Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2008) 6313–6350
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A quadrature-based third-order moment method for dilute gas-particle flows

R.O. Fox *

Department of Chemical and Biological Engineering, Iowa State University, 2114 Sweeney Hall, Ames, IA 50011-2230, USA

Received 14 October 2007; received in revised form 7 March 2008; accepted 10 March 2008
Available online 20 March 2008

Abstract

Dilute gas-particle flows can be described by a kinetic equation containing terms for spatial transport, gravity, fluid drag, and particle–particle collisions. However, the direct numerical solution of the kinetic equation is intractable for most applications due to the large number of independent variables. A useful alternative is to reformulate the problem in terms of the moments of the velocity distribution function. Closure of the moment equations is challenging for flows away from the equilibrium (Maxwellian) limit. In this work, a quadrature-based third-order moment closure is derived that can be applied to gas-particle flows at any Knudsen number. A key component of quadrature-based closures is the moment-inversion algorithm used to find the weights and abscissas. A robust inversion procedure is proposed for moments up to third order, and tested for three example applications (Riemann shock problem, impinging jets, and vertical channel flow). Extension of the moment-inversion algorithm to fifth (or higher) order is possible, but left to future work. The spatial fluxes in the moment equations are treated using a kinetic description and hence a gradient-diffusion model is not used to close the fluxes. Because the quadrature-based moment method employs the moment transport equations directly instead of a discretized form of the Boltzmann equation, the mass, momentum and energy are conserved for arbitrary Knudsen number (including the Euler limit). While developed here for dilute gas-particle flows, quadrature-based moment methods can, in principle, be applied to any application that can be modeled by a kinetic equation (e.g., thermal and non-isothermal flows currently treated using lattice Boltzmann methods), and examples are given from the literature.

Keywords: Quadrature method of moments; Velocity distribution function; Kinetic equation; Gas-particle flows; Boltzmann equation

1. Introduction

The numerical simulation of gas-particle flows is complicated by the wide range of phenomena that can occur in real applications [3,11,20,23–25,37,57,60,61,63,69]. In the absence of the gas phase, the particles behave as a granular flow. In the dilute limit, a granular flow is dominated by binary collisions and can be described by a kinetic equation [14] (i.e., it is a granular gas). On the other hand, in the dense limit sustained

* Tel.: +1 515 294 9104; fax: +1 515 294 2689.
E-mail address: rofox@iastate.edu

0021-9991/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2008.03.014

particle–particle contacts are dominant and the (local) kinetic description breaks down. As in rarefied gas dynamics [13], the dilute limit can be parameterized by a dimensionless number (Knudsen number, Kn) that represents the importance of particle–particle collisions relative to free transport. In the limit of small Knudsen number, collisions are dominant and the particle velocity distribution function is very near the equilibrium (Maxwellian) distribution. In the equilibrium limit, it is thus possible to describe a dilute granular flow by velocity moments up to second order (the so-called hydrodynamic limit [38]). For larger Knudsen number, the equilibrium distribution is no longer a good approximate and, eventually, one must solve the Boltzmann kinetic equation to adequately capture the flow physics of a non-equilibrium granular gas (as is done for rarefied gases [4,13,52]). For wall-bounded granular gases, non-equilibrium effects can be important near the walls even at small Knudsen numbers because the velocity distribution function is composed of incoming and outgoing particles with velocities far from equilibrium (e.g. a bimodal distribution can be observed near the walls).

Adding the fluid phase introduces new physics, the most important of which is the fluid-particle drag term. For an isolated particle in a uniform fluid, the particle Reynolds number (Re_p) determines the net force of the fluid on the particle. For gas-particle flows with moderately large particles, we usually have ($Re_p \gg 1$) and the drag force is parameterized by a drag coefficient [58]. Another important parameter in fluid-particle flows is the Stokes number (St), which is the ratio of the characteristic response time of the particle to the characteristic time scale of the fluid flow. For example, in a particle-laden impinging-jet flow the value of the Stokes number will determine whether or not the particles cross the impingement plane. If the Stokes number is sufficiently small ($St \ll 1$), the particles will not cross the plane and will have nearly the same velocity as the fluid. On the other hand, for very large Stokes numbers, the particles barely "feel" the fluid and thus will continue directly through the impingement plane. Note that if the flow is dilute, particles can readily cross the plane without risk of collision with other particles, i.e., particle trajectory crossing (PTC) will occur. Locally (i.e., at the impingement plane) the velocity distribution function will be bimodal (even when particles do not collide) with values corresponding to the velocities originating from each side of the plane. More generally, particles with finite Stokes number will tend to accumulate in regions of the fluid flow with small vorticity [60]. In turbulent flows, such regions correspond to local (time-dependent) impingement zones [19]. Because of the complex particle physics in such regions, which lead to particle velocity distributions far from Maxwellian, equilibrium closures for the moment equations will fail. For example, an equilibrium closure for the velocity moments will predict delta-shocks [6] in the local number density when, in fact, no such behavior is present in the flow [20]. In order to avoid such unphysical behavior, it is necessary to extend the moment closure to handle PTC, which is ubiquitous in turbulent gas-particle flows.

The range of Knudsen, Reynolds, and Stokes numbers observed in real gas-particle flows is very wide, and hence it is difficult to find a computational model that will work for all applications. Moreover, even in a given flow, the ranges of these numbers may be large due, for example, to local dense and dilute regions or to a wide range of gas-phase velocities. In general, a model for dilute gas-particle flow must include both the kinetic description for the particle phase and a coupled momentum balance equation for the gas phase (i.e., two-way coupling). Nevertheless, the principal modeling challenge is the treatment of the kinetic equation and, therefore, we will consider only one-way coupling (i.e., the fluid velocity is given) in this work. Furthermore, we will consider examples where the Stokes number is large enough that velocity fluctuations (i.e., due to gas-phase turbulence) can be neglected [57]. These assumptions will allow us to isolate the critical role of non-equilibrium effects in moment closures, and to investigate their treatment using quadrature-based moment closures. We should note, however, that because the coupling with the fluid phase occurs at the level of the moments, the conclusions drawn in this work have direct implications on the closures used in fully two-way coupled moment methods (e.g., found using the kinetic theory of granular flow [3]).

The quadrature method of moments (QMOM) was introduce by McGraw [49] as a closure for population balance equations (PBE) involving a number density function (NDF) $f(\xi, t)$ with independent variable ξ representing the particle volume or mass. For example, in an aerosol ξ might correspond to the droplet volume, which evolves in time due to physical processes such as condensation, evaporation, aggregation, and breakup. The PBE that governs the NDF is then an integro-differential equation that is difficult to solve numerically. In practice, the principal quantities of interest for comparison with data are the moments of the NDF defined by

$$m_k(t) = \int_0^\infty \xi^k f(\xi, t)\,\mathrm{d}\xi.$$

Thus, a closure of the PBE that can accurately predict the lower-order moments is often sufficient. In most cases, however, the moment equations derived from the PBE are not closed (see [43] for a discussion of moment closures) and a closure step is required. Using QMOM, the moments are expressed in terms of a finite set of N weights $n_\alpha(t)$ and N abscissas $\xi_\alpha(t)$, defined such that [49]

$$m_k = \sum_{\alpha=1}^{N} n_\alpha \xi_\alpha^k \quad \text{for } k = 0, 1, \ldots, 2N-1.$$

For higher-order moments (i.e. $k \geqslant 2N$) and non-integer moments, the quadrature weights and abscissas provide a consistent closure. For example, let $g(\xi)$ be a smooth function of ξ. The QMOM approximation of the moment involving g is then

$$\int_0^\infty g(\xi) f(\xi)\,\mathrm{d}\xi \approx \sum_{\alpha=1}^{N} n_\alpha g(\xi_\alpha).$$

In a similar manner, all of the terms in the moment equations for m_k with $k = 0, 1, \ldots, 2N-1$ can be closed, resulting in a closed system of $2N$ equations.

A key component of QMOM is the inversion algorithm used to find the weights and abscissas from the moments. A direct nonlinear solve is poorly conditioned (and gets worse for increasing N); however, the product-difference (PD) algorithm introduced by McGraw [49] overcomes this difficulty by replacing the nonlinear solve with a computationally efficient eigenvalue–eigenvector problem that is well conditioned even for large N. Using the PD algorithm, it is possible to solve for a spatially dependent NDF using standard flow codes to transport the moments [26,47,48,65,71]. Moreover, even with highly coupled and nonlinear aggregation kernels, good accuracy for the lower-order moments can usually be attained with $N = 4$ nodes [48].

Unfortunately, the PD algorithm only works for uni-variate distribution functions, but not for bi-variate (or higher) NDF where $f(\xi, \eta)$ depends on two (or more) internal coordinates. In order to overcome this limitation, one can work directly with transport equations the weights and abscissas (direct QMOM [46] or DQMOM), which works well when ξ and η represent passive scalars (e.g. droplet volume and surface area [27]). The DQMOM approach has also been applied to polydisperse gas–solid flows [23] to describe the fluid dynamics in the presence of multiple particle sizes. However, in that application the particle velocity is assumed to be in the equilibrium (Maxwellian) limit so that only the mean particle velocity (albeit conditioned on particle size) suffices to describe the flow dynamics. For non-equilibrium gas–solid flows (such as those considered in this work), the velocity distribution function can deviate strongly from the Maxwellian form, and higher-order velocity moments are required to close the transport equations. In principle, DQMOM can be used with the velocity moments to rewrite the moment transport equations in terms of transport equations for the weights and abscissas [29]. In the absence of collisions, the resulting transport equations for the velocity abscissas [29] have the form of the pressure-less gas dynamics equation [6], and each abscissa evolves independently of the others. Although the DQMOM formulation for velocity works well in most of the flow domain, it fails at "singular" points where the velocity abscissas change discontinuously [19] (e.g., the PTC points). Remarkably, a quadrature-based moment closure for the velocity distribution function is robust at such points [19,20]. However, for multi-dimensional velocity distributions (e.g. three-dimensional phase space), the key challenge of inverting the moments to find the weights and velocity abscissa remains.

The kinetic equation for the velocity distribution function is used in many applications besides gas-particle flows [4,9,12,13,16,17,33–35,39,44,45,52,55,56,67], and thus there have been many computational methods developed to find approximate solutions. In the context of quadrature methods, the most closely related techniques are those based on "multiphase" or "multi-branch" solutions to the kinetic equation (i.e. without the collision term in the Boltzmann equation) [9,22,32,33,39,40,44,45,55,67]. (See [56] for a recent review.) In fact, the quadrature formula used in moment-based methods [56] for these applications to relate the weights and abscissas to the moments is the same as in QMOM; however, in the literature on multiphase solutions a direct nonlinear solver is used (and is therefore limited to small N) instead of the PD algorithm. A more significant

difference between QMOM and multiphase solutions is that the latter are designed to yield *exact* solutions to the kinetic equation *without* collisions, while the former provides *approximate* solutions to the kinetic equation *with* collisions.[1] Despite these differences, the difficulty of representing multiple velocities generated by the spatial transport term is shared by many applications of the kinetic equation.

At present, there are two classes of methods that can be used to find accurate solutions to the kinetic equation: (i) direct solvers that discretize velocity phase space [4,10,12,31,51,52]; (ii) Lagrangian (or ray-tracing) methods [2,6,21,56,61,69]. However, the computational cost of using either of these methods in many applications (such as fluid-particle flows) is prohibitive. Moreover, in most applications we are not interested in knowing the exact form of the velocity distribution function, rather knowledge of its lower-order moments is sufficient. For these reasons, there is considerable motivation to develop *predictive* moment closures whose accuracy can be improved in a rational manner. Quadrature-based moment closures fall into this category because, in principle, the accuracy of these closures can be improved by increasing the number of quadrature nodes.[2] Nevertheless, as mentioned previously, a key technical challenge with quadrature-based moment closures is the development of efficient moment-inversion algorithms for multidimensional velocity distribution functions. We should note that because the weights are non-negative and the abscissas are located in velocity phase space, a quadrature-based moment method provides an *adaptive*[3] discretization of velocity phase space that is consistent with the underlying moments. Compared to direct solvers, this discretization is very sparse (equal to the number of nodes). An important open question is thus to determine the range of accuracy that can be achieved using quadrature in comparison to direct solvers. In this work, we address the problem of finding a moment-inversion method for velocity moments up to third order, and leave the question of increased accuracy using more nodes to future work.

