrm{c}} - \mathbf{U}_{\mathrm{d}}), \tag{39}$$

where again $\tau_{\mathrm{p}} = \frac{4}{3}\frac{d^2\rho_{\mathrm{d}}}{C_{\mathrm{D}}\mathrm{Re}\mu_{\mathrm{c}}}$ is the particle relaxation time. As it can be seen later in the test cases, it can be set as a fixed value or calculated from a specific drag model.

In typical coupling procedures, the relaxation time and particle phase velocity should be replaced by discretized values, namely:

$$\mathbf{M}_{\mathrm{drag}} = \rho_{\mathrm{d}}\mathbf{U}_{\mathrm{c}} \sum \frac{\alpha_{\mathrm{d}}^{i,j}}{\tau_{\mathrm{p}}^{i,j}} - \rho_{\mathrm{d}} \sum \frac{\alpha_{\mathrm{d}}^{i,j}}{\tau_{\mathrm{p}}^{i,j}} \mathbf{U}_{\mathrm{d}}^{i,j} \tag{40}$$

where

$$\tau_{\mathrm{p}}^{i,j} = \frac{4}{3}\frac{d^2\rho_{\mathrm{d}}}{C_{\mathrm{D}}^{i,j}\mathrm{Re}^{i,j}\mu_{\mathrm{c}}}, \mathrm{Re}^{i,j} = \frac{d\rho_{\mathrm{c}}|\mathbf{U}_{\mathrm{c}} - \mathbf{U}_{\mathrm{d}}^{i,j}|}{\mu_{\mathrm{c}}}. \tag{41}$$

Equation 42 can be treated either implicitly or explicitly in Eq. (2). This procedure is suitable for TP QMOM or GPBE solvers described in our previous works [37]. However, it introduces problems for CQMOM, since the reconstructed $\mathbf{U}_{\mathrm{d}}$ is related to permutations. To handle such issue, the key point is that the total momentum should be conserved. Therefore, the momentum transfer to the air phase should be identical with that exerted on the particle phase. The momentum density components in the x and y directions are equal to $m_{1,0}$ and $m_{0,1}$. The momentum change on x and y directions can be defined by:

$$\begin{aligned}
\mathbf{M}_{\mathrm{drag},x} &= \alpha_{\mathrm{d}}\rho_{\mathrm{d}}\frac{\partial(V_{\mathrm{d}}m_{1,0})}{\partial t} \rightarrow \mathbf{M}_{\mathrm{drag},x} = V_{\mathrm{d}}^2\rho_{\mathrm{d}}m_{0,0}\frac{m_{1,0}^{**} - m_{1,0}^{*}}{\Delta t}, \\
\mathbf{M}_{\mathrm{drag},y} &= \alpha_{\mathrm{d}}\rho_{\mathrm{d}}\frac{\partial(V_{\mathrm{d}}m_{0,1})}{\partial t} \rightarrow \mathbf{M}_{\mathrm{drag},y} = V_{\mathrm{d}}^2\rho_{\mathrm{d}}m_{0,0}\frac{m_{0,1}^{**} - m_{0,1}^{*}}{\Delta t},
\end{aligned} \tag{42}$$

where m^{*} are the moments after convection, m^{**} are the moments after acceleration (see Eq. (26)). In this manner, momentum transfer considers the permutation effects. It should be noted here that in Eq. (25), the effect of gravity is taken into account and one could get rid of the momentum transfer due to gravity in particle phase while using Eq. (42). Meanwhile, it can be seen that the momentum exchange term can be only treated explicitly, which requires smaller time step.

Substituting Eq. (42) into Eq. (2) leads to the predicted gas phase velocity, which should be further corrected to obey the continuity constraint. This is achieved by solving the following pressure Poission equation:

$$\nabla\cdot\left(\frac{\alpha_{\mathrm{c}}^2}{a_P}\nabla\left(\frac{p_{\mathrm{c}}}{\rho_{\mathrm{c}}}\right)\right) = \frac{\partial\alpha_{\mathrm{c}}}{\partial t} + \nabla\cdot\alpha_{\mathrm{c}}\left(\mathbf{HbyA} - \frac{1}{a_P}\mathbf{M}_{\mathrm{drag}}\right). \tag{43}$$

where **HbyA** is the predicted gas phase velocity, a_P is the matrix diagonal coefficients field. Readers are suggested to our previous work for a more comprehensive discussion of the derivation [37]. In summary, the two-way coupling procedure follows these steps:

1. given the initial boundary conditions, update particle phase variables by moments transport equations;
2. feed the particle phase fraction and velocities into the gas phase equation system, by using the iterative procedure to obtain divergence-free gas velocity and pressure;
3. move to next time step and repeat from step 1.

A chart flow of the whole two-way coupling procedure is reported in Fig. 2.

3. Test cases and discussions

In this section we describe the different test cases employed for the verification of our CHyQMOM implementation in OpenFOAM-7 and for its validation. First the implementation is verified on a simple zero-dimensional problem and the discussion is reported in Appendix B. Then the implementation is verified for one-dimensional problems, as well as for more complex and realistic test cases.

3.1. Phase segregation problem

In this section, we use HyQMOM to simulate the movement of particles in a one-dimensional (1D) problem, namely along the x direction as reported in Fig. 3. Similar test cases were also investigated in our previous work [37] in which `oneWayGPBEFoam` was employed. The difference is that in our previous work, the particles were assumed to be poly-disperse in size. The velocity was assumed to be a function of particle size and a velocity polynomial function was employed to close the system of governing equations (i.e. mono-kinetic assumption). In this work, the particles are assumed to be mono-disperse in size, but poly-disperse in velocity, namely the internal coordinate is now particle velocity (i.e. poly-kinetic system). The drag force is not considered at this stage. Therefore, the particles should move forward with different velocities and phase segregation finally occurs. In order to verify the capability of three node HyQMOM, three test cases with different initializations are considered. Specifically, in the first test case, the initial moments are calculated from a mono-disperse velocity distribution (i.e. all particles have the same velocity). In the second test case, the initial moments are calculated from a bi-disperse velocity distribution (i.e. particles are characterized by two velocity values). In the last test case, the initial moments are calculated from a tri-disperse velocity distribution. Therefore, with an initial condition that has all the particles in the same position, at the final time instant of the simulation, the first test case should end up with all the particles in the same position, the second one ends up with the particles divided in two locations, whereas in the last one the particles end up in three different locations. It should be noted here that the first and second test cases correspond to degenerate test cases. Three-node HyQMOM should be able to handle such problems without numerical difficulties.

This 1D problem is handled by 600 computational cells and cyclic boundary conditions, implying that after the particles flow out from the right boundary, they can flow back in from the left boundary. The time step is fixed as 0.0001 s so that the moments stay realizable. Simulation results for the values of m_0 predicted by HyQMOM with three nodes are reported in Fig. 4. The analytical solutions can be simply calculated and are not shown for brevity. It can be seen that three-node HyQMOM succeeds in predicting the test case initialized with mono- and bi-disperse velocity distributions, implying that it can successfully handle degenerate test cases. However, it fails in predicting the tri-disperse velocity

population. This can be explained by the theory of HyQMOM. As discussed in the previous section, the last central moment of the moment set is fixed equal to zero, instead of computing it from the moments. This means that $u_2 = 0$, a value which cannot be adjusted as in QMOM.

At last, it should also be noted that the three-node HyQMOM predicts the degenerated test cases very well. In the original QMOM, the degenerated test cases cannot be predicted since the singular matrix cannot be handled by the moments inversion algorithm. In HyQMOM, the degenerating test cases are automatically transformed to one node or two node QMOM. Meanwhile, it can be seen that the degenerating algorithm must be implemented in CFD codes.

3.2. Particle trajectory crossing problem

In this test case the particle crossing problem is simulated with three-node HyQMOM in the three different versions depicted in Fig. 5. In the first test case, one group of particles is still, whereas the other one is moving towards the right. The two groups of particles will thus eventually cross over each other as time advances and particle crossing occurs since the particles initialized at the left would move through the quiescent particles to the right. In these examples particles do not collide and therefore do not exchange momentum. In the second test case, particles are initialized at different positions with opposite velocities. These particles with opposite velocities will cross each other as well leading into opposite directions. In the last test case, three groups of particle are initialized with one of them fixed in the middle. The other groups of particles are initialized at the right and left with opposite velocities. As time advances the particles on the right and left will cross each other as well as the particles in the middle.

Simulation settings are identical with the test cases described in the previous section. However, we stop the simulation before the particles reached the wall to prevent boundary's effects on the results. Figure 6 illustrates the resulting time evolution for m_0. It can be seen that for the first test case, the group of particles on the left successfully cross through the quiescent particles and moves to the right. For the second test case, these two groups of particles cross each other and move to the left and right, respectively. For the third test case, the left group of particles goes to the right and the right group of particles goes to the left, they cross each other, however, at the end, HyQMOM cannot predict the movement of the three groups of particles. This is due again to the fact that the last central moment is fixed in HyQMOM resulting in a null velocity node.

3.3. Wall rebounding problem

In this section, we simulate wall rebounding problems. Similarly to the phase segregation problem, we initialize the population of particles with mono-, bi- and tri-modal velocity distributions. The restitution coefficient e is assumed to be equal to one, which corresponds to elastic rebounding.

Therefore, after these particles impinge the wall, they should move towards the opposite direction with opposite velocity vector and identical magnitude. For the first test case, a mono-disperse velocity distribution is considered. All the particles move to the left and rebound to the right with identical velocity magnitude, as reported in Fig. 7.

For the second test case, two different velocities values are assigned to the particles at the beginning and the particles move to the left with different velocities. In the vicinity of the wall, phase segregation occurs and two separated groups of particles emerge, which in turn are rebounded after impinging into wall. It is worth mentioning here that for both of these test cases, the particle velocity distribution is bi-modal near the impingement point, where particles moving to the right and the left coexist. Other methods are not able to reflect particles from the wall since they employ average velocities.

For the last test case, two different velocities were also given to the particles at the beginning. The difference between this test case and the previous ones is that the group of particles is located near the wall. Therefore, near the impingement point at the wall, the particles are characterized by four different velocity values: $(1, 0)$, $(0.5, 0)$, $(-1, 0)$, $(-0.5, 0)$. As it can be seen, HyQMOM cannot capture exactly the results as expected, since it does not predict the phase segregation as in the previous test case. It should be noted here that the initial condition of the particles of these two test cases are exactly identical, except for the particle location. In previous test cases, the particles are initialized on the right region which is far away from the impinging wall, before the particles impinge the wall, phase segregation occurs. In this test case, particles are initialized on the left region, which is close to the impinging wall. Before particles impinge the wall, fully phase segregation does occur.