The remainder of the paper is organized as follows. In Section 2 we introduce the kinetic equation for dilute gas-particle flows. In Section 3 we present the moment transport equations up to third order derived from the kinetic equation. Section 4 describes in detail the moment-inversion method for the velocity moments for one-, two-, and three-dimensional velocity phase space (using 2, 4, and 8 nodes, respectively) and compares the quadrature formulas to other methods available in the literature. In Section 5 we present the kinetic-based numerical algorithm used to solve the moment equations. Section 6 is devoted to example applications to test the numerical implementation. Finally, in Section 7 conclusions are drawn and the key characteristics of the proposed moment method and numerical algorithm are discussed.

2. Kinetic description of dilute gas-particles flows

Consider the following kinetic equation for the velocity distribution function $f(\mathbf{v};\mathbf{x},t)$ of dilute monodisperse solid particles in a gaseous flow [57]:

$$\partial_t f + \mathbf{v}\cdot\partial_{\mathbf{x}} f + \partial_{\mathbf{v}}\cdot(f\mathbf{g}) + \partial_{\mathbf{v}}\cdot\left(f\frac{\mathbf{F}}{m_{\mathrm{p}}}\right) = \mathbb{C}, \tag{1}$$

where $\mathbf{v}$ is the particle velocity vector, $\mathbf{g}$ is the gravity force, $\mathbf{F}$ is the drag force from the gas phase acting on a particle, m_{p} is the particle mass, and $\mathbb{C}$ is the particle–particle collision term. In this work, we will assume that the collision term is closed. For example, using the Bhatnagar–Gross–Krook (BGK) approximation [5], the collision term becomes

$$\mathbb{C} = \frac{1}{\tau}(f_{\mathrm{eq}} - f), \tag{2}$$

[1] More generally, multiphase solutions do not consider interactions between different branches or cases where the velocity distribution function is continuous in velocity (e.g. Maxwellian). Mathematically, the form of the kinetic equation in these applications is at most first-order derivatives in the independent variables [56]. Collisions and other non-local interaction terms (which are not first order) couple the velocity abscissas in a non-trivial manner [29,46].

[2] While this has been clearly demonstrated for "passive" moments (e.g. volume) [27,48,49], it remains to be shown for the velocity moments.

[3] The discretization is adaptive because it changes with the moments.

where τ is a collision time constant and f_{eq} is the equilibrium (Maxwellian) distribution. In d dimensions, f_{eq} is given by

$$f_{eq}(\mathbf{v}) = \frac{M^0}{(2\pi\sigma_{eq})^{d/2}} \exp\left(-\frac{|\mathbf{v}-\mathbf{U}_p|^2}{2\sigma_{eq}}\right), \tag{3}$$

where $M^0 = \int f \, d\mathbf{v}$ is the particle number density (zero-order moment), $\mathbf{U}_p$ and σ_{eq} are the mean particle velocity and equilibrium variance, respectively. For real particles, collisions are usually inelastic (i.e. σ_{eq} is not conserved). However, in this work, we will treat σ_{eq} as a conserved quantity.[4] Note that the Knudsen number is proportional to the collision time τ, so that the velocity distribution function is equal to f_{eq} when $Kn = 0$. In the opposite limit, the particles are collision-less and the velocity distribution function will be determined by the remaining terms in Eq. (1) (and hence can be very far from equilibrium).

For dilute gas-particle flows, the drag force can be approximated by

$$\mathbf{F}(\mathbf{U}_f, \mathbf{v}) = \frac{3 m_p \rho_f}{4 d_p \rho_p} C_d |\mathbf{U}_f - \mathbf{v}| (\mathbf{U}_f - \mathbf{v}), \tag{4}$$

where $\mathbf{U}_f(\mathbf{x}, t)$ is the fluid velocity (assumed to be known in this work), d_p is the particle diameter, ρ_f the fluid density, ρ_p the particle density, and C_d the particle drag coefficient given by the following correlation due to [58]:

$$C_d = \frac{24}{Re_p}(1 + 0.15 Re_p^{0.687}) \tag{5}$$

with $Re_p = |\mathbf{U}_f - \mathbf{v}| d_p / \nu_f$ being the particle Reynolds number where ν_f is the fluid kinematic viscosity. In some cases, we will also use the simpler Stokes drag coefficient: $C_d = 24/Re_p$. The drag term can then be expressed as

$$\mathbf{F}(\mathbf{U}_f, \mathbf{v}) = \frac{m_p}{\tau_p}(\mathbf{U}_f - \mathbf{v}), \tag{6}$$

where $\tau_p = \rho_p d_p^2 / (18 \mu_f)$. The particle Stokes number can then be defined as $St = \tau_p / \tau_f$ where τ_f is the characteristic time for the fluid velocity $\mathbf{U}_f$. For very small St the particles will follow the fluid ($\mathbf{v} \approx \mathbf{U}_f$) and hence they will not collide in dilute flows. Moreover, in this limit, flow-induced particle segregation is not observed in turbulent flows. On the other hand, for very large St the particles respond very slowly to changes in the fluid velocity and act as a granular gas. In between, for $St \approx 1$ flow-induced segregation in turbulent flow is significant and has a strong effect on collision statistics [24,60]. In any case, the Stokes number has a strong effect on the velocity distribution function, which can be described by Eq. (1), and makes the behavior of gas-particle flows significantly different than molecular gases (i.e. $St = \infty$). Note that drag can be a source or a sink for σ_{eq}, depending on how the particles respond to fluid velocity fluctuations. However, at large St the drag will lead to dissipation of σ_{eq} because fluid velocity fluctuations will have no effect on the particle velocity fluctuations [57].

In practice, due to the complexity of the kinetic description, the velocity distribution function must be found using a Lagrangian particle tracking code (or DSMC) coupled to a turbulent flow solver for $\mathbf{U}_f$. Although accurate, this solution method is rather expensive and subject to statistical errors due to finite sample sizes. Our interest here is to approximate solutions to Eq. (1) for dilute gas-particle flows [11,24,25,60,63,69], such as the vertical channel flows described in [37,57,61], using Eulerian moment methods. However, the numerical method developed in this work can, in principle, be used for other physical situations that can be described by Eq. (1) (e.g., dilute sprays [66]). For example, neglecting the terms involving the gravity and drag force, Eq. (1) is the simplest version of the Boltzmann equation, for which there exists an extensive literature on numerical methods for its solution in various limiting cases [6,13], including small-scale gaseous hydrodynamics [35] and the semiconductor Boltzmann–Poisson system [12]. We will thus use the Riemann shock problem from gas dynamics [4,16,52,62] as one of the test cases in Section 6. Another important

[4] Inelastic collisions can be introduced using a restitution coefficient $0 \leqslant e \leqslant 1$ [38]. Then, in the moment equations, one simply replaces σ_{eq} with $e^2\sigma_{eq}$. For example, the decay rate for σ_{eq} is then equal to $(e^2 - 1)\sigma_{eq}$. Isothermal flows can be treated by setting σ_{eq} to a constant value.

test case for dilute gas-particle flows is impinging particle jets [19,20] with or without collisions, which is a model for particle trajectory crossing (PTC) at finite Stokes numbers. For this example, the velocity distribution function is locally bimodal and hence far removed from f_{eq}. Moment methods developed for flows near equilibrium (i.e. collision dominated or small Kn) cannot be used to describe PTC because they cannot handle locally multi-valued velocities. However, quadrature-based moment closures perform well for this case in the collision-less limit [19,20]. As noted in the Introduction, the kinetic equation (i.e. without collisions) also appears in other applications such as geometric optics [56].

3. Moment transport equations

In this work we develop a third-order moment closure using quadrature. In three dimensions, the 20 moments up to third order will be denoted by

$$W^3 = (M^0, M^1_1, M^1_2, M^1_3, M^2_{11}, M^2_{12}, M^2_{13}, M^2_{22}, M^2_{23}, M^2_{33}, M^3_{111}, M^3_{112}, M^3_{113}, M^3_{122}, M^3_{123}, M^3_{133}, M^3_{222}, M^3_{223}, M^3_{233}, M^3_{333}).$$

The moments are found from the velocity distribution function by integration:

$$\begin{aligned} M^0 &= \int f \,\mathrm{d}\mathbf{v}, \quad M^1_i = \int v_i f \,\mathrm{d}\mathbf{v}, \\ M^2_{ij} &= \int v_i v_j f \,\mathrm{d}\mathbf{v}, \quad M^3_{ijk} = \int v_i v_j v_k f \,\mathrm{d}\mathbf{v}. \end{aligned} \tag{7}$$

Thus the transport equations for the moments can be easily found starting from Eq. (1) (repeated indices imply summation):

$$\begin{aligned} &\frac{\partial M^0}{\partial t} + \frac{\partial M^1_i}{\partial x_i} = 0, \\ &\frac{\partial M^1_i}{\partial t} + \frac{\partial M^2_{ij}}{\partial x_j} = g_i M^0 + D^1_i, \\ &\frac{\partial M^2_{ij}}{\partial t} + \frac{\partial M^3_{ijk}}{\partial x_k} = g_i M^1_j + g_j M^1_i + \frac{M^0}{\tau}(\sigma_{eq}\delta_{ij} - \sigma_{ij}) + D^2_{ij}, \\ &\frac{\partial M^3_{ijk}}{\partial t} + \frac{\partial M^4_{ijkl}}{\partial x_l} = g_i M^2_{jk} + g_j M^2_{ik} + g_k M^2_{ij} + \frac{1}{\tau}(\Delta_{ijk} - M^3_{ijk}) + D^3_{ijk}, \end{aligned} \tag{8}$$

where the final terms on the right-hand sides are due to drag.

The moments of the equilibrium distribution are defined by

$$\Delta_{ijk} = M^0(\delta_{ij}U_{pk} + \delta_{ik}U_{pj} + \delta_{jk}U_{pi})\sigma_{eq} + M^0 U_{pi}U_{pj}U_{pk}, \tag{9}$$

where the mean particle velocity is $U_{pi} = M^1_i/M^0$. Note that the collision term only affects directly the second- and third-order moments, moving them towards the equilibrium values. Nevertheless, the collisions affect all moments indirectly through the transport terms. The unclosed terms in the moment transport equations (M^4_{ijkl}, D^1_i, D^2_{ij} and D^3_{ijk}) are closed using quadrature as described in Section 4.

Of particular importance is the treatment of the spatial fluxes, which are represented by a kinetic description [8,18,53,54]. For example, the flux term for the zero-order moment is separated into two contributions:

$$M^1_i = Q^-_i + Q^+_i = \int_{-\infty}^{0} v_i \left(\int f(\mathbf{v}) \,\mathrm{d}v_j \,\mathrm{d}v_k \right) \mathrm{d}v_i + \int_0^{\infty} v_i \left(\int f(\mathbf{v}) \,\mathrm{d}v_j \,\mathrm{d}v_k \right) \mathrm{d}v_i. \tag{10}$$

The fluxes for the higher-order moments are constructed in an analogous manner. The quadrature-based closure of the right-hand side of Eq. (10) is discussed in Section 4.2.

4. Quadrature-based third-order moment closure

As noted in the Introduction, quadrature-based moment methods distinguish themselves from other moment methods by the use of quadrature weights and abscissas to model unclosed terms in the moment transport equations. Thus, when developing a quadrature method, an important task is to define the algorithm for computing the weights and abscissas from the moments. In one-dimensional systems, this is easily accomplished using the PD algorithm [49]. The extension to higher dimensions is less straightforward and is the main objective of this section. Here we limit ourselves to quadrature formulas for moments up to third order, and use one-dimensional product formulas [59]. Thus, the number of quadrature nodes in each direction of velocity phase space will be two.

4.1. Relationship between moments and quadrature nodes

Let $d \in (1,2,3)$ denote the number of phase-space dimensions, and define $\beta = 2^d$. Let $V_\beta = [(n_\alpha, \mathbf{U}_\alpha)]$ with $\alpha \in (1,\ldots,\beta)$ denote the set of weights and abscissas for the β-node quadrature approximation of f. Note that the set of quadrature nodes V_β contains $(1+d)\beta$ unknowns (i.e. β weights, and β d-component velocity vectors). To find the components of V_β, we work with the velocity moments up to third order (for $i \leqslant j \leqslant k \in 1,\ldots,d$), which are related to the quadrature weights and abscissas by

$$\begin{aligned} M^0 &= \sum_{\alpha=1}^{\beta} n_\alpha, \quad M_i^1 = \sum_{\alpha=1}^{\beta} n_\alpha U_{\alpha i}, \\ M_{ij}^2 &= \sum_{\alpha=1}^{\beta} n_\alpha U_{\alpha i} U_{\alpha j}, \quad M_{ijk}^3 = \sum_{\alpha=1}^{\beta} n_\alpha U_{\alpha i} U_{\alpha j} U_{\alpha k}. \end{aligned} \tag{11}$$

In the following, we will let W^d denote the set of third-order moments in d dimensions. The principal objective of this section is to describe an algorithm for finding V_β from W^d. The inverse operation (finding W^d from V_β) is Eq. (11), which we will refer to as *projection* for reasons that will become apparent later. In general, it will not be possible to represent all possible moment sets in W^d using weights and abscissas in V_β. We will therefore define the set of representable moments as $W^{d\dagger} \subset W^d$.

4.2. Quadrature-based closure of the moment transport equations

The moment transport equations derived in Section 3 contain unclosed terms. Using quadrature, these terms can be expressed in terms of the weights and abscissas:

$$D_i^1 = \sum_{\alpha=1}^{\beta} \frac{n_\alpha}{m_\mathrm{p}} F_{i\alpha}, \tag{12}$$

$$D_{ij}^2 = \sum_{\alpha=1}^{\beta} \frac{n_\alpha}{m_\mathrm{p}} (U_{i\alpha} F_{j\alpha} + F_{i\alpha} U_{j\alpha}), \tag{13}$$

$$D_{ijk}^3 = \sum_{\alpha=1}^{\beta} \frac{n_\alpha}{m_\mathrm{p}} (U_{i\alpha} U_{j\alpha} F_{k\alpha} + F_{i\alpha} U_{j\alpha} U_{k\alpha} + U_{i\alpha} F_{j\alpha} U_{k\alpha}), \tag{14}$$

$$M_{ijkl}^4 = \sum_{\alpha=1}^{\beta} n_\alpha U_{i\alpha} U_{j\alpha} U_{k\alpha} U_{l\alpha}, \tag{15}$$

where

$$\mathbf{F}_\alpha = \mathbf{F}(\mathbf{U}_\mathrm{f}, \mathbf{U}_\alpha) = \frac{m_\mathrm{p}}{\tau_\alpha} (\mathbf{U}_\mathrm{f} - \mathbf{U}_\alpha) \tag{16}$$

and τ_α is the characteristic drag time for node α.

Quadrature is also used to write the spatial fluxes in terms of the weights and abscissas [19,20]. The fluxes are based on the kinetic description given in Eq. (10) using a delta-function representation of the distribution function:

$$f(\mathbf{v}) = \sum_{\alpha=1}^{\beta} n_\alpha \delta(\mathbf{v} - \mathbf{U}_\alpha). \tag{17}$$

For example, the negative and positive contributions to the flux terms for the zero-order moment are expressed as

$$Q_i^- = \sum_{\alpha=1}^{\beta} n_\alpha \min(0, U_{i\alpha}) \quad \text{and} \quad Q_i^+ = \sum_{\alpha=1}^{\beta} n_\alpha \max(0, U_{i\alpha}). \tag{18}$$

Likewise, the fluxes for higher-order moments have analogous forms [19,20]. This flux definition has important ramifications on the numerical algorithm used to solve the moment transport equations. For one thing, it is obvious that the fluxes are tightly coupled to the quadrature algorithm because the latter determines the abscissas. In particular, the fluxes will vary smoothly in space only if the abscissas are smoothly varying. It can also be noted that because the fluxes are written in terms of a finite set of velocities, the closed model equations do not contain gradient-diffusion terms as is usually the case in other moment methods. Moreover, because of the form of the fluxes, there is no advantage gained by rewriting the transport equations in terms of the central velocity moments (i.e. moments of the fluctuating velocity).

Finally, we should note that fluxes as defined above are not guaranteed to produce moments that can be represented by the proposed quadrature algorithm (i.e. the moments lie in W^d but not necessarily in $W^{d\dagger}$, and only the latter can be represented by the weights and abscissas). For this reason, after advancing the moments due to the fluxes (or any other process that does not remain in W^d), it is necessary to project the moments back into $W^{d\dagger}$. This is accomplished simply by using the moments to compute the weights and abscissas, and then using Eq. (11) to recompute the moments. We will refer to this operation as a *projection step*.

Our principal objective is to show that if

$$\text{(i)} \qquad M^0 > 0 \quad \text{and} \quad \text{(ii)} \qquad \text{for all } i \in (1, \ldots, d) \quad M_{ii}^2 > (M_i^1)^2 / M^0,$$

then V_β can be found from W^d in a well-defined manner. For clarity, we will first describe the one-dimensional case with $d = 1$ and two nodes: $\beta = 2$. The higher-dimensional cases ($\beta = 4$ and 8) are constructed in a similar manner, and are discussed later.

4.3. Two-node quadrature: V_2 from W^1

For the one-dimensional case, we begin by defining the mean particle velocity:

$$U_\mathrm{p} = M_1^1 / M^0, \tag{19}$$

and the velocity variance:

$$\sigma_U = \sigma^2 = M_{11}^2 / M^0 - U_{\mathrm{p}1}^2, \tag{20}$$

so that $\sigma_\mathrm{eq} = \sigma^2$. With a two-node quadrature approximation, we introduce a new variable X defined by a linear transformation of v_1[5]:

$$X = (v_1 - U_\mathrm{p})/\sigma \quad \text{so that } v_1 = \sigma X + U_\mathrm{p}. \tag{21}$$

Note that X is a scalar representing the normalized distance (in velocity phase space) from the mean velocity in the direction of the velocity fluctuations.

Denoting the four normalized moments with respect to X up to third order by

$$W^{1*} = (m^0, m^1, m^2, m^3),$$

[5] Introducing X when $d = 1$ is superfluous. However, for consistency with higher dimensions, we do so anyway.

it is straightforward to show that these moments are related to the velocity moments by

$$m^0 = 1, \quad m^1 = 0, \quad m^2 = 1, \quad m^3 = h(U_p, \sigma, M_1^3/M^0), \tag{22}$$

where h denotes a function of its arguments. This function is rather long (see Appendix A), but can easily be written out starting from the definition of X in terms of v_1 given in Eq. (21). Likewise, the weights and abscissas in X phase space, denoted by $V_2^* = [(n_\alpha^*, X_\alpha), \alpha \in (1,2)]$, are related to V_2 by

$$n_\alpha = M^0 n_\alpha^* \quad \text{and} \quad U_\alpha = \sigma X_\alpha + U_p. \tag{23}$$

Thus, we can compute V_2^* in terms of W^{1*}, and then use Eq. (23) to find V_2.

The moment set W^{1*} is related to V_2^* by

$$m^0 = \sum_{\alpha=1}^{2} n_\alpha^*, \quad m^1 = \sum_{\alpha=1}^{2} n_\alpha^* X_\alpha, \quad m^2 = \sum_{\alpha=1}^{2} n_\alpha^* X_\alpha^2, \quad m^3 = \sum_{\alpha=1}^{2} n_\alpha^* X_\alpha^3. \tag{24}$$

Note that W^{1*} contains four moments. Thus, if W^{1*} is known, we can invert Eq. (24) to find the four unknowns in V_2^*, from which we can compute V_2 using Eq. (23). Using two-node quadrature [49], the moments of X can be easily inverted to find V_2^*:

$$\begin{aligned} n_1^* &= 0.5 + \gamma, \quad X_1 = -\left(\frac{1-2\gamma}{1+2\gamma}\right)^{1/2}, \\ n_2^* &= 0.5 - \gamma, \quad X_2 = \left(\frac{1+2\gamma}{1-2\gamma}\right)^{1/2}, \end{aligned} \tag{25}$$

where $(-1/2 < \gamma < 1/2)$

$$\gamma = \frac{m^3/2}{\left[(m^3)^2 + 4\right]^{1/2}}. \tag{26}$$

It is then possible to compute W^1, defined by

$$W^1 = (M^0, M_1^1, M_{11}^2, M_{111}^3),$$

using V_2 in Eq. (11). These operations can be expressed as

$$W^1 \leftrightarrow W^{1*} \leftrightarrow V_2^* \leftrightarrow V_2 \leftrightarrow W^1.$$

Note that in the case of $d = 1$ all four moments in W^1 can be exactly reproduced by V_2^*. This will not be the case for higher dimensions.

4.4. Four-node quadrature: V_4 from W^2

In two dimensions, the set of 10 moments up to third order is defined by

$$W^2 = (M^0, M_1^1, M_2^1, M_{11}^2, M_{12}^2, M_{22}^2, M_{111}^3, M_{112}^3, M_{122}^3, M_{222}^3),$$

and we seek to define a four-node quadrature based on these moments. Again we begin by defining the mean particle velocity vector:

$$\mathbf{U}_p = \begin{bmatrix} M_1^1/M^0 \\ M_2^1/M^0 \end{bmatrix}, \tag{27}$$

and the velocity covariance matrix:

$$\sigma_U = [\sigma_{ij}] = \begin{bmatrix} M_{11}^2/M^0 - U_{p1}^2 & M_{12}^2/M^0 - U_{p1}U_{p2} \\ M_{12}^2/M^0 - U_{p1}U_{p2} & M_{22}^2/M^0 - U_{p2}^2 \end{bmatrix}, \tag{28}$$

so that $\sigma_{eq} = (\sigma_{11} + \sigma_{22})/2$.

The next step is to introduce a linear transformation $\mathbf{A}$ to diagonalize σ_U.[6] The choice of the linear transformation is not unique,[7] but for reasons that will be described later, we choose to use the Cholesky decomposition defined such that $\mathbf{L}^T\mathbf{L} = \sigma_U$ and $\mathbf{L}$ is upper triangular:

$$\mathbf{L} = \begin{bmatrix} \sigma_{11}^{1/2} & \sigma_{12}/\sigma_{11}^{1/2} \\ 0 & (\sigma_{22} - \sigma_{12}^2/\sigma_{11})^{1/2} \end{bmatrix}. \tag{29}$$

With this choice we set $\mathbf{A} = \mathbf{L}^T$ and introduce a two-component vector $\mathbf{X} = [X_1 X_2]^T$ defined by

$$\mathbf{X} = \mathbf{A}^{-1}(\mathbf{v} - \mathbf{U}_p) \quad \text{so that } \mathbf{v} = \mathbf{A}\mathbf{X} + \mathbf{U}_p. \tag{30}$$

If we denote the first four moments of X_i by $m_i^k, k \in (0,1,2,3)$; then they are related to the velocity moments by

$$m_i^0 = 1, \quad m_i^1 = 0, \quad m_i^2 = 1, \quad m_i^3 = h_i(\mathbf{A}, \mathbf{U}_p, M_{111}^3/M^0, \ldots, M_{222}^3/M^0), \tag{31}$$

where h_i depends, in general, on all 10 third-order velocity moments (see Appendix A).