3.4. Two-dimensional PTC and collision problems

In this section, we simulate PTC problems. It is well known that kinetic equations without collision flux yield multi-valued solutions. Only Lagrangian particle tracking can predict reasonable results, as it was proven that Eulerian methods typically yield delta shock. In this test case at the beginning the particles are injected from the bottom of the computational domain consisting of a square zone $x \sqsubset [0 : 1], y \sqsubset [0 : 1]$. The top of the computational domain is free outlet. The left and right sides of the domain are cyclic boundary conditions. The bottom is ordinary wall boundary condition with two inlets. Particles are injected from the inlets with different velocities. Therefore, these two streams of particles will collide in the middle of the computational domain and PTC will occur. The total particle number density $m_{0,0}$ predicted by CHyQMOM with and without drag are reported in Fig. 8.

As it can be seen CHyQMOM can treat non-regular angles without any problem. It can be seen that the mean particle velocity is calculated in almost all the cells of the computational domain. Only a small portion has vacuum states (e.g., lower left and right corners). This is because of the intrinsic diffusive nature of the Eulerian method. Moreover, the maximum value of the particle velocity exists

just above the lower-left corner, where the weights are extremely low. In order to maintain the realizability of the moments set, it is better to ensure that the CFL condition is respect and that the CFL number is kept smaller than one. In such cases, high velocity may introduce stability problems. In our test cases, no such problems are found. However, one can use a limiter to handle such issue and it was discussed in other works [46]. When the drag force is exerted, it can be seen that the particles tends to move along the fluid velocity. If the fluid velocity is null, the particles eventually stop at certain location, with unavoidable numerical diffusion.

3.5. Taylor-Green flow with one-way coupling

In this section, we simulate the particle flow in a Taylor-Green vortex (TGV) as done in other works with cyclic boundary conditions [15]. The uniform mesh consists of 10.000 cells. The fluid velocity is stationary and at the beginning, particles are distributed in a small circle just above the center of the TGV and their velocity is null. Due to the drag force, particles move along the streamline. Since the particles are off the center, the symmetry of the particle flow is soon broken, so that it can be easily observed. For zero Stokes number the particle velocity is identical to the fluid velocity and particles follows exactly the fluid streamlines. For finite non-zero Stokes numbers the particles will simply rotate and remain inside the TGV without PTC, eventually all the particles would stop at the TGV corners where the fluid velocity is null. For large Stokes numbers some particles will cross the TGV border and enter to other vortex cells. Such behavior leads to PTC.

The larger the Stokes number is, the more evident PTC is. Such features make it an excellent test case for verifying the algorithm implementation, since the traditional method would fail in predicting the particle flow for large Stokes number. Our results for different Stokes numbers are shown in Figs. 9 and 10.

For small Stokes number (e.g., $\mathrm{St} = 0.2$), it can be seen that particles begin to enter the neighbor TGV cell at $t \approx 1.5$. At $t \approx 2$ and $t \approx 2.3$, PTC occurs at the left and right top corners of the computational domain. After that, particles keep moving to other TGV cells without being trapped at the corners. At $t \approx 3.3$, PTC occurs at the right bottom corner. At all the time instants, CHyQMOM successfully handle the PTC without any problems. For larger Stokes numbers, particles do not respond quickly to the fluid drag and particle-crossing trajectories occur quite often. Similar particle flows are also found at the beginning of the simulations, but they differ with previous cases after $t \approx 2.6$. This is because more PTC occurs, more CHyQMOM can capture the main PTC features. However, at each vortex corner, CHyQMOM cannot capture the fine details due to the fact that only three nodes are employed. Moreover, it can be observed from the results, as reported for $t = 3, 3.3, 3.5, 3.6$ s, that a strong particle stripe movement is formed. This is because the particles are distributed asymmetrically in the computational domain. Particles movement in a symmetric TGV is

provided in Appendix C, whereas the animation of the evolution of $m_{0,0}$ in time for these test cases is also provided in the supplementary materials.

3.6. Two-dimensional simulation of a turbulent riser with two-way coupling

At last a typical turbulent dilute gas-solid riser is simulated. It is well-known that dilute gas-particle flows cannot be simulated by the traditional two-fluid model (TFM) since it assumes that flow is collision dominated and nearly at equilibrium. Typically Lagrange particle tracking is a good alternative. In this section, we employ our OpenFOAM-7 implementation of CHyQMOM to simulate dilute gas-particle flows to investigate the related numerical aspects. The physical properties used in the turbulent riser are taken from other works for comparison [45], in which the TP QMOM was used. The computational domain consists of 20×200 cells. The top and bottom are cyclic boundary conditions. Left and right are rebounded walls.

The instantaneous particle phase volume fraction in the riser is reported in Fig. 11. At the beginning of the simulation, we observe a more concentrated core zone and a wall region zone with fewer particles. This is because the particles start to move away from the walls towards the center of the channel since they are reflected by the wall. This leads to the formation of two strips almost parallel to the wall. Such predictions were also observed in other works [25, 45]. Specifically, in the work of Helland et al. [25], Lagrangian particle tracking was employed simulate particle clustering and similar strips were formed at the beginning of the simulation. However, this particle strip is not stable. At a certain time instant, it leads to the typical chaotic flow behaviour. Moreover, the granular temperature is high near the wall. This is due to the wall-particle collisions, since the particles are reflected once they impinge the wall. As it can be seen in Fig. 11 at the core region of the channel, particles are convected upwards due to the upward gas velocity. As time elapses, particles tend to form clusters and the particle phase volume fraction increases. Finally, the effect of particle clustering is strong enough and they tend to fall down. At the wall region, it can be also observed that the particles tend to fall along the wall. On the other hand, in certain regions, the gas velocity can be strong enough to stop the particle falling and convect them upwards. Readers are suggested to the supplementary file in which we provide an animation of our simulation results.

In order to gain a deeper understanding of the numerical challenges that our CHyQMOM implementation in OpenFOAM-7 is capable of overcoming, the results obtained by employing different time steps and restitution coefficients are discussed. The time step is important in QBMM simulations as demonstrated in other works [30, 45]. In this test case, we employ the following method to calculate the time step:

$$\Delta t = \frac{1}{\mathrm{Te}} \min\left(\tau_{\mathrm{c}}^{m}, \tau_{\mathrm{p}}^{m}\right), \tag{44}$$

where τ_c^m is the minimum collision time in the computational domain, τ_p^m is the minimum particle relaxation time, Te is a user-defined parameter, which we set it to be 5, 10 and 20 separately. The particle phase volume fractions predicted at different Te values are all comparable with each other. However, for large Te, the time-averaged gas velocity shows a non-symmetrical radial profile, which is not consistent with other results. It is suggested to employ a smaller time step in order to obtain reasonable results. It should be also stressed here that in some cells, the collision time is very small and that makes the calculation less efficient. At last, the simulation launched with $e = 0.95$ is reported in Fig. 12. As it can be seen in certain cells, the predicted particle volume phase fraction is extremely high. It implies that the collision time is extremely small and the simulation quite unstable. This is also consistent with the physics observed in the work by Cloete et al. [12] in which the phase fraction predicted by the Eulerian-Lagrangian method is higher than 0.6 locally.

4. Conclusions

In this work, CHyQMOM for the kinetic/Boltzmann equation was implemented in the open-source CFD package OpenFOAM-7 for the simulation of gas-solid flows. The transport equations for the moments of the NDF, representing the fate of the solid particles, are coupled with the Navier-Stokes equation for the gas phase, under one- and two-way coupling. Several test cases were considered, leading to the following conclusions:

- In the one-way coupling test cases, we showed that the three-node CHyQMOM implementation in OpenFOAM-7 can successfully predict PTC; the effect of PTC also depends on the collision term, which in turn depends on particle physical properties;
- Our CHyQMOM implementation in OpenFOAM-7 can capture the main features of the asymmetric and symmetric particle Taylor-Green vortex flow;
- As far as two-way coupling is concerned, since HyQMOM predicts permutation dependent particle velocities, a more general operator splitting procedure, was implemented, to handle complex drag term, which was never reported in previous works;
- A turbulent gas-particle riser was also considered, showing that our three-node CHyQMOM implementation in OpenFOAM-7 is robust enough to capture the dynamics of these complex gas-particle flows; the predicted results also agree well with the theory.

Our implementation confirmed some issues of CHyQMOM including that:

- the three-node HyQMOM cannot reconstruct accurately a tri-disperse velocity distribution, since in the equation system the last moment is fixed in order to ensure hyperbolicity;

- the collision flux should always be included in the simulations even when the global particle phase fraction is low; it was in fact found that in certain cells the phase fraction was high enough not to allow for the collision flux to be neglected;
- for collision dominated cells, the time step should be small enough to ensure that reasonable results are obtained.

Acknowledgement

Dongyue Li thanks Shaoze Xiao (Tongji University) for preparing Fig B.13. Dongyue Li also wants to acknowledge the small, but active, CFD community in China for all their continued encouragement.

Supplementary files

The animation of asymmetric particle flows in TGV and turbulent riser are provided as supplementary files.

References

[1] G. Boccardo, I.M. Sokolov, and A. Paster. An improved scheme for a robin boundary condition in discrete-time random walk algorithms. *Journal of Computational Physics*, 374:1152–1165, 2018.

[2] G. Boccardo, R. Sethi, and D.L. Marchisio. Fine and ultrafine particle deposition in packed-bed catalytic reactors. *Chemical Engineering Science*, 198:290–304, 2019.

[3] G. Boccardo, E. Crevacore, A. Passalacqua, and M. Icardi. Computational analysis of transport in three-dimensional heterogeneous materials: An OpenFOAM-based simulation framework. *Computing and Visualization in Science*, 23, 2020.

[4] N. Böhmer and M. Torrilhon. Entropic quadrature for moment approximations of the Boltzmann-BGK equation. *Journal of Computational Physics*, 401:108992, 2020.

[5] C. Chalons, R.O. Fox, and M. Massot. A multi-gaussian quadrature method of moments for gas-particle flows in a LES framework. In *Proceedings of the Summer Program*, pages 347–358. Center for Turbulence Research, 2010.

[6] C. Chalons, D. Kah, and M. Massot. Beyond pressureless gas dynamics: quadrature-based velocity moment models. *arXiv preprint arXiv:1011.2974*, 2010.

[7] C. Chalons, R.O. Fox, F. Laurent, M. Massot, and A. Vié. Multivariate Gaussian extended quadrature method of moments for turbulent disperse multiphase flow. *Multiscale Modeling & Simulation*, 15:1553–1583, 2017.

[8] X. Chen and J. Wang. A comparison of two-fluid model, dense discrete particle model and CFD-DEM method for modeling impinging gas–solid flows. *Powder technology*, 254:94–102, 2014.

[9] J. Cheng and R.O. Fox. Kinetic modeling of nanoprecipitation using cfd coupled with a population balance. *Industrial & Engineering Chemistry Research*, 49:10651–10662, 2010.

[10] J. Cheng, R.D. Vigil, and R.O. Fox. A competitive aggregation model for flash nanoprecipitation. *Journal of Colloid And Interface Science*, 351:330–342, 2010.