Using the two-node quadrature formulas (Eq. (25)), the moments of X_i can be inverted for $i \in (1,2)$ to find $(n_{(1)}, n_{(2)}, X_{(1)}, X_{(2)})_i$:

$$\begin{aligned} n_{(i)1} &= 0.5 + \gamma_i, \quad X_{(i)1} = -\left(\frac{1-2\gamma_i}{1+2\gamma_i}\right)^{1/2}, \\ n_{(i)2} &= 0.5 - \gamma_i, \quad X_{(i)2} = \left(\frac{1+2\gamma_i}{1-2\gamma_i}\right)^{1/2}, \end{aligned} \tag{32}$$

where $(-1/2 < \gamma_i < 1/2)$

$$\gamma_i = \frac{m_i^3/2}{[(m_i^3)^2 + 4]^{1/2}}. \tag{33}$$

The locations of the nodes before and after the linear transformation are illustrated in Fig. 1 for a Maxwellian distribution with zero mean velocity.

The four-node quadrature approximation is then defined using the tensor product of the one-dimensional abscissas as

$$V_4^* = [(n_1^*, X_{(1)1}, X_{(2)1}), (n_2^*, X_{(1)1}, X_{(2)2}), (n_3^*, X_{(1)2}, X_{(2)1}), (n_4^*, X_{(1)2}, X_{(2)2})], \tag{34}$$

where the (as yet) unknown weights n_α^* must obey the linear equations

$$\begin{aligned} n_1^* + n_2^* &= n_{(1)1} \\ n_3^* + n_4^* &= n_{(1)2} \\ n_1^* + n_3^* &= n_{(2)1} \\ n_2^* + n_4^* &= n_{(2)2} \end{aligned} \quad \Rightarrow \quad \begin{aligned} n_1^* - n_4^* &= n_{(1)1} - n_{(2)2} \\ n_2^* + n_4^* &= n_{(2)2} \\ n_3^* + n_4^* &= n_{(1)2} \end{aligned}. \tag{35}$$

The equations on the left-hand side are the weight constraints for each node, and the equations on the right-hand side are a linearly independent set. The right-hand sides of Eq. (35) are known, and have the property that $n_{(1)1} + n_{(1)2} = 1$ and $n_{(2)1} + n_{(2)2} = 1$.

The linear system in Eq. (35) has rank three. We must therefore add another linear equation to define the four weights. For this purpose, we will use the cross moment $m_{12}^2 = \langle X_1 X_2 \rangle = 0$, the value of which follows from the definition of $\mathbf{A}$. In terms of the weights and abscissas in Eq. (34), we have

[6] Strictly speaking, it is not necessary to diagonalize the covariance matrix. However, if this is not done, it can be shown that certain realizable moment sets lead to negative weights.

[7] The transformation is not unique because we can always introduce an orthogonal matrix $\mathbf{C}$ with the property $\mathbf{C}^T = \mathbf{C}^{-1}$ such that $\mathbf{L}^T\mathbf{C}^T\mathbf{C}\mathbf{L} = \sigma_U$. The linear transformation can then be defined as $\mathbf{A} = (\mathbf{C}\mathbf{L})^T$. There are an infinite number of choices for $\mathbf{C}$, e.g., the rotation matrices.

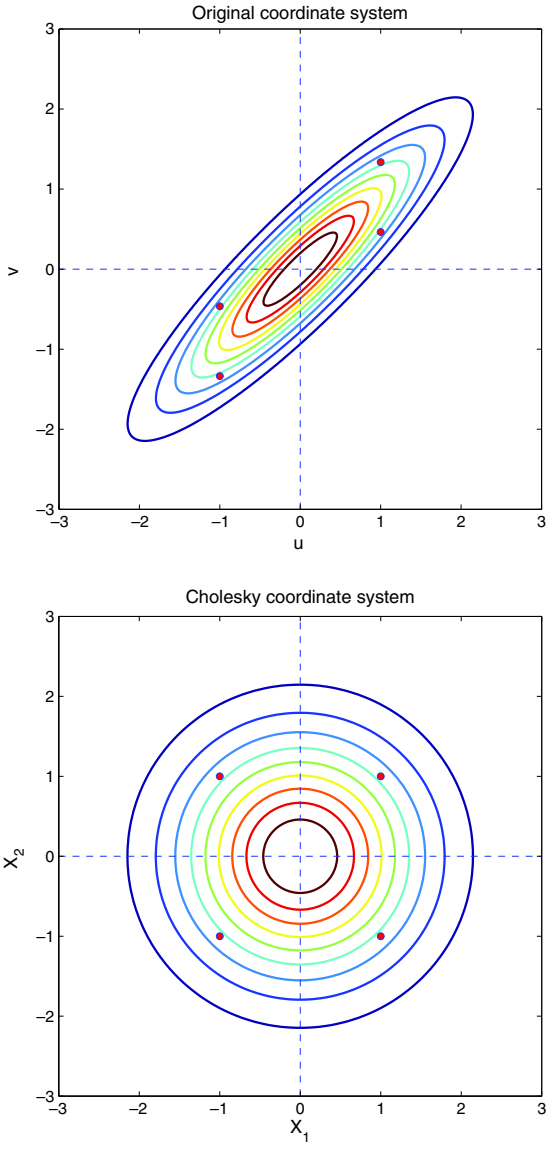


Fig. 1. Four-node quadrature representation of the two-dimensional equilibrium distribution function $f_{\mathrm{eq}}(u,v)$ (equally-spaced contours) in velocity phase space (top) and in transformed phase space (bottom). The abscissas are denoted by the four filled circles. For this example the weights are all equal to $M^0/4$.

$$X_{(1)1}X_{(2)1}n_1^* + X_{(1)1}X_{(2)2}n_2^* + X_{(1)2}X_{(2)1}n_3^* + X_{(1)2}X_{(2)2}n_4^* = 0. \tag{36}$$

Recall that the abscissas ($X_{(i)\alpha}$) are known; hence, this expression is linear in the weights (n_α^*), and we can use it with Eq. (35) to form a full-rank linear system. The resulting system can be inverted analytically to find

$$\begin{aligned} n_1^* &= n_{(1)1}n_{(2)1} = (0.5+\gamma_1)(0.5+\gamma_2) \\ n_2^* &= n_{(1)1}n_{(2)2} = (0.5+\gamma_1)(0.5-\gamma_2) \\ n_3^* &= n_{(1)2}n_{(2)1} = (0.5-\gamma_1)(0.5+\gamma_2) \\ n_4^* &= n_{(1)2}n_{(2)2} = (0.5-\gamma_1)(0.5-\gamma_2). \end{aligned} \tag{37}$$

Note that these weights are always non-negative.

In summary, the weights and abscissas in V_4 are found from those in V_4^* using Eq. (30) to invert the abscissas and $n_\alpha = M^0 n_\alpha^*$. The eight moments controlled in this process are

$$W^{2*} = (m^0, m_1^1, m_2^1, m_1^2, m_{12}^2, m_2^2, m_1^3, m_2^3).$$

Note that the two third-order moments in W^{2*} are a linear combination of the four third-order moments in W^2. Hence, W^{2*} is a subset of W^2 containing eight independent moments (instead of 10). However, given moments in W^2 it is straightforward to project them (using the weights and abscissas) into $W^{2\dagger}$, i.e., the eight-dimensional moment subspace that can be represented by V^4 is $W^{2\dagger}$. The overall procedure can be represented as

$$W^2 \to W^{2*} \leftrightarrow V_4^* \leftrightarrow V_4 \leftrightarrow W^{2\dagger} \subset W^2,$$

where a projection step is used to define $W^{2\dagger}$. We will return to this point when discussing the numerical implementation below.

4.5. Eight-node quadrature: V_8 from W^3

In three dimensions, the set of 20 moments up to third order is W^3, and we seek to define a eight-node quadrature based on these moments. We begin again by defining the mean particle velocity vector:

$$\mathbf{U}_\mathrm{p} = \begin{bmatrix} M_1^1/M^0 \\ M_2^1/M^0 \\ M_3^1/M^0 \end{bmatrix}, \tag{38}$$

and the velocity covariance matrix:

$$\sigma_U = \begin{bmatrix} M_{11}^2/M^0 - U_{\mathrm{p}1}^2 & M_{12}^2/M^0 - U_{\mathrm{p}1}U_{\mathrm{p}2} & M_{13}^2/M^0 - U_{\mathrm{p}1}U_{\mathrm{p}3} \\ M_{12}^2/M^0 - U_{\mathrm{p}1}U_{\mathrm{p}2} & M_{22}^2/M^0 - U_{\mathrm{p}2}^2 & M_{23}^2/M^0 - U_{\mathrm{p}2}U_{\mathrm{p}3} \\ M_{13}^2/M^0 - U_{\mathrm{p}1}U_{\mathrm{p}3} & M_{23}^2/M^0 - U_{\mathrm{p}2}U_{\mathrm{p}3} & M_{33}^2/M^0 - U_{\mathrm{p}3}^2 \end{bmatrix}, \tag{39}$$

so that $\sigma_\mathrm{eq} = (\sigma_{11} + \sigma_{22} + \sigma_{33})/3$.

The eight-node extension of the procedure described above uses the three-component vector $\mathbf{X} = [X_1 X_2 X_3]^\mathrm{T}$ defined by

$$\mathbf{X} = \mathbf{A}^{-1}(\mathbf{v} - \mathbf{U}_\mathrm{p}) \quad \text{so that } \mathbf{v} = \mathbf{A}\mathbf{X} + \mathbf{U}_\mathrm{p}, \tag{40}$$

where $\mathbf{A} = \mathbf{L}^\mathrm{T}$ is again defined in terms of the Cholesky decomposition of σ_U:

$$\mathbf{L} = \begin{bmatrix} \sigma_{11}^{1/2} & \sigma_{12}/\sigma_{11}^{1/2} & \sigma_{13}/\sigma_{11}^{1/2} \\ 0 & (\sigma_{22} - \sigma_{12}^2/\sigma_{11})^{1/2} & \frac{\sigma_{11}\sigma_{23} - \sigma_{13}\sigma_{12}}{\sigma_{11}^{1/2}(\sigma_{11}\sigma_{22} - \sigma_{12}^2)^{1/2}} \\ 0 & 0 & \left(\sigma_{33} - \sigma_{13}^2/\sigma_{11} - \frac{(\sigma_{11}\sigma_{23} - \sigma_{13}\sigma_{12})^2}{\sigma_{11}(\sigma_{11}\sigma_{22} - \sigma_{12}^2)}\right)^{1/2} \end{bmatrix}. \tag{41}$$

As before, the two-node quadrature formulas (Eq. (32) with $i \in (1, 2, 3)$) are used to find the weights and abscissas in each direction. The eight-node quadrature approximation is then defined using the tensor product of the one-dimensional abscissas by

$$\begin{aligned} V_8^* = [&(n_1^*, X_{(1)1}, X_{(2)1}, X_{(3)1}), (n_2^*, X_{(1)2}, X_{(2)1}, X_{(3)1}), (n_3^*, X_{(1)1}, X_{(2)2}, X_{(3)1}), (n_4^*, X_{(1)2}, X_{(2)2}, X_{(3)1}), \\ &(n_5^*, X_{(1)1}, X_{(2)1}, X_{(3)2}), (n_6^*, X_{(1)2}, X_{(2)1}, X_{(3)2}), (n_7^*, X_{(1)1}, X_{(2)2}, X_{(3)2}), (n_8^*, X_{(1)2}, X_{(2)2}, X_{(3)2})]. \end{aligned} \tag{42}$$

The linear system for the weights:

$$\begin{aligned}
&n_1^* + n_3^* + n_5^* + n_7^* = n_{(1)1}\\
&n_2^* + n_4^* + n_6^* + n_8^* = n_{(1)2}\\
&n_1^* + n_2^* + n_5^* + n_6^* = n_{(2)1}\\
&n_3^* + n_4^* + n_7^* + n_8^* = n_{(2)2}\\
&n_1^* + n_2^* + n_3^* + n_4^* = n_{(3)1}\\
&n_5^* + n_6^* + n_7^* + n_8^* = n_{(3)2}
\end{aligned}
\quad\Rightarrow\quad
\begin{aligned}
&n_1^* + n_3^* + n_5^* + n_7^* = n_{(1)1}\\
&n_2^* + n_4^* + n_6^* + n_8^* = 1 - n_{(1)1}\\
&n_3^* + n_4^* + n_7^* + n_8^* = 1 - n_{(2)1}\\
&n_5^* + n_6^* + n_7^* + n_8^* = 1 - n_{(3)1}
\end{aligned}
\tag{43}$$

where $n_{(i)1} + n_{(i)2} = 1, i \in (1,2,3)$; has rank 4. Thus, in general, four additional equations are required to determine n_α^*. Three of these equations come from the second-order moments $\langle X_i X_j \rangle$:

$$\begin{aligned}
&X_{(1)1}X_{(2)1}n_1^* + X_{(1)2}X_{(2)1}n_2^* + X_{(1)1}X_{(2)2}n_3^* + X_{(1)2}X_{(2)2}n_4^* + X_{(1)1}X_{(2)1}n_5^*\\
&\quad + X_{(1)2}X_{(2)1}n_6^* + X_{(1)1}X_{(2)2}n_7^* + X_{(1)2}X_{(2)2}n_8^* = m_{12}^2 = 0,\\
&X_{(1)1}X_{(3)1}n_1^* + X_{(1)2}X_{(3)1}n_2^* + X_{(1)1}X_{(3)1}n_3^* + X_{(1)2}X_{(3)1}n_4^* + X_{(1)1}X_{(3)2}n_5^*\\
&\quad + X_{(1)2}X_{(3)2}n_6^* + X_{(1)1}X_{(3)2}n_7^* + X_{(1)2}X_{(3)2}n_8^* = m_{13}^2 = 0,\\
&X_{(2)1}X_{(3)1}n_1^* + X_{(2)1}X_{(3)1}n_2^* + X_{(2)2}X_{(3)1}n_3^* + X_{(2)2}X_{(3)1}n_4^* + X_{(2)1}X_{(3)2}n_5^*\\
&\quad + X_{(2)1}X_{(3)2}n_6^* + X_{(2)2}X_{(3)2}n_7^* + X_{(2)2}X_{(3)2}n_8^* = m_{23}^2 = 0;
\end{aligned}
\tag{44}$$

and the fourth equation comes from the third-order moment $\langle X_1 X_2 X_3 \rangle$[8]:

$$\begin{aligned}
&X_{(1)1}X_{(2)1}X_{(3)1}n_1^* + X_{(1)2}X_{(2)1}X_{(3)1}n_2^* + X_{(1)1}X_{(2)2}X_{(3)1}n_3^* + X_{(1)2}X_{(2)2}X_{(3)1}n_4^* + X_{(1)1}X_{(2)1}X_{(3)2}n_5^*\\
&\quad + X_{(1)2}X_{(2)1}X_{(3)2}n_6^* + X_{(1)1}X_{(2)2}X_{(3)2}n_7^* + X_{(1)2}X_{(2)2}X_{(3)2}n_8^*\\
&\quad = m_{123}^3.
\end{aligned}
\tag{45}$$

The moment m_{123}^3 and the abscissas are known. The linear system of eight equations (Eqs. (43)–(45)) can be solved analytically.

In order to ensure that the weights are non-negative, we define

$$\rho_{123}^* = m_{123}^3 [a(1-a)b(1-b)c(1-c)]^{1/2}, \tag{46}$$

where $a = n_{(1)1} = 0.5 + \gamma_1$, $b = n_{(2)1} = 0.5 + \gamma_2$, $c = n_{(3)1} = 0.5 + \gamma_3$. Then, if $m_{123}^3 \geqslant 0$, we let

$$\rho_{123} = \min[\rho_{123}^*, abc, (1-a)(1-b)c, (1-a)b(1-c), a(1-b)(1-c)]. \tag{47}$$

Otherwise, if $m_{123}^3 < 0$, we let

$$\rho_{123} = -\min[-\rho_{123}^*, (1-a)bc, a(1-b)c, ab(1-c), (1-a)(1-b)(1-c)]. \tag{48}$$

The weights are then given by

$$\begin{aligned}
n_1^* &= abc - \rho_{123},\\
n_2^* &= (1-a)bc + \rho_{123},\\
n_3^* &= a(1-b)c + \rho_{123},\\
n_4^* &= (1-a)(1-b)c - \rho_{123},\\
n_5^* &= ab(1-c) + \rho_{123},\\
n_6^* &= (1-a)b(1-c) - \rho_{123},\\
n_7^* &= a(1-b)(1-c) - \rho_{123},\\
n_8^* &= (1-a)(1-b)(1-c) + \rho_{123}.
\end{aligned}
\tag{49}$$

[8] Obviously this choice is not unique. Any linear combination of the seven uncontrolled third-order moments could be used: m_{112}^3, m_{113}^3, m_{122}^3, m_{123}^3, m_{133}^3, m_{223}^3, m_{233}^3. The choice of m_{123}^3 has the "advantage" that is symmetric in the indices and it is easy to control the resulting weights so that they are non-negative.

Note that for cases where $\rho_{123} \neq \rho^*_{123}$, one of the weights will be zero. Hence, in order to ensure non-negative weights, we essentially drop Eq. (45) and set the offending weight to zero. The remaining seven weights are determined by the other seven linear equations.

In summary, the 14 moments[9] controlled with eight-node quadrature are

$$W^{3*} = (m^0, m^1_1, m^1_2, m^1_3, m^2_1, m^2_2, m^2_3, m^2_{12}, m^2_{13}, m^2_{23}, m^3_1, m^3_2, m^3_3, m^3_{123}).$$

The overall procedure can be represented as

$$W^3 \to W^{3*} \leftrightarrow V^*_8 \leftrightarrow V_8 \leftrightarrow W^{3\dagger} \subset W^3,$$

where a projection step is used to define $W^{3\dagger}$. Similar to two dimensions, only four of the 10 third-order moments appear in W^{3*}. These four moments are independent linear combinations of the 10 third-order velocity moments.

4.6. Comparison to other methods

In the two-node quadrature method of [19,20], a different approach was used to fix the weights and abscissas. The difference lies essentially in which moments are controlled. In [19,20] the eight moments

$$(M^0, M^1_1, M^1_2, M^1_3, M^2_{11}, M^2_{22}, M^2_{33}, M^3_{111} + M^3_{222} + M^3_{333})$$

were used. Note that this moment set does not include cross moments of second order (e.g. M^2_{12}). In contrast, in the new formulation the covariance matrix is exactly reproduced. This of course comes at the expense of transporting more moments (20 instead of eight).

Note that in the previous method the number of moments is equal to the number of unknown weights and abscissas. It might be argued that the new method could be improved for $d > 1$ because the number of moments controlled is less than the number of unknowns. For example, if we consider the eight-node case (32 unknowns), we can observe that the 14 moments in W^{3*} do not include all third-order moments. However, [28] has shown that an "optimal" moment set exists wherein 32 moments, including all moments of third order, can be used to find the weights and abscissas in V^*_8. Likewise, the eight moments in W^{2*} can be augmented by four additional moments: m^3_{112}, m^3_{122}, m^4_{1112}, and m^4_{1222}, to form an optimal set for computing V^*_4. In practice, the use of the optimal moment sets introduces two new issues. First, the weights and abscissas must be found numerically by solving a nonlinear system of equations that can be poorly conditioned. Second, the optimal moment sets contain fourth- and fifth-order moments, all of which would have to be added to W (i.e., more moment transport equations are needed).

In order to test the feasibility of using optimal moments, we have implemented a nonlinear (NL) solver to compute V^*_4 using as initial guess the results given in Section 4.4. We found that in most cases considered (e.g., the Riemann problem), the NL solver was able to converge (albeit very slowly), but not always. The non-convergent cases typically have weights that are very unequal. However, it was not possible to predict in advance which moment sets would not converge. In all likelihood, the convergence problems will be more acute for V^*_8 given the larger number of moments in the optimal set that are not controlled by the new method. Thus, given that the computational cost is orders of magnitude larger and convergence is not guaranteed, it is unlikely that a NL solver will provide a computationally efficient method for inverting moment sets. This conclusion is consistent with the findings of [70]. In general, since moment inversion in one dimension is well established and reliable [49], it will likely be much more fruitful to increase the number of nodes and the order of moments (e.g. computing $\beta = 3^d$ nodes requires fifth-order moments) to achieve greater accuracy.

We should note that moment-inversion method proposed here is very similar in spirit to the method proposed by Yoon and McGraw [70] for aerosol applications (i.e., passive transport of a distribution function). The principal differences are (1) in [70] they define the linear transformation matrix $\mathbf{A}$ in terms of the eigenvectors of the covariance matrix, and (2) they compute the weights without attempting to control m^3_{123} (i.e.,

[9] Only 13 moments are controlled when one weight is null. However, this never occurs for any of the examples considered in this paper.

their formula is equivalent to setting $\rho_{123} = 0$). Of these differences, the first is the most important for the applications considered in this work. We have, in fact, attempted to define **A** as done by Yoon and McGraw [70]. However, because the velocity is a dynamic variable (i.e., the spatial fluxes are computed using the velocity abscissas), the properties of the eigenvectors make them a poor choice for $d = 3$.[10] The fundamental difficulty is the fact that the eigenvectors of σ_U do not vary smoothly with its components. In other words, small changes in the correlation structure cause the orientation of the eigenvectors to jump in a discontinuous manner. Thus, at two neighboring spatial locations, the resulting velocity abscissas can be completely different (even though the moments are continuous). The net result is that the fluxes computed from the abscissas are then discontinuous, leading to "random" fluctuations in the moments. In contrast, the Cholesky matrix **L** varies smoothly with the components of σ_U and, hence, the fluxes are well behaved. The only known disadvantage of using the Cholesky matrix is that it depends on the ordering of the covariance matrix, and is thus different for each of the six permutations of the coordinates. In essence, one direction is chosen as the principal direction (i.e., X_1 in Fig. 1). For the examples considered in this work, this turns out not to be a problem because of symmetry. However, it would be desirable to replace **L** with a smoothly varying, permutation-invariant linear transform that generates abscissas close to those found using **L** (if it exists) and still diagonalizes σ_U.

Finally, we can note that quadrature-based moment closures bear some resemblance to discrete velocity models [10,31,51] and to the lattice Boltzmann method (LBM) [30,41], including multispeed off-lattice Boltzmann methods [1]; however, the differences are significant. (See Appendix B for more details.) For example, in LBM the discretizations of velocity space, physical space and time are strongly coupled and determine the numerical algorithm and properties [41], while quadrature-based moment methods do not directly imply a physical-space and time discretization scheme. This can be seen from the fact that the moment equations closed by quadrature are still written as continuous functions of physical space and time. The principal similarity between quadrature methods and LBM is that both represent velocity phase space by a discrete set of velocities. However, a very significant difference between the two methods is that in LBM the discrete velocities are *fixed* (depending on the lattice, spatial dimensions, etc. [41]), while in quadrature methods they are variables (i.e., fields) that vary in space (velocity and physical) and time according to the local flow physics (i.e., the velocity moments). Another important difference is the number of unknowns available to control the moments. In LBM, only the weights are used to construct a *linear* mapping into the moments [41] and, hence, the maximum number of moments that can be controlled is equal to the number of weights (e.g., the 13-velocity model D3Q13 corresponds to 13 moments up to third order in three dimensions). In contrast, we have seen that by using quadrature eight velocity abscissas could, in principle, control 32 moments if all the weights and abscissas were allowed to vary. However, because of the difficulty with inverting the *nonlinear* mapping between the moments and the quadrature nodes, the full quadrature procedure is intractable. Nevertheless, the partial quadrature method presented here still allows us to control 14 moments with only eight nodes (i.e., essentially the same number of moments as with D3Q13). Another difference is that LBM uses a linear relaxation process to represent collisions [41], while quadrature can be developed for any closed collision operator (e.g. the Boltzmann hard-sphere collision operator [64]). Finally, we can note that as shown in [19] quadrature-based methods can be used for arbitrary collision times (or even collision-less cases), while LBM methods are designed for collision-dominated flows [41]. Based on these differences, we can expect that the range of flow phenomena that can be captured using quadrature methods will be significantly larger than with LBM (or with multispeed off-lattice Boltzmann methods [1]).