[11] J. Cloete, S. Cloete, S. Radl, and S. Amini. Development and verification of anisotropic solids stress closures for filtered two fluid models. *Chemical Engineering Science*, 192:906–929, 2018.

[12] S. Cloete, S. Johansen, M. Braun, B. Popoff, and S. Amini. Evaluation of a lagrangian discrete phase modeling approach for resolving cluster formation in CFB risers. In *7th International Conference on Multiphase Flow*, 2010.

[13] E. Crevacore, T. Tosco, R. Sethi, G. Boccardo, and D.L. Marchisio. Recirculation zones induce non-fickian transport in three-dimensional periodic porous media. *Physical Review E*, 94:053118, 2016.

[14] E. Crevacore, G. Boccardo, A. Grillo, D. Marchisio, and R. Sethi. Pore-scale simulations of particle transport for groundwater remediation: the effect of gravitational settling. *Chemical Engineering Transactions*, 60:193–198, 2017.

[15] O. Desjardins, R.O. Fox, and P. Villedieu. A quadrature-based moment method for dilute fluid-particle flows. *Journal of Computational Physics*, 227:2514–2539, 2008.

[16] F. Forgues, L. Ivan, A. Trottier, and J.G. McDonald. A Gaussian moment method for polydisperse multiphase flow modelling. *Journal of Computational Physics*, 398:108839, 2019.

[17] R.O. Fox. A quadrature-based third-order moment method for dilute gas-particle flows. *Journal of Computational Physics*, 227:6313–6350, 2008.

[18] R.O. Fox. Higher-order quadrature-based moment methods for kinetic equations. *Journal of Computational Physics*, 228:7771–7791, 2009.

[19] R.O. Fox, F. Laurent, and A. Vié. Conditional hyperbolic quadrature method of moments for kinetic equations. *Journal of Computational Physics*, 365:269–293, 2018.

[20] H. Grad. On the kinetic theory of rarefied gases. *Communications On Pure And Applied Mathematics*, 2:331–407, 1949.

[21] C.P.T. Groth and J.G. McDonald. Towards physically realizable and hyperbolic moment closures for kinetic theory. *Continuum Mechanics and Thermodynamics*, 21:467–493, 2009.

[22] V.K. Gupta. Moment theories for a d-dimensional dilute granular gas of Maxwell molecules. *arXiv preprint arXiv:1903.00966*, 2019.

[23] V.K. Gupta, H. Struchtrup, and M. Torrilhon. Regularized moment equations for binary gas mixtures: Derivation and linear analysis. *Physics of Fluids*, 28:042003, 2016.

[24] V.K. Gupta, P. Shukla, and M. Torrilhon. Higher-order moment theories for dilute granular gases of smooth hard spheres. *Journal of Fluid Mechanics*, 836:451–501, 2018.

[25] E. Helland, R. Occelli, and L. Tadrist. Numerical study of cluster formation in a gas–particle circulating fluidized bed. *Powder Technology*, 110:210–221, 2000.

[26] Q. Huang, S. Li, and W. Yong. Stability analysis of quadrature-based moment methods for kinetic equations. *SIAM Journal on Applied Mathematics*, 80:206–231, 2020.

[27] X. Ji, L. Pan, W. Shyy, and K. Xu. A compact fourth-order gas-kinetic scheme for the Euler and Navier–Stokes equations. *Journal of Computational Physics*, 372:446–472, 2018.

[28] B. Kong and R.O. Fox. A solution algorithm for fluid–particle flows across all flow regimes. *Journal of Computational Physics*, 344:575–594, 2017.

[29] B. Kong and R.O. Fox. A moment-based kinetic theory model for polydisperse gas–particle flows. *Powder Technology*, 365:92–105, 2019.

[30] B. Kong, R.O. Fox, H. Feng, J. Capecelatro, R. Patel, O. Desjardins, and R.O. Fox. Euler–euler anisotropic gaussian mesoscale simulation of homogeneous cluster-induced gas–particle turbulence. *AIChE Journal*, 63:2630–2643, 2017.

[31] F. Laurent and T. Nguyen. Realizable second-order finite-volume schemes for the advection of moment sets of the particle size distribution. *Journal of Computational Physics*, 337:309–338, 2017.

[32] D. Li, A. Buffo, W. Podgórska, D. Marchisio, and Z. Gao. Investigation of droplet breakup in liquid–liquid dispersions by CFD–PBM simulations: The influence of the surfactant type. *Chinese Journal of Chemical Engineering*, 25:1369–1380, 2017.

[33] D. Li, Z. Gao, A. Buffo, W. Podgorska, and D. Marchisio. Droplet breakage and coalescence in liquid–liquid dispersions: Comparison of different kernels with EQMOM and QMOM. *AIChE Journal*, 63:2293–2311, 2017.

[34] D. Li, Z. Li, and Z. Gao. Compressibility induced bubble size variation in bubble column reactors: Simulations by the CFD–PBE. *Chinese Journal of Chemical Engineering*, 26(10):2009–2013, 2018.

[35] D. Li, Z. Li, and Z. Gao. Quadrature-based moment methods for the population balance equation: An algorithm review. *Chinese Journal of Chemical Engineering*, 27:483–500, 2019.

[36] D. Li, D. Marchisio, C. Hasse, and D. Lucas. Comparison of Eulerian QBMM and classical Eulerian–Eulerian method for the simulation of polydisperse bubbly flows. *AIChE Journal*, 65: e16732, 2019.

[37] D. Li, D. Marchisio, C. Hasse, and D. Lucas. twoWayGPBEFoam: An open-source Eulerian QBMM solver for monokinetic bubbly flows. *Computer Physics Communications*, page in press, 2019.

[38] H.Å. Lindborg, M. Lysberg, and H.A. Jakobsen. Practical validation of the two-fluid model applied to dense gas–solid flows in fluidized beds. *Chemical Engineering Science*, 62:5854–5869, 2007.

[39] C. Liu, Z. Wang, and K. Xu. A unified gas-kinetic scheme for continuum and rarefied flows VI: Dilute disperse gas-particle multiphase system. *Journal of Computational Physics*, 386:264–295, 2019.

[40] D.L. Marchisio and R.O. Fox. *Computational models for polydisperse particulate and multiphase systems*. Cambridge University Press, 2013.

[41] J. McDonald and C. Groth. Numerical modeling of micron-scale flows using the gaussian moment closure. In *35th AIAA Fluid Dynamics Conference and Exhibit*, page 5035, 2005.

[42] J. McDonald and M. Torrilhon. Affordable robust moment closures for CFD based on the maximum-entropy hierarchy. *Journal of Computational Physics*, 251:500–523, 2013.

[43] R. McGraw. Description of aerosol dynamics by the quadrature method of moments. *Aerosol Science and Technology*, 27:255–265, 1997.

[44] T. Nguyen, F. Laurent, R.O. Fox, and M. Massot. Solution of population balance equations in applications with fine particles: mathematical modeling and numerical schemes. *Journal of Computational Physics*, 325:129–156, 2016.

[45] A. Passalacqua and R.O. Fox. Implementation of an iterative solution procedure for multi-fluid gas–particle flow models on unstructured grids. *Powder Technology*, 213:174–187, 2011.

[46] R.G. Patel, O. Desjardins, and R.O. Fox. Three-dimensional conditional hyperbolic quadrature method of moments. *Journal of Computational Physics: X*, 1:100006, 2019.

[47] C. Peng, B. Kong, J. Zhou, B. Sun, A. Passalacqua, S. Subramaniam, and R.O. Fox. Implementation of pseudo-turbulence closures in an eulerian–eulerian two-fluid model for non-isothermal gas–solid flow. *Chemical Engineering Science*, 207:663–671, 2019.

[48] J.S. Sachdev, C.P.T. Groth, and J.J. Gottlieb. Numerical solution scheme for inert, disperse, and dilute gas-particle flows. *International journal of multiphase flow*, 33(3):282–299, 2007.

[49] H. Struchtrup. Macroscopic transport equations for rarefied gas flows. In *Macroscopic Transport Equations for Rarefied Gas Flows*, pages 145–160. Springer, 2005.

[50] H. Struchtrup and M. Torrilhon. Regularization of Grad's 13 moment equations: derivation and linear analysis. *Physics of Fluids*, 15:2668–2680, 2003.

[51] D. Sun, A. Garmory, and G.J. Page. A robust two-node, 13 moment quadrature method of moments for dilute particle flows including wall bouncing. *Journal of Computational Physics*, 330:493–509, 2017.

[52] A. Vié, F. Doisneau, and M. Massot. On the anisotropic gaussian velocity closure for inertial-particle laden flows. *Communications in Computational Physics*, 17:1–46, 2015.

[53] 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*, 230:5328–5352, 2011.

[54] J. Wang, M.A. van der Hoef, and J.A.M. Kuipers. Why the two-fluid model fails to predict the bed expansion characteristics of Geldart A particles in gas-fluidized beds: a tentative answer. *Chemical Engineering Science*, 64:622–625, 2009.

[55] C.Y. Wen and Y.H. Yu. A generalized method for predicting the minimum fluidization velocity. *AIChE Journal*, 12:610–612, 1966.

[56] T. Xiao, C. Liu, K. Xu, and Q. Cai. A velocity-space adaptive unified gas kinetic scheme for continuum and rarefied flows. *Journal of Computational Physics*, 415:109–535, 2020.

[57] K. Xu and J.C. Huang. A unified gas-kinetic scheme for continuum and rarefied flows. *Journal of Computational Physics*, 229:7747–7764, 2010.

[58] C. Yoon and R. McGraw. Representation of generally mixed multivariate aerosols by the quadrature method of moments: I. Statistical foundation. *Journal of Aerosol Science*, 35:561–576, 2004.

[59] C. Yuan and R.O. Fox. Conditional quadrature method of moments for kinetic equations. *Journal of Computational Physics*, 230:8216–8246, 2011.

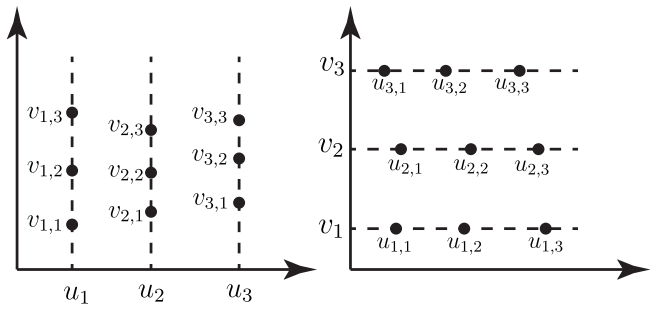


Figure 1: Positions in the (u, v)-plane of the nine nodes of bivariate CQMOM/CHyQMOM for $N_1 = N_2 = 3$. First permutation: $f(u, v) = f(u)f(v|u)$ (left). Second permutation: $f(u, v) = f(v)f(u|v)$ (right).