5. Numerical algorithm

The system of moment transport equations can be solved numerically for three-dimensional problems by extending the numerical methods described in [19,20]. Here, however, we are interested in solutions that depend on only one spatial dimension and time. Hence we will describe the method in the context of one-dimensional fluxes.

[10] The problem can be fixed for $d = 2$ because the eigenvectors can be constructed to be independent of σ_{12}.

5.1. Transport equations

The examples considered in this work have only one inhomogeneous direction (x_1), and we can take gravity (when needed) to be in the $-x_2$ direction. Thus we need to define the numerical scheme in one spatial dimension (albeit with a three-dimensional velocity phase space) for the following moment equations:

$$\begin{aligned}
&\frac{\partial M^0}{\partial t} + \frac{\partial M_1^1}{\partial x_1} = 0,\\
&\frac{\partial M_i^1}{\partial t} + \frac{\partial M_{i1}^2}{\partial x_1} = D_i^1,\\
&\frac{\partial M_{ij}^2}{\partial t} + \frac{\partial M_{ij1}^3}{\partial x_1} = D_{ij}^2 + C_{ij}^2,\\
&\frac{\partial M_{ijk}^3}{\partial t} + \frac{\partial M_{ijk1}^4}{\partial x_1} = D_{ijk}^3 + C_{ijk}^3.
\end{aligned} \tag{50}$$

The source terms due to gravity and drag are defined by

$$\begin{aligned}
D_i^1 &= \sum_{\alpha=1}^{\beta} n_\alpha \left(\frac{F_{i\alpha}}{m_\mathrm{p}} - \delta_{i2} g \right),\\
D_{ij}^2 &= \sum_{\alpha=1}^{\beta} n_\alpha \left[\left(\frac{F_{i\alpha}}{m_\mathrm{p}} - g\delta_{i2} \right) U_{j\alpha} + \left(\frac{F_{j\alpha}}{m_\mathrm{p}} - g\delta_{j2} \right) U_{i\alpha} \right],\\
D_{ijk}^3 &= \sum_{\alpha=1}^{\beta} n_\alpha \left[\left(\frac{F_{i\alpha}}{m_\mathrm{p}} - g\delta_{i2} \right) U_{j\alpha} U_{k\alpha} + \left(\frac{F_{j\alpha}}{m_\mathrm{p}} - g\delta_{j2} \right) U_{k\alpha} U_{i\alpha} + \left(\frac{F_{k\alpha}}{m_\mathrm{p}} - g\delta_{k2} \right) U_{i\alpha} U_{j\alpha} \right],
\end{aligned} \tag{51}$$

and the source terms due to collisions by

$$\begin{aligned}
C_{ij}^2 &= \frac{M^0}{\tau} (\sigma_\mathrm{eq} \delta_{ij} - \sigma_{ij}),\\
C_{ijk}^3 &= \frac{1}{\tau} (\Delta_{ijk} - M_{ijk}^3).
\end{aligned} \tag{52}$$

We shall see below that dividing the source terms into two contributions follows naturally from the effects of gravity and drag on the weights and abscissas (i.e. they are convective processes in phase space).

5.2. The solution algorithm

The steps in the proposed solution algorithm are as follows:

(1) Initialize the weights and abscissa in V_β^* and the moments in $W^{d\dagger}$.
(2) Advance in time (Δt) the moments in W^d using the moment transport equations (time split):
   - advance moments due to fluxes;
   - advance moments due to gravity and drag;
   - advance moments due to collisions.

(3) Invert the moments in W^d to find the weights and abscissas in V_β.
(4) Compute moments in $W^{d\dagger}$ using V_β (projection step).
(5) Return to step 2.

The time step Δt is set to a fraction of the smallest characteristic time (convection, drag, or collisions).

5.3. A first-order, one-dimensional scheme

Because of the conservative form of Eq. (50), the finite-volume method [42] is a natural candidate for its discretization. The underlying kinetic equation (Eq. (1)) can be used for the derivation of a numerical flux formula that ensures the robustness of the corresponding scheme. We begin by introducing the following notation for the vector of velocity moments up to third order, their spatial flux terms, and their source terms [19,20]:

$$W = \begin{pmatrix} M^0 \\ M^1 \\ M^2 \\ M^3 \end{pmatrix}, \quad H(W) = \begin{pmatrix} M^1 \\ M^2 \\ M^3 \\ M^4 \end{pmatrix}, \quad S(W) = \begin{pmatrix} 0 \\ D^1 \\ D^2 + C^2 \\ D^3 + C^3 \end{pmatrix}.$$

Using classical notation, a fractional two-step, first-order, explicit, finite-volume scheme for Eq. (50) reads

$$\begin{aligned} W_i^* &= W_i^n - \frac{\Delta t}{\Delta x}[G(W_i^n, W_{i+1}^n) - G(W_{i-1}^n, W_i^n)], \\ W_i^{n+1} &= \widetilde{W}(W_i^*, \Delta t), \end{aligned} \tag{53}$$

where G is the corresponding numerical flux function (described below) and $\widetilde{W}(W_i^*, \Delta t)$ is an approximate solution at time $t = \Delta t$ of the differential system

$$\frac{\mathrm{d}W}{\mathrm{d}t} = S(W) \quad \text{with } W(0) = W_i^*. \tag{54}$$

The advantage of a fractional-step algorithm is the possibility of using a quasi-analytic solution for the second step of the scheme to handle the stiffness of the source terms. Due to their forms, we will split the contributions to the source terms into two parts: (i) drag and gravity; (ii) collisions.

5.3.1. Drag and gravity

To compute the first part, we note that the drag and gravity terms in Eq. (51) are equivalent to

$$\begin{aligned} \frac{\mathrm{d}n_\alpha}{\mathrm{d}t} &= 0, \\ \frac{\mathrm{d}U_{i\alpha}}{\mathrm{d}t} &= \frac{F_{i\alpha}}{m_\mathrm{p}} - g\delta_{i2}. \end{aligned} \tag{55}$$

These expressions can be explicitly solved to find the changes in the weights and abscissas due to drag and gravity:

$$\begin{aligned} n_\alpha^{**} &= n_\alpha^*, \\ U_{i\alpha}^{**} &= \exp(-\Delta t/\tau_\alpha)U_{i\alpha}^* + [1 - \exp(-\Delta t/\tau_\alpha)](U_{\mathrm{f}i} - g\delta_{i2}\tau_\alpha), \end{aligned} \tag{56}$$

where n_α^* and $U_{i\alpha}^*$ are the weights and abscissas found from W_i^* using quadrature, and τ_α is the corresponding drag time (see Eq. (4)) evaluated at the beginning of the step. The results in Eq. (56) are then employed to compute the moments W_i^{**} using projection.

5.3.2. Collisions

The next step is to compute the contribution due to collisions:

$$\frac{\mathrm{d}W}{\mathrm{d}t} = C(W) \quad \text{with } W(0) = W_i^{**}. \tag{57}$$

This can be done explicitly:

$$W_i^{***} = \exp(-\Delta t/\tau_i)W_i^{**} + [1 - \exp(-\Delta t/\tau_i)]\Delta(W_i^*), \tag{58}$$

where τ_i is the local collision time and $\Delta(W)$ denotes the vector of equilibrium moments. To finish up, quadrature is used to compute n_α^{n+1} and $U_{i\alpha}^{n+1}$ from W_i^{***}, and W_i^{n+1} is computed using projection.

5.4. Kinetic-based definition of the fluxes

Using the ideas of kinetic-based schemes [54,53], it is natural to adopt the following expression for the numerical flux function (see Eq. (18)):

$$G(W_{\mathrm{l}}, W_{\mathrm{r}}) = Q^{+}(W_{\mathrm{l}}) + Q^{-}(W_{\mathrm{r}}) \tag{59}$$

with

$$Q^{+}(W_{\mathrm{l}}) = \sum_{\alpha=1}^{\beta} n_{\alpha \mathrm{l}} \max(0, U_{1\alpha \mathrm{l}}) \begin{pmatrix} 1 \\ U_{i\alpha \mathrm{l}} \\ U_{i\alpha \mathrm{l}} U_{j\alpha \mathrm{l}} \\ U_{i\alpha \mathrm{l}} U_{j\alpha \mathrm{l}} U_{k\alpha \mathrm{l}} \end{pmatrix} \tag{60}$$

and

$$Q^{-}(W_{\mathrm{r}}) = \sum_{\alpha=1}^{\beta} n_{\alpha \mathrm{r}} \min(0, U_{1\alpha \mathrm{r}}) \begin{pmatrix} 1 \\ U_{i\alpha \mathrm{r}} \\ U_{i\alpha \mathrm{r}} U_{j\alpha \mathrm{r}} \\ U_{i\alpha \mathrm{r}} U_{j\alpha \mathrm{r}} U_{k\alpha \mathrm{r}} \end{pmatrix}, \tag{61}$$

where the indices (i, j, k) on the velocity abscissas in the column vector (of length 20) on the right-hand side are chosen to represent the moments up to third order in W^d. Note that although these expressions have been written for fluxes in the x_1 direction, the extension to other directions is straightforward. As noted earlier, the use of kinetic-based fluxes is a key difference between quadrature-based moment methods and "standard" moment methods.

5.5. Boundary conditions

In order to evaluate the fluxes in Eq. (59), we will need to specify boundary conditions. Without loss of generality, we can consider the left-hand side of the domain and denote the boundary cell by $i = 0$. From the definition of the fluxes, we can observe that it suffices to specify the weights and abscissas at $i = 0$. For the examples in Section 6, we will use reflective boundary conditions:

$$\begin{pmatrix} n_\alpha \\ U_{1\alpha} \\ U_{2\alpha} \\ U_{3\alpha} \end{pmatrix}_{i=0} = \begin{pmatrix} n_\alpha / e_w \\ -e_w U_{1\alpha} \\ U_{2\alpha} \\ U_{3\alpha} \end{pmatrix}_{i=1}, \tag{62}$$

where $0 < e_w \leqslant 1$ is the coefficient of restitution for the wall. If needed, the moments at $i = 0$ can be computed from Eq. (62). Note that other types of boundary conditions can be readily accommodated. For example, if the wall distribution function f_{w} is known, then it suffices to compute its moments $W_{i=0}$ and the corresponding weights and abscissas. The latter would then be used in place of Eq. (62). In fact, because quadrature methods use realizable weights and abscissas to represent the moments, such methods can employ any boundary condition used in a Lagrangian method (e.g., in DSMC [6]).

6. Example applications

We will consider three example applications that test different aspects of the proposed closure. The first example, the Riemann shock problem, tests the ability of the method to capture shock propagation in non-isothermal flows at arbitrary Knudsen and Mach numbers. However, we should emphasize that this is not our target application and hence we do not expect the quadrature-based closure to perform as well as models specifically designed to treat shocks [4,6,13,16,17,34,41,52–54,62]. The second example, impinging particle jets, includes the fluid drag force and demonstrates the ability of the method to capture particle trajectory crossing

[19,20] with different collision frequencies (i.e. from no collisions to instantaneous collisions). The third example, vertical gas-particle channel flow, tests all aspects of the model (gravity, drag, collisions, bounded domain) needed to describe dilute gas-particle flows.

6.1. The Riemann problem

The first test problem is the Riemann shock problem described in [4] with an initial density ratio of $\rho_4/\rho_1 = 3.093$, which corresponds to a Mach number of 1.25. For this problem the mean velocity is initially zero everywhere, and the velocity variance (temperature) is normalized to unity everywhere. (Non-uniform temperature distributions pose no additional difficulties.) The drag and gravity terms are not present in this flow. In the limit where the collision time is null ($\tau = 0$), the moment transport equations reduce to the Euler equation for an inviscid compressible fluid. As discussed in [4,16,17,52], the time evolution of the system is well known and includes a left-moving rarefaction wave (or expansion fan), a contact surface, and a right-moving shock wave. The velocity of the shock wave in the Euler limit can be computed from the density ratio and the Mach number. When the collision time is finite ($\tau > 0$) the structure of the solution changes and the temperature in the shock is no longer isotropic (i.e., the velocity variance in the direction of the flow is larger than the variances in the normal directions). In the absence of collisions ($\tau = \infty$) particles move without changing their velocity. Note that because we are using a simple BGK collision model, no attempt will be made to relate τ to a molecular collision time scale. (See [16] for a discussion of the BGK model.) Instead, we will run the model with different values of $\tau = \kappa/M^0$ to investigate the numerical stability of the quadrature-based moment closure.

For simplicity, we will take the normalized flow domain to be $0 < y < 1$ ($y = x_1$) and will stop the simulation before the shock wave reaches a wall (i.e. the wall boundary conditions have no effect on the results). The numerical algorithm is run with a CFL number of 0.5 and 400 uniformly spaced grid cells. (The CFL number is defined in terms of the largest magnitude of all velocity abscissas [20].) Grid refinement studies showed no further improvements on a finer grid. Cases with $\kappa = 0$ are run with a time step determined by the CFL condition and the collisions reset the moments to the equilibrium values at the end of the time step (but before inverting the moments to find weights and abscissas). Cases with finite κ use a time step found from the minimum of the CFL time and $\kappa/10$. Note that because the moment equations are solved directly in conservative form and the quadrature inversion procedure exactly conserved the moments, the lower-order moments are conserved for all values of κ. (See discussion in [4,17,52,62] on difficulties with conserving moments with other methods.) Finally, we should note that the model can be solved on a personal computer with run times on the order of seconds.

The structure of the flow can be seen from the temperature contour plots in Fig. 2. For $\kappa = 0$ the classical shock structure is found and agrees well with the known solution to the Euler equation [4]. For $\kappa = 0.1$ the shock and contact surface broaden and the shock speed is reduced. Finally, for $\kappa = \infty$ the flow structure is very different with symmetric high temperature waves traveling to the left and right, separated by a low-temperature zone.

An example of the 20 moments is shown in Fig. 3 for the case $\kappa = 0$ at time $t = 0.23$ (i.e., at the end of the simulation). It can be noted that due to the symmetry of the Riemann problem, 12 of the moments are null. (The eight non-zero moments are M^0, M^1_2, M^2_{11}, M^2_{22}, M^2_{33}, M^3_{112}, M^3_{222}, and M^3_{233}.) Note that because the off-diagonal second-order moments are null, the Cholesky decomposition ($\mathbf{L}$) is diagonal. This implies that the third-order moments controlled by the quadrature method are M^3_{111}, M^3_{222}, M^3_{333}, and M^3_{123}, while the other third-order moments are found by projection. Hence, the results in Fig. 3 show that the projection operation does not introduce errors in moments that should be null by symmetry. Finally, we can note that if one is only interested in solving the Riemann problem the computational efficiency can be increased by solving only for the non-zero moments needed for quadrature inversion.[11]

[11] A better alternative for the Riemann problem would be to rewrite the density function into two separate univariate density functions as proposed by Chu [16] and done, for example, in [52]. One-dimensional quadrature could then be used for each density function (i.e. two sets of weights and abscissas). Because 1D quadrature can use the PD algorithm [49] it would be straightforward to go to higher-order moments.

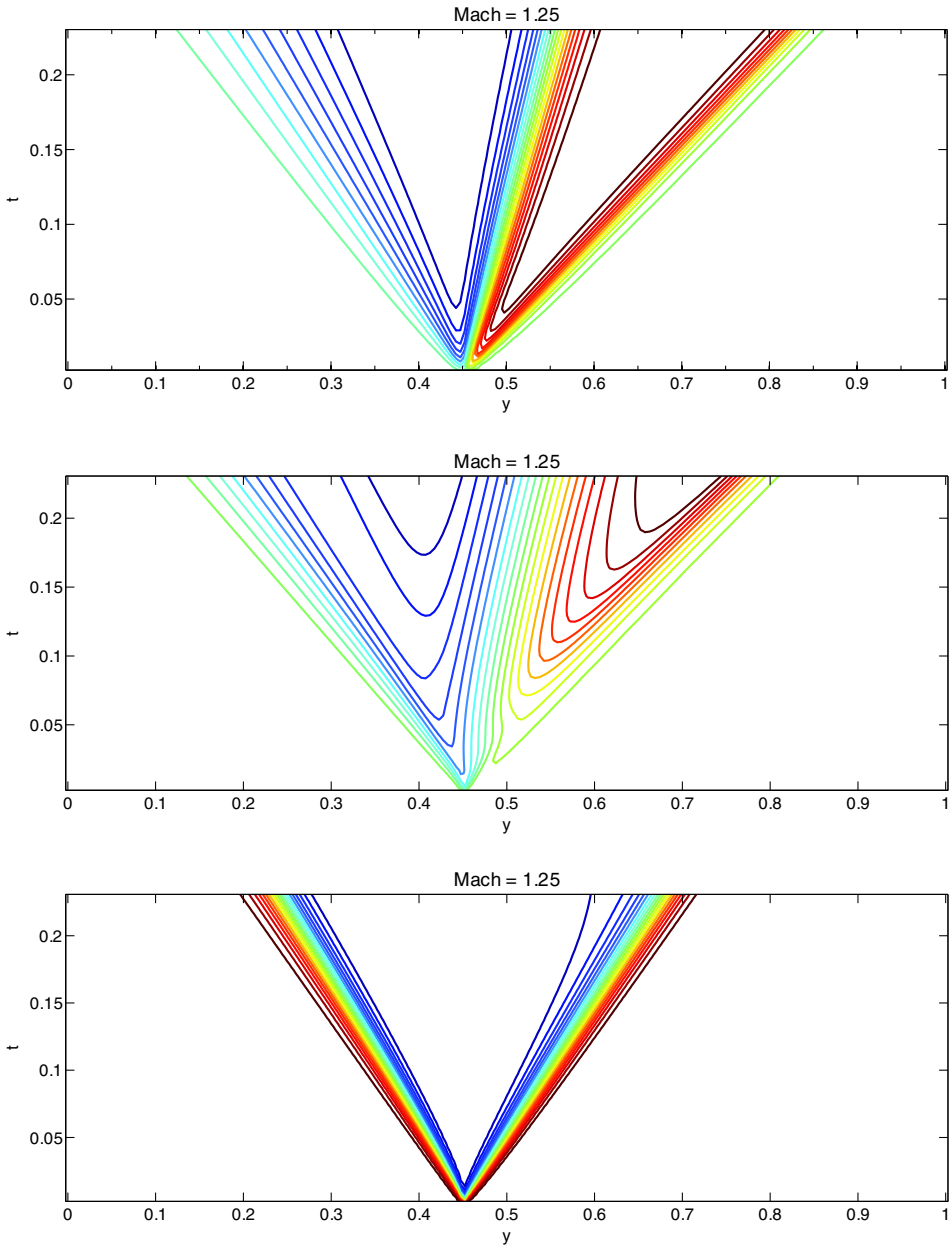


Fig. 2. Temperature contour plots (with equally-spaced contours) for the Riemann problem with $Ma = 1.25$. Top: $\kappa = 0$. Middle: $\kappa = 0.1$. Bottom: $\kappa = 10$. Low to high T is shown as blue to red contours. See Fig. 5 for a typical temperature range.

In Fig. 4 we present results for selected flow quantities with $\kappa = 0.01$ at $t = 0.23$. In the first row, the weights and velocity abscissas computed from the moments using quadrature are presented. Note that due to the relatively small value of κ all eight weights are nearly equal. (For $\kappa = 0$ all weights are equal.) Likewise, the U and W abscissas exhibit a high degree of symmetry, while the V abscissas are asymmetric due to the mean velocity in the y direction. Note that due to the diagonal form for $\mathbf{L}$ the velocity abscissa vectors lie at the eights corners of a cube in velocity phase space centered at the mean velocity. Hence each curve in the plots of the abscissas actually corresponds to four abscissas. In the second row of Fig. 4 the density $\rho = M^0$ and the mean velocities are plotted. As expected, the U and W components are null (due to symmetry)

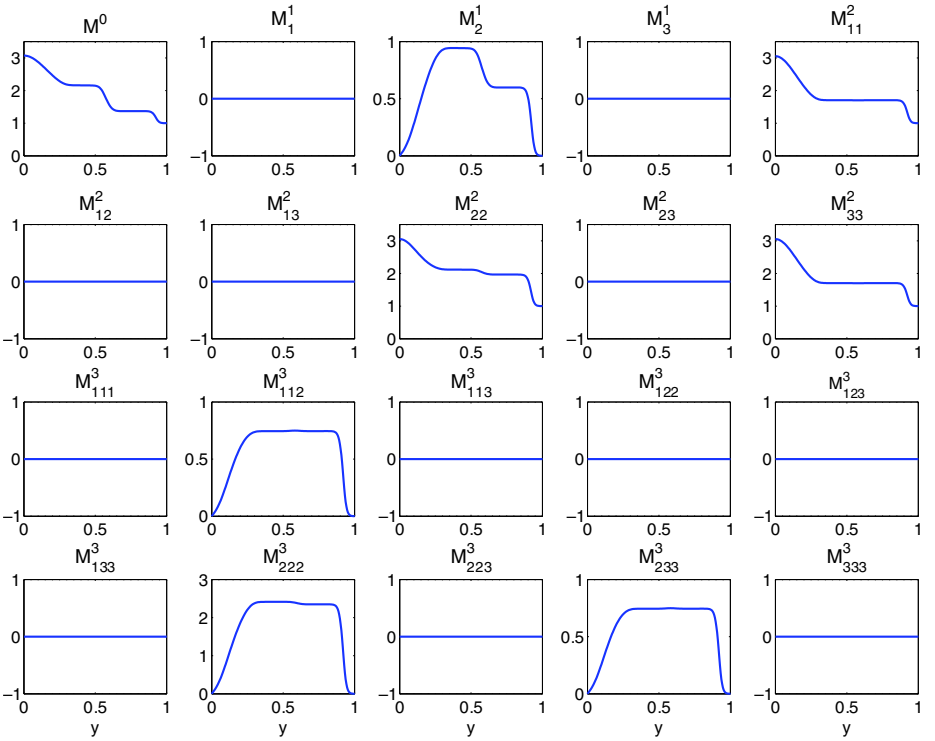


Fig. 3. Twenty transported moments in the Riemann problem with $Ma = 1.25$ and $\kappa = 0$ at $t = 0.23$. Note that due to symmetry 12 of the moments are null.

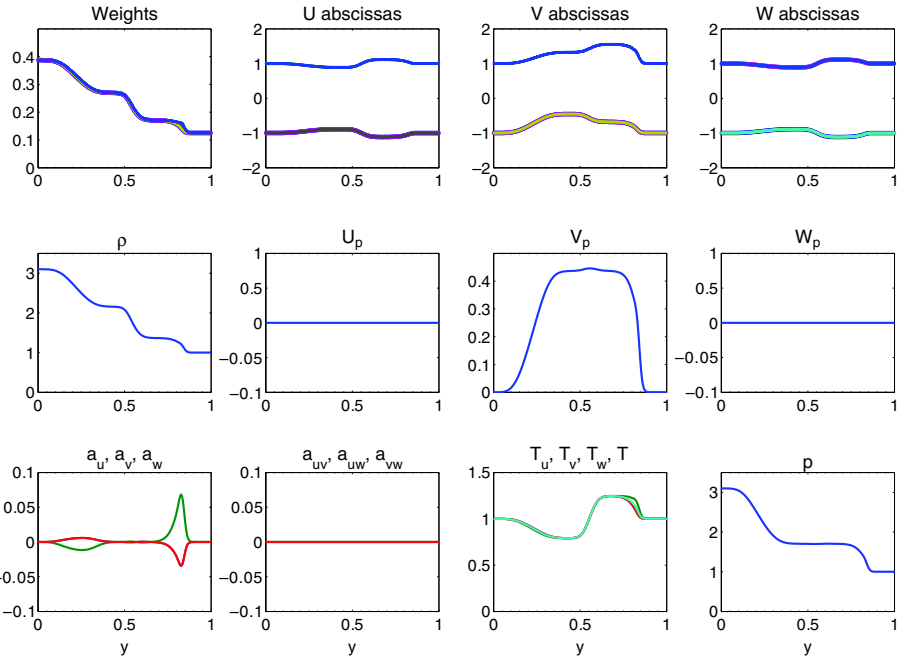


Fig. 4. The Riemann problem with $Ma = 1.25$ and $\kappa = 0.01$ at $t = 0.23$. The density ρ is shown in the second row, first column.