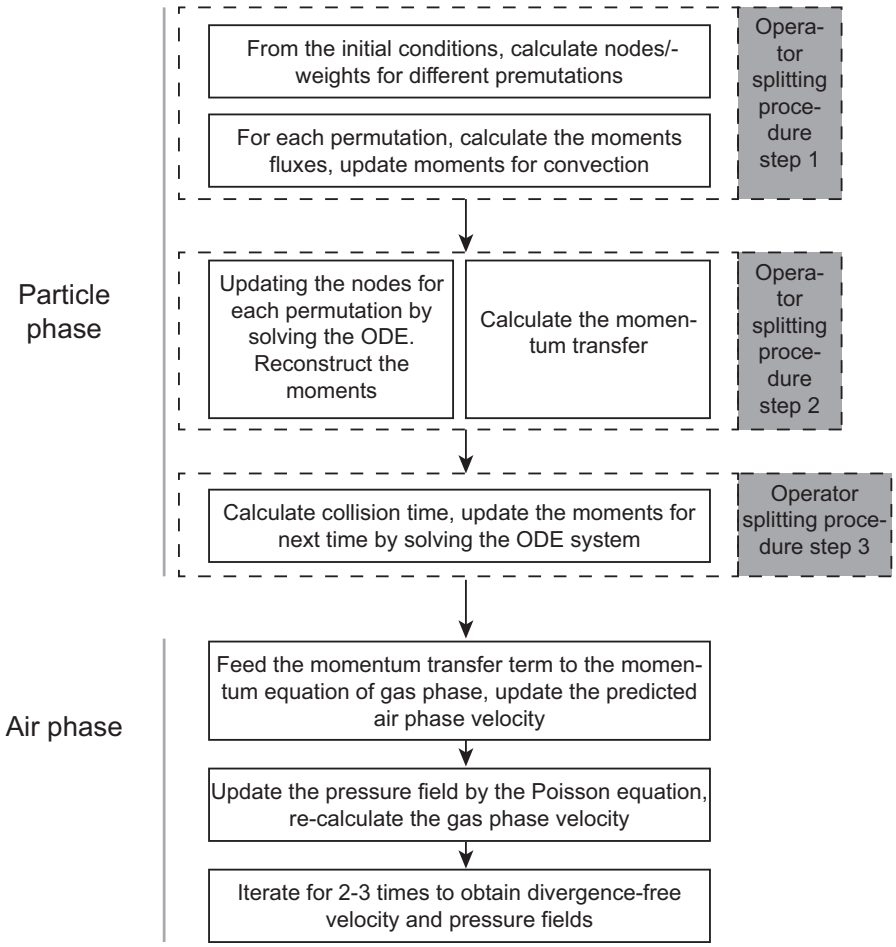


Figure 2: Two-way coupling procedure employed in this work.

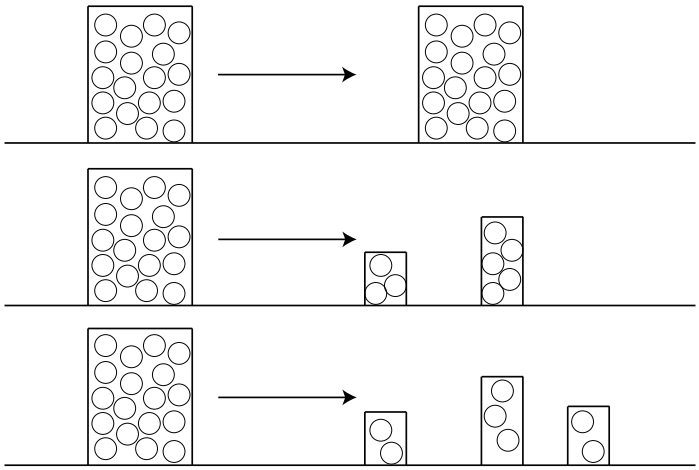


Figure 3: Particle size segregation phenomenon simulated by three-nodes HyQMOM. Top: initialization with one velocity, middle: initialization with two velocities, bottom: initialization with three velocities.

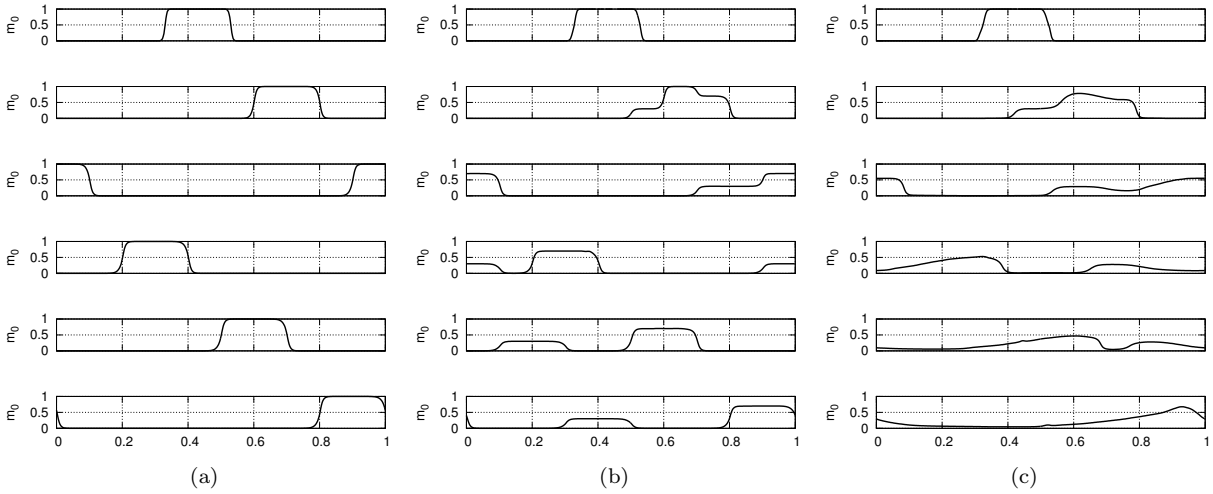


Figure 4: Time evolution of m_0 for particle size segregation at different time instants (from top to bottom: $t = 0.05$ s, $t = 0.5$ s, $t = 1.5$ s, $t = 2$ s, $t = 2.5$ s). (a) mono-disperse velocity distribution : $u_0 = u_1 = u_2 = 0.6$ m/s, $w_0 = w_1 = w_2 = 1$. (b) bi-disperse velocity distribution: $u_0 = 0.6$ m/s, $u_1 = 0.4$ m/s $u_2 = 0$ m/s. $w_0 = 0.7, w_1 = 0.3, w_2 = 0$. (c) tri-disperse velocity distribution: $u_0 = 0.6$ m/s, $u_1 = 0.4$ m/s $u_2 = 0.2$ m/s, $w_0 = 0.5, w_1 = 0.3, w_2 = 0.2$.

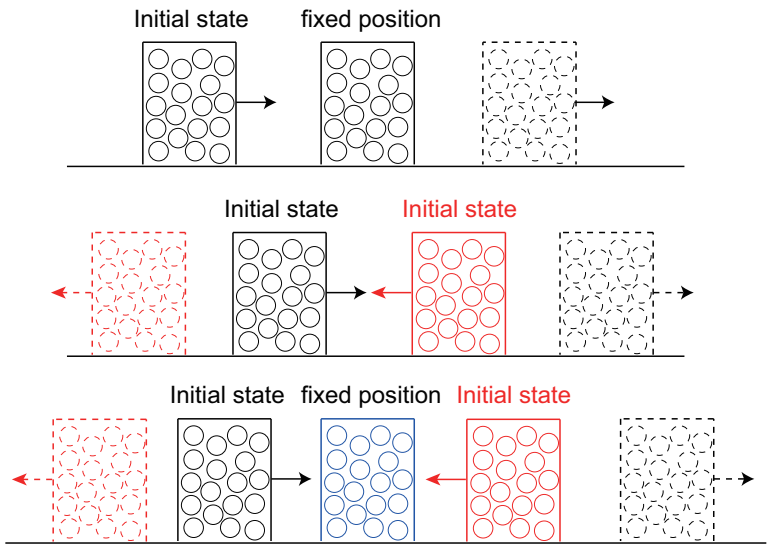


Figure 5: Particle crossing phenomenon simulated by three-node HyQMOM. Top: initialization with two groups of particles, one group fixed and the other one moving to the right. Middle: initialization with two groups of particles, one is moving to the right, whereas the other one is moving to the left. Bottom: initialization with three groups of particles. One is moving to the right whereas the other one is moving to the left. One is fixed in the middle.

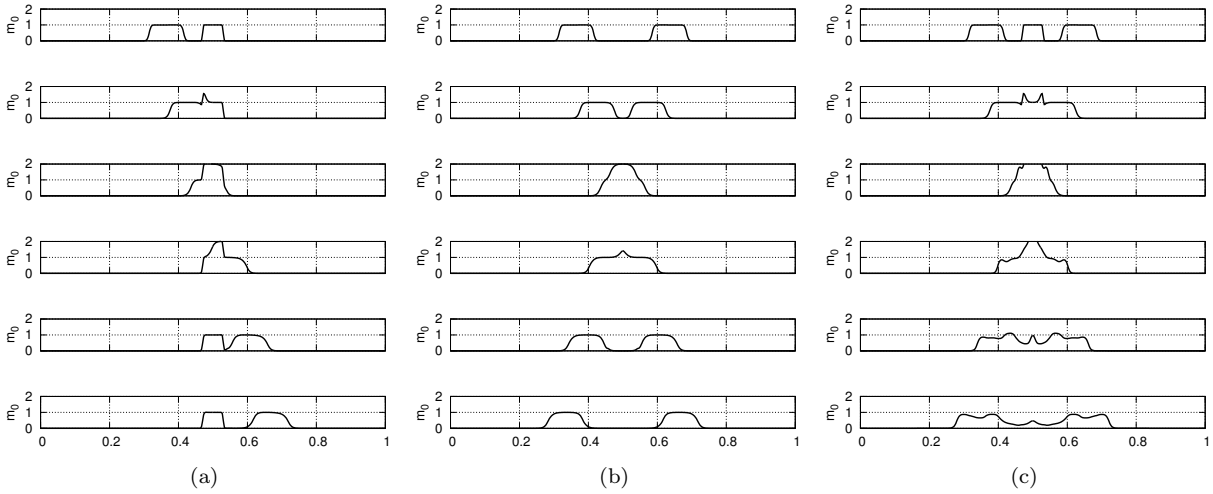


Figure 6: Time evolution of m_0 for particle crossing problems at different time instants (from top to bottom). Left: initialization with two groups of particles, one group fixed and the other one moving to the right. Middle: initialization with two groups of particles, one moving to the right whereas the other one moving to the left. Right: initialization with three groups of particles. One is moving to the right whereas the other one is moving to the left. One is fixed in the middle.

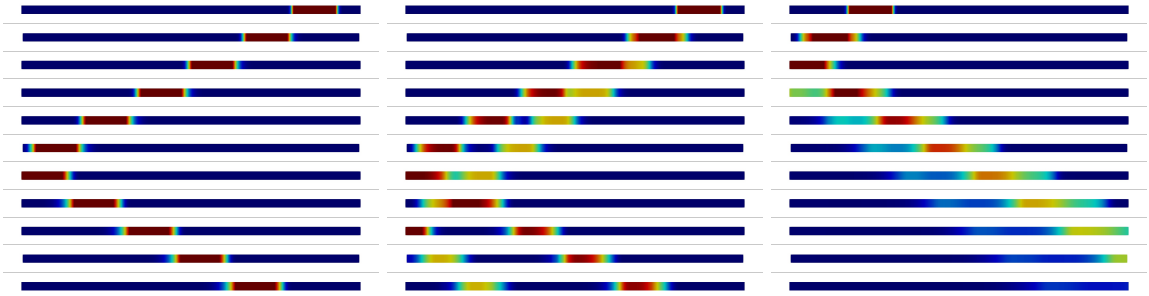


Figure 7: Time evolution of m_0 for wall rebounding problem (from top to bottom). Left: mono-disperse initial velocity distribution. Middle: bi-disperse initial velocity distribution. Right: tri-disperse initial velocity distribution.

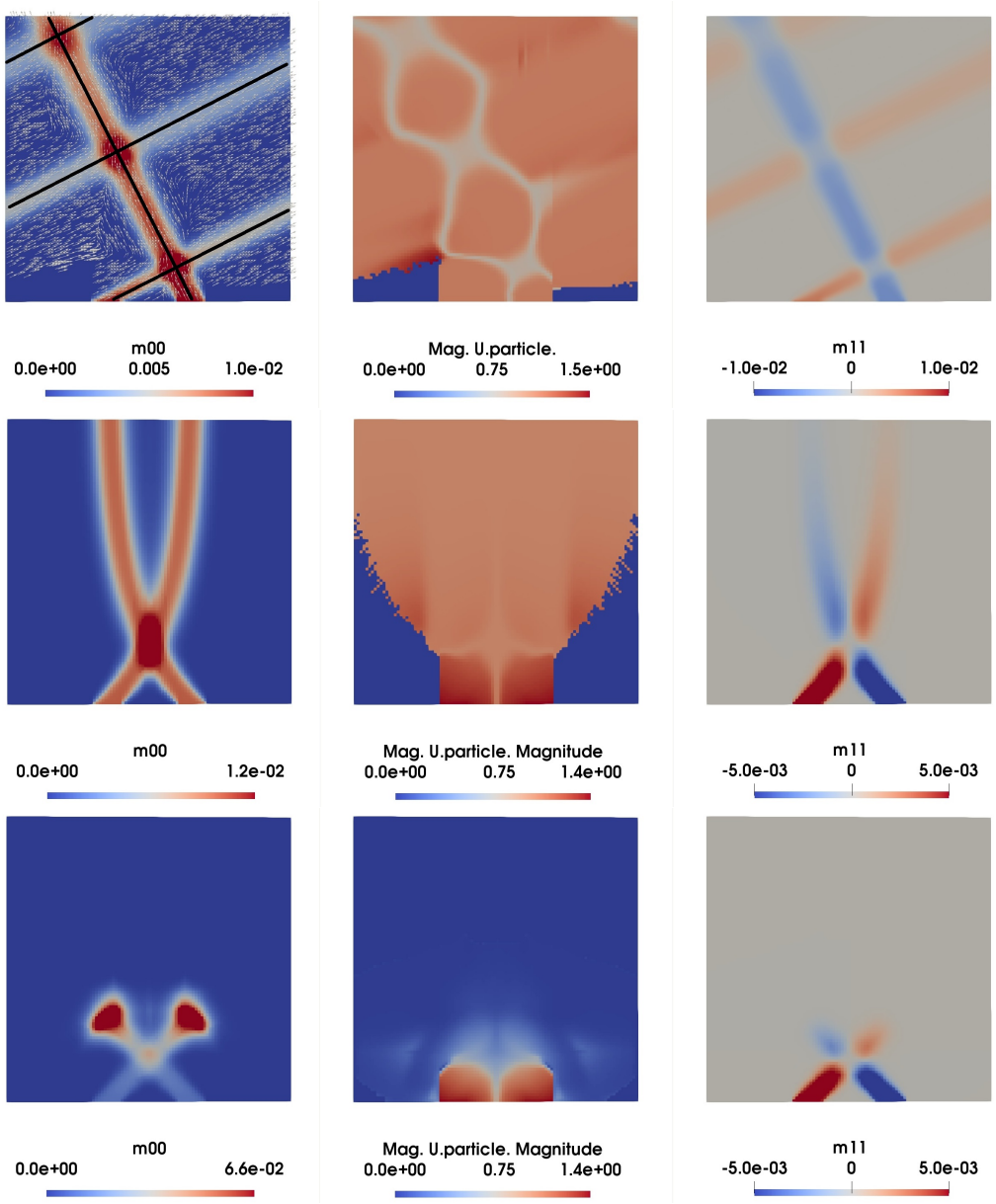


Figure 8: Contour plots of total particle number density $m_{0,0}$ (left), magnitude of the particle velocity (middle) and $m_{1,1}$ (right) predicted by three-node CHyQMOM. Top: the inlet velocity of left inlet is $(1, 0.5)$, the inlet of velocity of right inlet is $(1, 2)$. Analytical solution: black lines. Drag force is not included. Middle: the inlet velocity of left inlet is $(1, 1)$, the inlet of velocity of right inlet is $(1, -1)$. Fluid velocity is $(0, 1)$. Constant drag force is included. Bottom: the inlet velocity of left inlet is $(1, 1)$, the inlet of velocity of right inlet is $(1, -1)$. Fluid velocity: $(0, 0)$. Constant drag force is included.

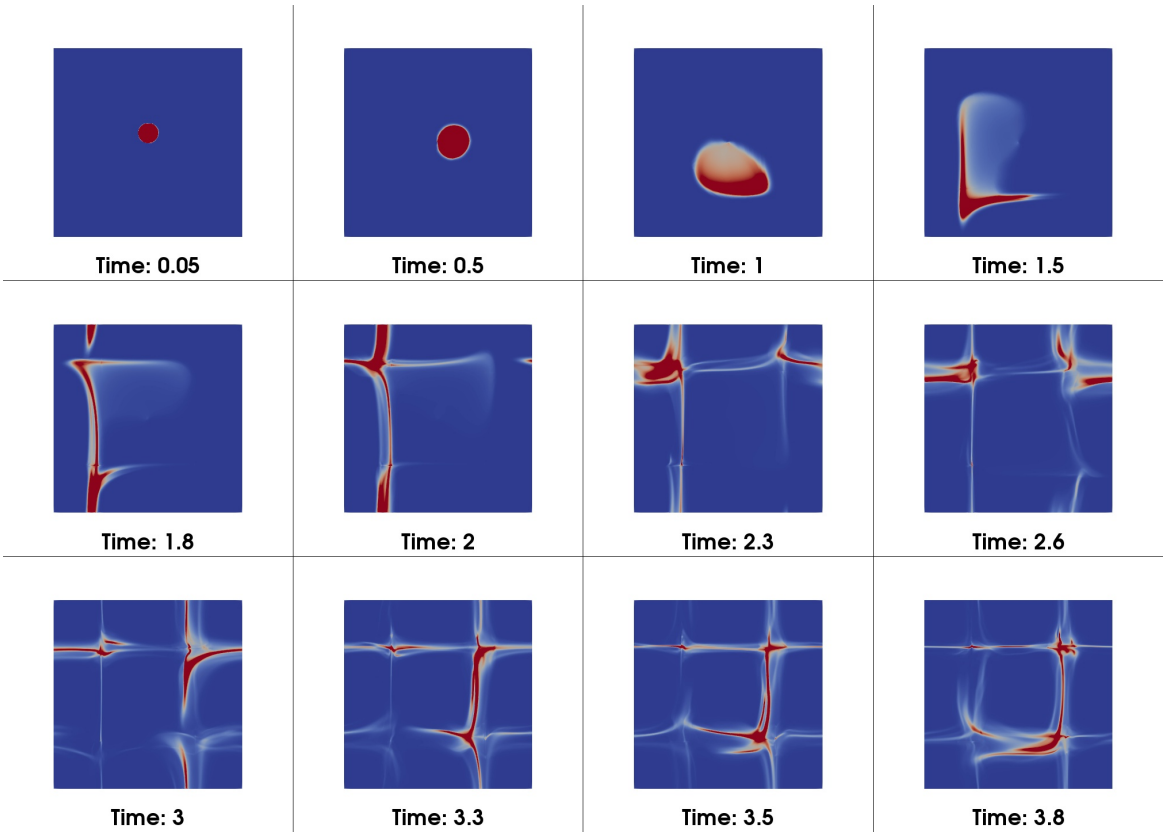


Figure 9: Contour plots of $m_{0,0}$ predicted by three-node CHyQMOM for Taylor-Green vortex gas-particle flow at different time instants for $\mathrm{St} = 0.2$.

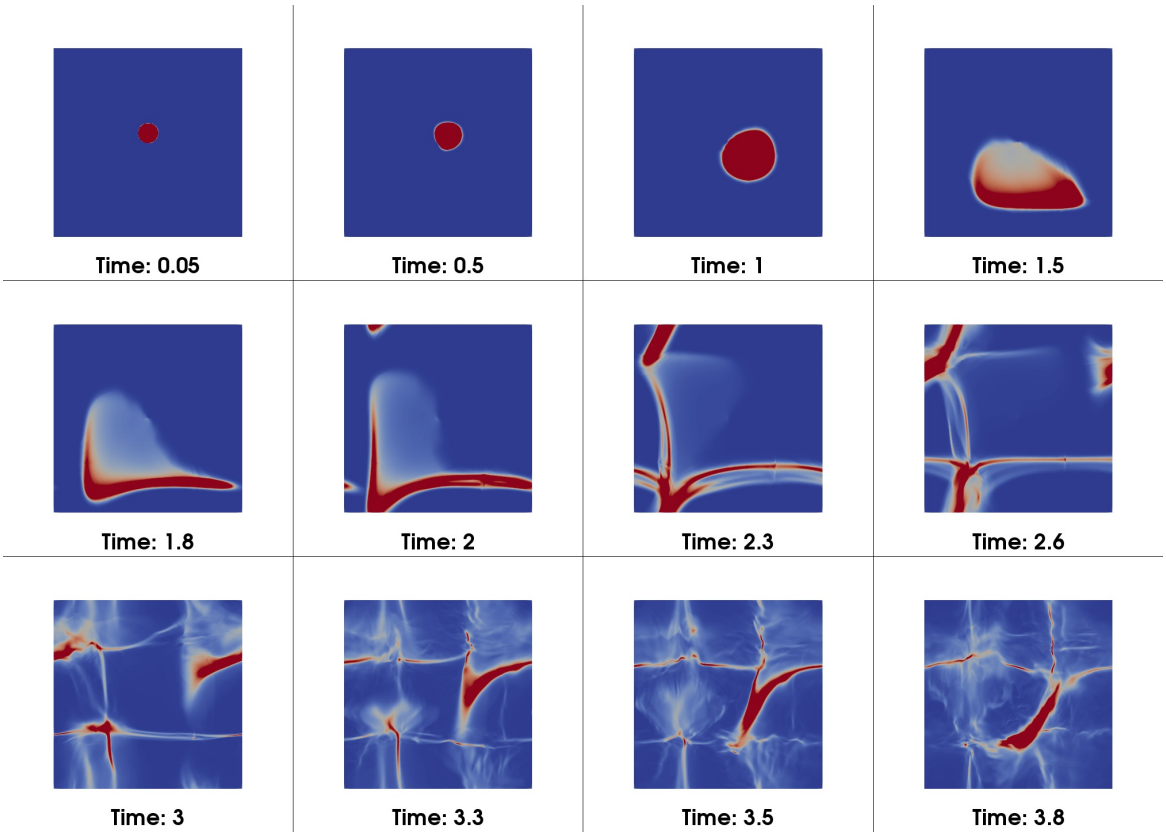


Figure 10: Contour plots of $m_{0,0}$ predicted by three-node CHyQMOM for Taylor-Green vortex gas-particle flow at different time instants for $\mathrm{St} = 0.5$.

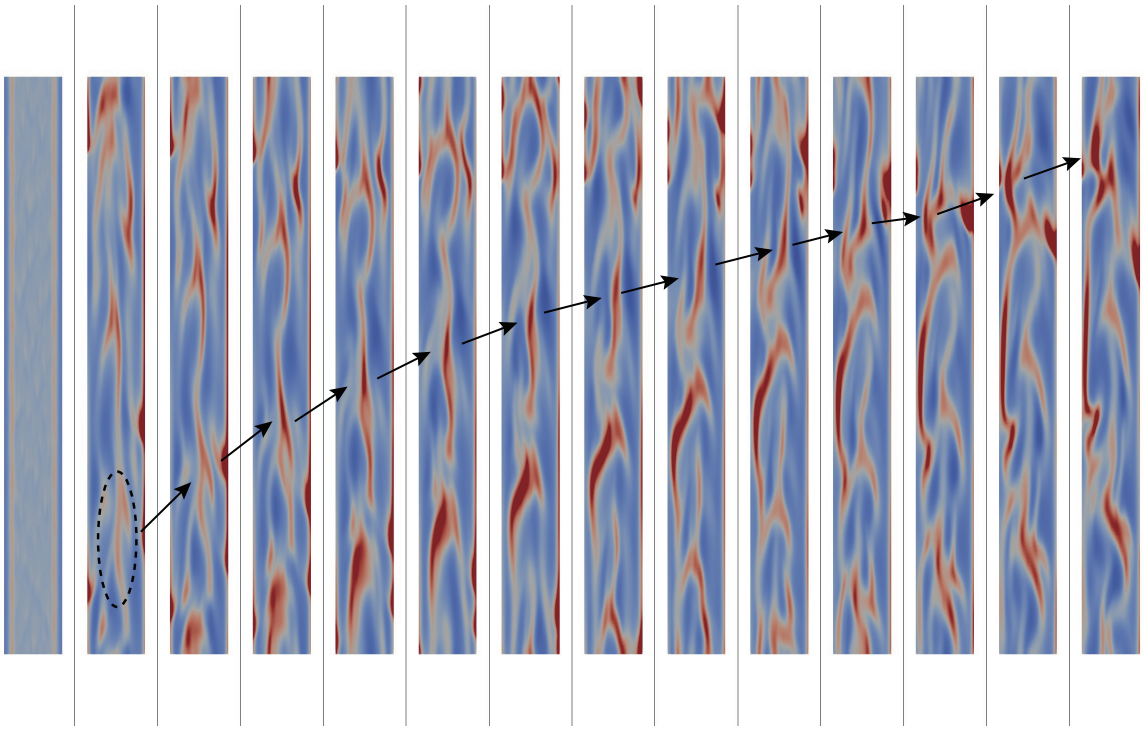


Figure 11: Instantaneous particle phase fraction in the riser. $0 < \alpha_{\mathrm{d}} < 0.1$. From left to right: $t = 0.2$s, $t = 2.05$s, $t = 2.08$s, $t = 2.11$s, $t = 2.14$s, $t = 2.17$s, $t = 2.2$s, $t = 2.23$s, $t = 2.26$s, $t = 2.29$s, $t = 2.32$s,$t = 2.35$s, $t = 2.38$s, $t = 2.41$s.

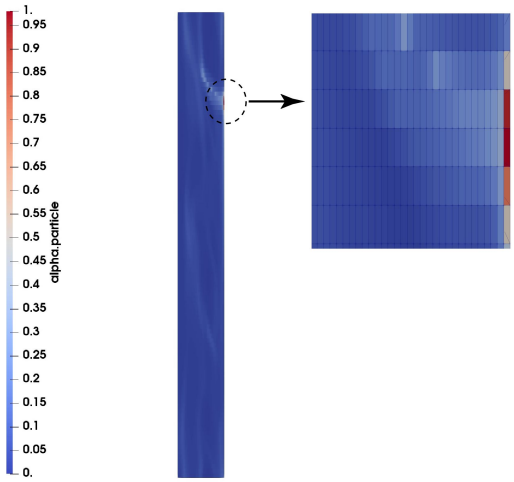


Figure 12: Contour plot of particle phase fraction predicted by the CHyQMOM for $e = 0.95$.

Appendix A. Bivariate CQMOM and CHyQMOM

Let us first consider a bivariate NDF $f(u,v)$ which can be written as follows:

$$f(u,v) = f(u)f(v|u), \tag{A.1}$$

where $f(v|u)$ corresponds to the conditional PDF for v when u is equal to a given value. It should be noted here that the NDF can be also written as follows:

$$f(u,v) = f(v)f(u|v) \tag{A.2}$$

resulting into another permutation. If we choose u as the primary nodes, the mixed moments of the NDF are defined by the following expressions:

$$m_{i,j} = \int u^i f(u) \left(\int v^j f(v|u) \mathrm{d}v \right) \mathrm{d}u = \sum_{\alpha=1}^{3} w_\alpha \left(u_\alpha + \frac{m_{1,0}}{m_{0,0}} \right)^i \left(\int v^j f\left(v|u_\alpha + m_{1,0}/m_{0,0}\right) \mathrm{d}v \right), \tag{A.3}$$

where $\frac{m_{1,0}}{m_{0,0}}$ is the average solid-particle velocity along the x direction. As well-known the lower-order moments have specific physical meanings. For example, $m_{0,0}$ denotes the total particle number density, $m_{1,0}$ denotes the x component of the particle momentum density, $m_{0,1}$ denotes the y component of the particle momentum density.

Similarly, if we choose v as the primary nodes, the mixed moments of the VDF are defined by:

$$m_{i,j} = \int v^j f(v) \left(\int u^i f(u|v) \mathrm{d}u \right) \mathrm{d}v = \sum_{\alpha=1}^{3} w_\alpha \left(v_\alpha + \frac{m_{0,1}}{m_{0,0}} \right)^i \left(\int u^j f(u|v_\alpha + m_{0,1}/m_{0,0}) \mathrm{d}u \right), \tag{A.4}$$

where $\frac{m_{0,1}}{m_{0,0}}$ is the average solid particle velocity along the y direction. The pure moments along the x and y directions can be defined by respectively as follows:

$$m_{i,0} = \sum_{\alpha=1}^{3} w_\alpha \left(\frac{m_{1,0}}{m_{0,0}} + u_\alpha \right)^i, \quad m_{0,j} = \sum_{\alpha=1}^{3} w_\alpha \left(\frac{m_{0,1}}{m_{0,0}} + v_\alpha \right)^j. \tag{A.5}$$

As long as the pure moments are defined, the primary weights and nodes in each direction (permutation) can be calculated from the pure central moments.

In order to calculate the secondary weights and nodes, mixed moments are required. When we use u as primary variable and v as secondary variable the conditional moments are defined as follows:

$$m_{j|\alpha} = \int \left(v - \frac{m_{0,1}}{m_{0,0}} \right)^j f(v|u_\alpha + m_{1,0}/m_{0,0}) \mathrm{d}v. \tag{A.6}$$

They are the unknown univariate moments of v conditioned on $u = \frac{m_{1,0}}{m_{0,0}} + u_\alpha$. Meanwhile, it can be seen that the mixed central moments can be calculated as follows:

$$\begin{aligned} C_{i,j} &= \frac{1}{m_{0,0}} \int \left(u - \frac{m_{1,0}}{m_{0,0}} \right)^i f(u) \left(\int \left(v - \frac{m_{0,1}}{m_{0,0}} \right)^j f(v|u) \mathrm{d}v \right) \mathrm{d}u \\ &= \frac{1}{m_{0,0}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^i \left(\int \left(v - \frac{m_{0,1}}{m_{0,0}} \right)^j f(v|u_\alpha + m_{1,0}/m_{0,0}) \mathrm{d}v \right) \\ &= \frac{1}{m_{0,0}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^i m_{j|\alpha} = \frac{1}{m_{0,0}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^i \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}. \end{aligned} \tag{A.7}$$

The conditional moments $m_{j|\alpha}$ can be calculated with the CQMOM algorithm by building the Vandermonde linear system. After the conditional moments are computed, the secondary nodes $v_{\alpha\beta}$ and weights $w_{\alpha\beta}$ can be calculated by the moment inversion algorithm.

The final bivariate NDF can be expressed as follows:

$$f(u,v) = \sum_{\alpha=1}^{3} w_\alpha \delta \left(u - \frac{m_{1,0}}{m_{0,0}} - u_\alpha \right) \sum_{\beta=1}^{3} w_{\alpha\beta} \delta \left(v - \frac{m_{0,1}}{m_{0,0}} - v_{\alpha\beta} \right). \tag{A.8}$$

However, building the Vandermonde system requires a larger moments set. Readers are referred to the seminal work in which CQMOM was developed [59] or to Fig. 4 of our previous work [35]. One aim of CHyQMOM is to reduce the computational cost by employing a smaller moment set.

Instead of Eq. (A.7), in the modified CHyQMOM, the mixed central moments are calculated by using the following expression:

$$C_{i,j} = \frac{1}{m_{0,0}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^i \sum_{\beta=1}^{3} w_{\alpha\beta} (m_{1|\alpha} + v_{\alpha\beta})^j. \tag{A.9}$$

Therefore, the bivariate distribution in the modified CHyQMOM can be written as follows:

$$f(u,v) = \sum_{\alpha=1}^{3} w_\alpha \delta \left(u - \frac{m_{1,0}}{m_{0,0}} - u_\alpha \right) \sum_{\beta=1}^{3} w_{\alpha\beta} \delta \left(v - \frac{m_{0,1}}{m_{0,0}} - m_{1|\alpha} - v_{\alpha\beta} \right). \tag{A.10}$$

As long as $m_{1|\alpha}$, $w_{\alpha\beta}$ and $v_{\alpha\beta}$ are calculated, the velocity distribution can be reconstructed. The first step of the modified CQMOM is to seek expansion in the form for the first-order conditional moments:

$$m_{1|\alpha} = a_0 + a_1 u_\alpha + a_2 u_\alpha^2 + \cdots + a_{n-1} u_\alpha^{n-1}, \tag{A.11}$$

where $a_0, a_1, ..., a_{n-1}$ are the polynomial coefficients. It is obvious that $m_{1|\alpha}$ can be seen as velocity and they are also called conditional velocities. Because $\sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta} = 0$ as it is seen in Eq. (A.21),

substituting Eq. (A.11) into Eq. (A.7) yields:

$$C_{i,1} = \frac{1}{m_{0,0}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^i \left(a_0 + a_1 u_\alpha + a_2 u_\alpha^2 + \cdots + a_{n-1} u_\alpha^{n-1}\right). \tag{A.12}$$

The polynomial coefficients can be calculated from the following linear system:

$$\begin{pmatrix} C_{0,1} \\ C_{1,1} \\ C_{2,1} \\ \vdots \\ C_{n-1,1} \end{pmatrix} = \frac{1}{m_{0,0}} \begin{pmatrix} 1 & 0 & C_{2,0} & \cdots & C_{n-1,0} \\ 0 & C_{2,0} & C_{3,0} & \cdots & C_{n,0} \\ C_{2,0} & C_{3,0} & C_{4,0} & \cdots & C_{n+1,0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{n-1,0} & C_{n,0} & C_{n+1,0} & \cdots & C_{2n-2,0} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_{n-1} \end{pmatrix}. \tag{A.13}$$

In order to calculate the polynomial coefficients, the mixed central moments are required. In this work, we employ the minimal first order approximation:

$$m_{1|\alpha} = a_0 + a_1 u_\alpha, \tag{A.14}$$

and the coefficients can be calculated simply by imposing the following equalities:

$$C_{0,1} = \frac{a_0}{m_{0,0}} \sum_{\alpha=1}^{3} w_\alpha = 0, \quad C_{1,1} = \frac{1}{m_{0,0}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha (a_0 + a_1 u_\alpha). \tag{A.15}$$

This results in:

$$a_0 = 0, \quad a_1 = \frac{C_{1,1}}{C_{2,0}} \tag{A.16}$$

and

$$m_{1|\alpha} = \frac{C_{1,1}}{C_{2,0}} u_\alpha. \tag{A.17}$$

In this manner only ten velocity moments are necessary and transported:

$$\begin{bmatrix} m_{0,0} & m_{0,1} & m_{0,2} & m_{0,3} & m_{0,4} \\ m_{1,0} & m_{1,1} & & & \\ m_{2,0} & & & & \\ m_{3,0} & & & & \\ m_{4,0} & & & & \end{bmatrix}, \begin{bmatrix} C_{0,0} & C_{0,1} & C_{0,2} & C_{0,3} & C_{0,4} \\ C_{1,0} & C_{1,1} & & & \\ C_{2,0} & & & & \\ C_{3,0} & & & & \\ C_{4,0} & & & & \end{bmatrix} \tag{A.18}$$

in which $m_{1,1}$ and $C_{1,1}$ are used for calculating a_1 and after a_0, a_1 are calculated, the conditional moments/velocities $m_{1|\alpha}$ can be computed.

Eventually by expanding Eq. (A.9), we obtain:

$$
\begin{aligned}
C_{0,0} &= \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha = 1, \\
C_{1,0} &= \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha = 0, \\
C_{2,0} &= \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^2, \\
C_{3,0} &= \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^3, \\
C_{4,0} &= \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha^4.
\end{aligned}
\tag{A.19}
$$

It can be seen that the primary weights w_α and nodes u_α can be calculated from $C_{i,0}$ as discussed in the previous section (note that one of the node value is zero). Since $C_{1,2}$ and $C_{2,1}$ are not transported, the modified CHyQMOM close them by defining:

$$
C_{2,1} = \frac{C_{1,1} C_{3,0}}{C_{2,0}}, C_{1,2} = \frac{C_{1,1} C_{0,3}}{C_{0,2}}. \tag{A.20}
$$

Substituting Eq. (A.9) into Eq. (A.20) leads to:

$$
\sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta} = 0. \tag{A.21}
$$

By defining the conditional central moments as:

$$
C_{j|\alpha} = \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}^j, \tag{A.22}
$$

it can be seen that as long as $C_{j|\alpha}$ is known, $w_{\alpha\beta}$ and $v_{\alpha\beta}$ can be calculated. From the definition, we therefore have:

$$
C_{0|\alpha} = 1, \tag{A.23}
$$

$$
C_{1|\alpha} = 0. \tag{A.24}
$$

Meanwhile, according to Eq. (A.9), $C_{0,i}$ can be written as

$$
C_{0,0} = 1, \tag{A.25}
$$

$$C_{0,1} = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta}(m_{1|\alpha} + v_{\alpha\beta})$$
$$= \frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} \sum_{\beta=1}^{3} w_{\alpha\beta} + \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta} \right) = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} = 0. \quad \text{(A.26)}$$

Eq. (A.26) can be seen as a constrain condition. Moreover, we have:

$$C_{0,2} = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta}(m_{1|\alpha}^2 + v_{\alpha\beta}^2 + 2m_{1|\alpha} v_{\alpha\beta})$$
$$= \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 + \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}^2 = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 + \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha C_{2|\alpha} \quad \text{(A.27)}$$

$$C_{0,3} = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta}(m_{1|\alpha} + v_{\alpha\beta})^3$$
$$= \frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^3 + \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}^3 + 3 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}^2 \right)$$
$$= \frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^3 + \sum_{\alpha=1}^{3} w_\alpha C_{3|\alpha} + 3 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} C_{2|\alpha} \right) \quad \text{(A.28)}$$

$$C_{0,4} = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta}(m_{1|\alpha} + v_{\alpha\beta})^4 =$$
$$\frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^4 + \sum_{\alpha=1}^{3} w_\alpha \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}^4 + 6 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}^2 + 4 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} \sum_{\beta=1}^{3} w_{\alpha\beta} v_{\alpha\beta}^3 \right)$$
$$= \frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^4 + \sum_{\alpha=1}^{3} w_\alpha C_{4|\alpha} + 6 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 C_{2|\alpha} + 4 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} C_{3|\alpha} \right) \quad \text{(A.29)}$$

and:

$$C_{1,2} = \frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha u_\alpha m_{1|\alpha}^2 + \sum_{\alpha=1}^{3} w_\alpha u_\alpha C_{2|\alpha} \right). \quad \text{(A.30)}$$

In order to obtain $C_{2|\alpha}$, $C_{3|\alpha}$ and $C_{4|\alpha}$, define $C_{2|\alpha}$ as follows:

$$C_{2|\alpha} = C_{0,2} \left(b_0 + b_1 \frac{m_{1\alpha}}{\sqrt{C_{2,0}}} \right), \quad \text{(A.31)}$$

where b_0 and b_1 needs to be calculated. Substituting Eq. (A.31) and Eq. (A.26) to Eq. (A.27) leads

to

$$C_{0,2} = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 + \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha C_{0,2} \left(b_0 + b_1 \frac{m_{1\alpha}}{\sqrt{C_{2,0}}} \right) = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 + C_{0,2} b_0 \quad \text{(A.32)}$$

and

$$b_0 = \frac{C_{0,2} - \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 / m_{0,0}}{C_{0,2}}. \quad \text{(A.33)}$$

Meanwhile, since $\sum_{\alpha=1}^{3} w_\alpha u_\alpha = 0$, Eq. (A.30) can be written as

$$C_{1,2} = \frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha u_\alpha m_{1|\alpha}^2 + \sum_{\alpha=1}^{3} w_\alpha u_\alpha C_{0,2} \left(b_0 + b_1 \frac{u_\alpha}{\sqrt{C_{2,0}}} \right) \right) = \frac{1}{m_{00}} \sum_{\alpha=1}^{3} w_\alpha u_\alpha m_{1|\alpha}^2 + b_1 C_{0,2} C_{2,0}^{1/2}. \quad \text{(A.34)}$$

Therefore, b_1 can be calculated:

$$b_1 = \frac{C_{1,2} - \sum_{\alpha=1}^{3} w_\alpha u_\alpha m_{1|\alpha}^2 / m_{0,0}}{C_{0,2} C_{2,0}^{1/2}} = \frac{\frac{C_{1,1} C_{0,3}}{C_{0,2}} - \sum_{\alpha=1}^{3} w_\alpha u_\alpha m_{1|\alpha/m_{0,0}}^2}{C_{0,2} C_{2,0}^{1/2}}. \quad \text{(A.35)}$$

In order to calculate the secondary weights and nodes, $C_{3|\alpha}$ and $C_{4|\alpha}$ should be also calculated. However, Eq. (A.28) and Eq. (A.29) represent a non-linear system which is difficult to solve. In order to handle this problem, it is assumed that $C_{3|\alpha}$ and $C_{4|\alpha}$ depend on $C_{2|\alpha}$ as follows:

$$C_{3|\alpha} = q^* C_{2|\alpha}^{3/2}, C_{4|\alpha} = \eta^* C_{2|\alpha}^2. \quad \text{(A.36)}$$

Substituting Eq. (A.36) into Eq. (A.28) and Eq. (A.29), we have

$$C_{0,3} = \frac{1}{m_{00}} \left(\sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^3 + \sum_{\alpha=1}^{3} w_\alpha q^* C_{2|\alpha}^{3/2} + 3 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} C_{2|\alpha} \right) \quad \text{(A.37)}$$

$$q^* = \left(m_{0,0} C_{0,3} - \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^3 - 3 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} C_{2|\alpha} \right) / \sum_{\alpha=1}^{3} w_\alpha C_{2|\alpha}^{3/2} \quad \text{(A.38)}$$

$$C_{0,4} = \frac{1}{m_{0,0}} \left(\sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^4 + \sum_{\alpha=1}^{3} w_\alpha \eta^* C_{2|\alpha}^2 + 6 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 C_{2|\alpha} + 4 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} C_{3|\alpha} \right) \quad \text{(A.39)}$$

$$\eta^* = \left(C_{0,4} - \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^4 - 6 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha}^2 C_{2|\alpha} - 4 \sum_{\alpha=1}^{3} w_\alpha m_{1|\alpha} q^* C_{2|\alpha}^{3/2} \right) / \sum_{\alpha=1}^{3} w_\alpha C_{2|\alpha}^2 \quad \text{(A.40)}$$

After q^* and η^* are calculated, $C_{3|\alpha}$ and $C_{4|\alpha}$ can be updated and the secondary weights/nodes can be calculated according to Eq. (A.22).

It should be noted here that all the central moments cannot be negative. Assuming that $C_{0,2}$ is non-negative, b_0 should be non-negative. Meanwhile, b_1 must be limited to the value that makes the most negative $C_{2|\alpha} = 0$, then all the other conditional moments will be non-negative. In practice, we found that b_0 has a very small negative value (e.g., 1.00×10^{-15}) due to the random error. It implies that the correlation coefficient is larger than one, which is not physical. Therefore, we simply set b_0 equal to zero as if it were negative. In this way, all the conditional moments are ensured to be non-negative and the weights/nodes can be calculated.

Appendix B. Reconstructed velocity

Before we employ CHyQMOM to simulate gas-particle flows under one-way and two-way coupling, we firstly investigate the reconstructed velocities predicted by three-node CHyQMOM. The algorithm is tested on the following bivariate Gaussian distribution:

$$f(u,v) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(u-\mu_1)^2}{\sigma_1^2} - \frac{2\rho(u-\mu_1)(v-\mu_2)}{\sigma_1\sigma_2} + \frac{(v-\mu_2)^2}{\sigma_2^2}\right]\right), \quad \text{(B.1)}$$

where $\mu_1 = 10, \mu_2 = 20, \sigma_1 = \sigma_2 = 2, \rho = 0.375$. The moment-generating function, $M_{i,j}$, can be written as follows:

$$m_{k,l} = \exp\left(k\mu_1 + l\mu_2 + 0.5(\sigma_1^2 k^2 + 2\rho\sigma_1\sigma_2 kl + \sigma_2^2 l^2)\right), \quad \text{(B.2)}$$

resulting in the following mixed moments:

$$\begin{aligned}
m_{0,0} &= 1, \\
m_{1,0} &= 10, \\
m_{2,0} &= 104, \\
m_{3,0} &= 1120, \\
m_{4,0} &= 12448, \\
m_{0,1} &= 20, \\
m_{0,2} &= 404, \\
m_{0,3} &= 8240, \\
m_{0,3} &= 169648, \\
m_{1,1} &= 203.
\end{aligned} \quad \text{(B.3)}$$

The velocity quadrature nodes reconstructed by the three-node CHyQMOM are shown in Fig. B.13. The results obtained with the TP QMOM are also provided for comparison [40]. It should be noted here that when ρ changes, the mixed moments (e.g., $m_{1,1}$) also changes. However, the TP procedure only employs the pure moments to calculate the nodes. Therefore, the quadrature nodes predicted by TP QMOM cannot change no matter what the value of ρ is. TP can only adjust the weights to satisfy the moments. In certain cases (e.g., $\rho > 0.5$), negative weights would be predicted, resulting in turn in non-realizable moments. TP QMOM predicts exactly the same quadrature nodes, independently from the permutation, making it easier to treat two-way coupling problems. When CHyQMOM is employed, these nodes automatically move in the velocity space without rotating the reference frame. However, it can be found that the final quadrature nodes predicted by CHyQMOM depend on the specific permutations. When u is selected to calculate the primary nodes, the secondary nodes v shift automatically, and vice-versa.

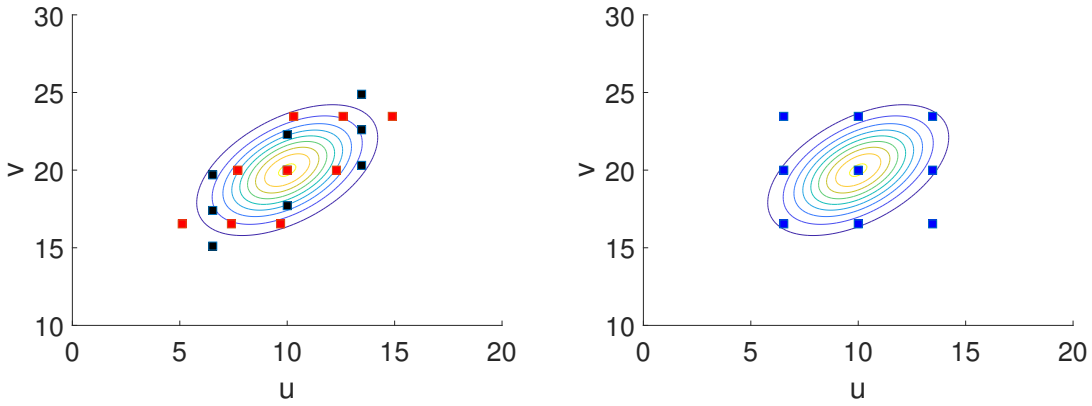
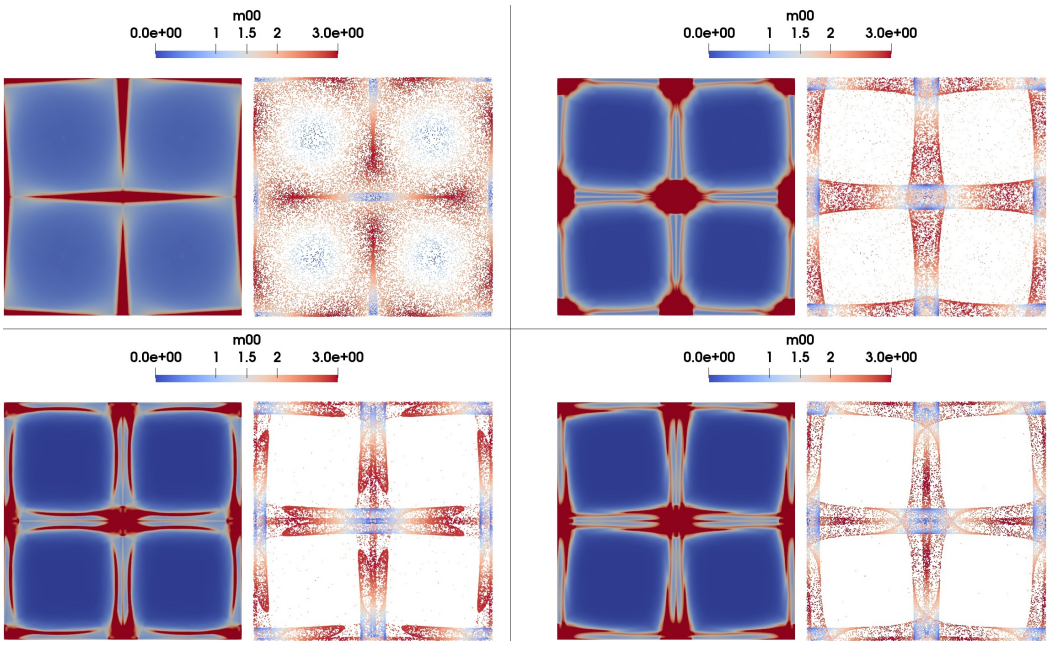


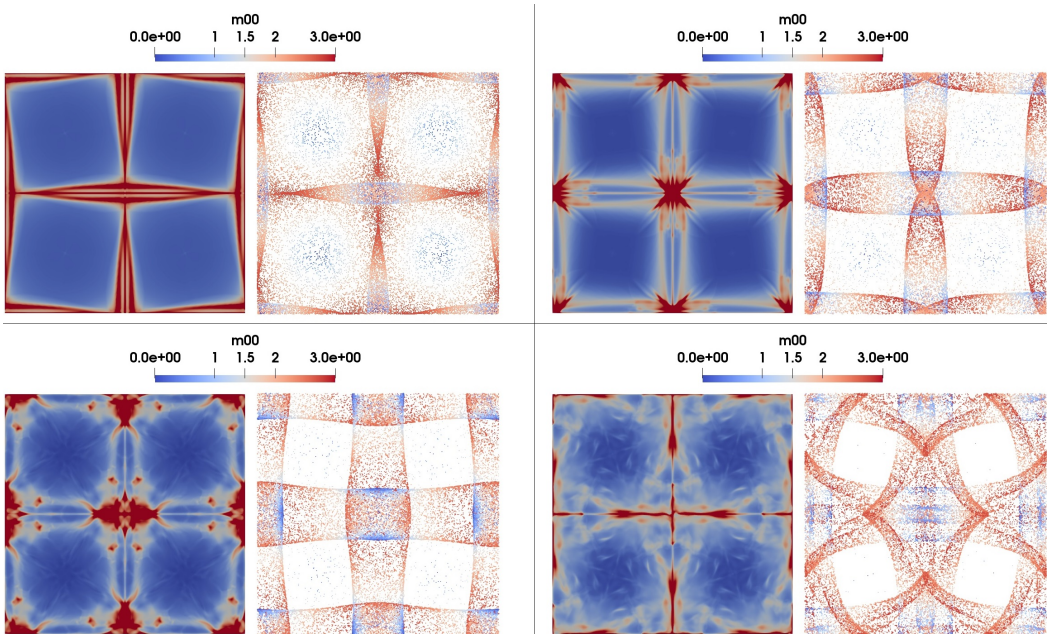
Figure B.13: Quadrature nodes predicted by three-node CHyQMOM (left, u conditioned on v: red squares, v conditioned on u: black squares) and TP QMOM (right, blue squares) for a bivariate Gaussian distribution. $\mu_1 = 10, \mu_2 = 20, \sigma_1 = \sigma_2 = 2, \rho = 0.375$.

Appendix C. Symmetric particle flows in TGV

Here we report the symmetric particle flows in TGV which were also reported in other works [19, 59, 51]. In our work we report and highlight the transient results. The difference of this test with that reported in the manuscript is that all the particles ($m_{0,0}$) are distributed uniformly at the beginning. Predictions of $m_{0,0}$ by CHyQMOM are reported in Fig. C.14. At the beginning, CHyQMOM predicts high accurate results compared with Lagrangian simulations since only small PTC events occur. As time elapses, it can be seen that for smaller Stokes number, CHyQMOM can still predict comparable results with Lagrangian simulations. For large Stokes number, CHyQMOM fails in predicting the particle strips as predicted in Lagrangian results. In both cases, near the center of the computational domain, CHyQMOM cannot capture the finer details as predicted by the Lagrangian method since multiple PTC occur. However, other moment methods can predict the stripe structure for large Stokes numbers [53, 51, 59]. This is due to the last moment which was fixed in CHyQMOM and fewer moments are controlled to achieve computational economic.



(a) Stokes number is 0.2



(b) Stokes number is 0.5

Figure C.14: Contour plots of $m_{0,0}$ predicted by three node CHyQMOM and Lagrangian method for Taylor-Green vortices at different time instants under different Stokes numbers.