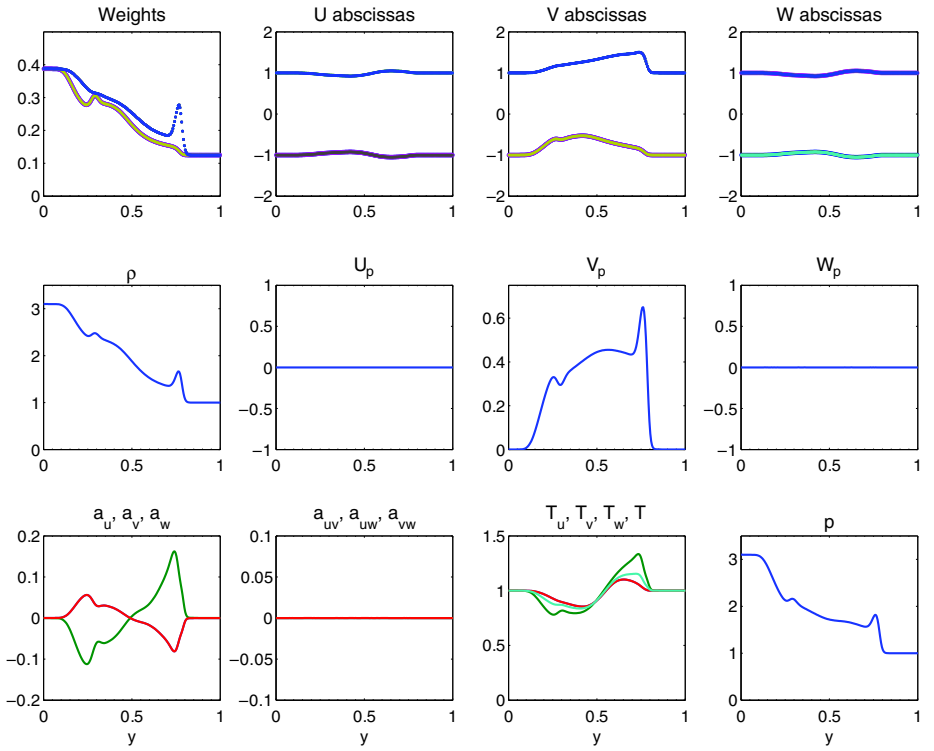


Fig. 5. The Riemann problem with $Ma = 1.25$ and $\kappa = 0.1$ at $t = 0.23$. The density ρ is shown in the second row, first column.

and the V component is positive between the expansion wave and the shock front. In the third row of Fig. 4 the anisotropy components of the velocity, defined by

$$a_{ij} = (\sigma_{ij}/T - \delta_{ij}) \quad i, j = u, v, w,$$

where $T = (T_u + T_v + T_w)/3$ and $T_i = \sigma_{ii}$, are plotted. The temperatures in each direction and pressure ($p = \rho T$) are also shown. Note that because κ is finite, the diagonal anisotropy components are non-zero in the shock wave and in the expansion fan. This non-equilibrium behavior is also observed in the temperature plots. The reader can compare the results in Fig. 4 to those presented in Fig. 10 of [4], and it can be clearly observed that the quadrature-based moment closure captures the essential features of finite Knudsen number flow.[12]

Results for progressively larger values of κ are shown in Figs. 5–7. For $\kappa = 0.1$ (Fig. 5) it can be seen that the weights are unequal – a clear sign that the flow is far from equilibrium. The weights corresponding to V abscissas moving in the positive direction exhibit a local maximum near $y = 0.75$, which is due to the formation of a delta-shock [7,15]. This delta-shock is caused by the form of the forward-moving V abscissa (i.e. the velocity is higher upstream). The formation of delta-shocks is a sign that the number of V abscissas available to describe the flow is insufficient. (Recall that we use only two abscissas in each direction.) A higher-order (than third) moment closure would be required to increase the number of quadrature nodes. For $\kappa = 1$ (Fig. 6) the delta-shock is not observed since particle collisions are relatively infrequent. The same is true for $\kappa = 10$ where the velocity abscissas remain very close to their initial values. For this case, the weights are transported to the left (right) with their own velocities. The shock wave seen with $\kappa = 0$ is completely absent and the flow is close to the free-molecular regime. Note that because the density function is represented by only two abscissas in each direction of phase space, the weights separate as two moving steps. By increasing the order of the quadrature, more abscissas would be introduced and the density field would be smoother.

[12] The Kundsen number for this case will be proportional to κ.

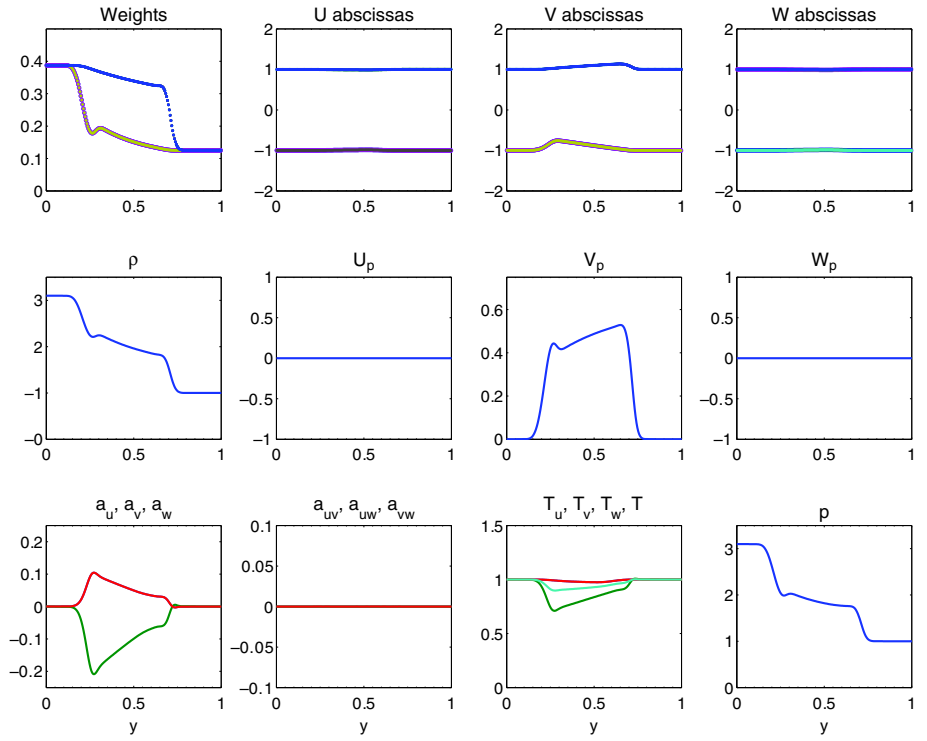


Fig. 6. The Riemann problem with $Ma = 1.25$ and $\kappa = 1$ at $t = 0.23$. The density ρ is shown in the second row, first column.

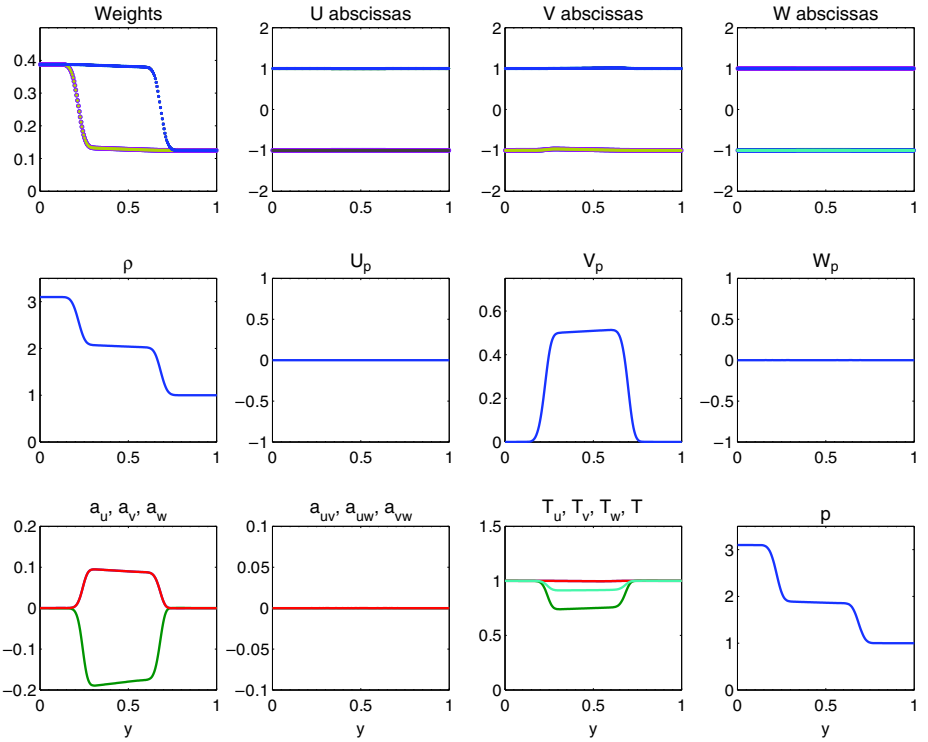


Fig. 7. The Riemann problem with $Ma = 1.25$ and $\kappa = 10$ at $t = 0.23$. The density ρ is shown in the second row, first column.

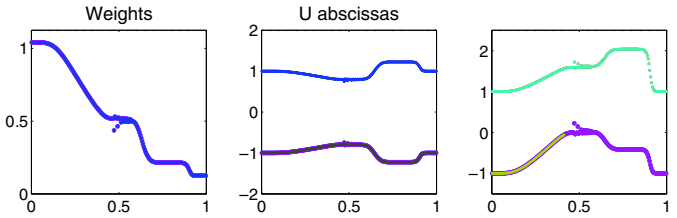
These observations are consistent our intuitive understanding of moment closures. As the density function moves farther away from equilibrium (i.e. larger Knudsen number), more moments will be required to accurately describe the flow. Note, however, that quadrature-based moment closures fail "gracefully" in the sense that the weights and abscissas (and hence the moments) always remain realizable over the entire range of Knudsen numbers. This is not the case for other third-order moment closures (e.g. [34,62]) where the corresponding density function becomes negative (i.e., the moments generate an unrealizable density function for sufficiently large Knudsen numbers).

In order to explore the range of applicability of the third-order quadrature-based moment closure, we have tested higher density ratios (i.e. higher Mach numbers) for the Riemann problem. In Fig. 8 we show results for $\rho_4/\rho_1 = 8.333$ where the lower velocity abscissas have a point with zero V velocity. It can be observed that this leads to small density fluctuations. For higher density ratios (e.g. hyper-sonic flows) no such oscillations are observed, even though all the V velocity abscissas are all positive behind the shock. The oscillations seen in Fig. 8 are thus related to abscissas being tangent to the zero V velocity line. As can be anticipated from the earlier figures, the onset of the density oscillations depends on κ. For example, in Fig. 9 the same density ratio is used with $\kappa = 0.1$ and the lower V abscissa remains negative. Note, however, that the delta-shock is even larger for this case. Finally, from the behavior exhibited in Fig. 7, we can observe that oscillations will never occur when $\kappa = \infty$.

6.2. *Impinging particle flow*

In order to add the effect of drag on particle motion, we will consider an example where the gas flow impinges on the plane $y = 0$:

$$V_f(y) = -\sin(\pi y) \quad -1 < y < 1.$$



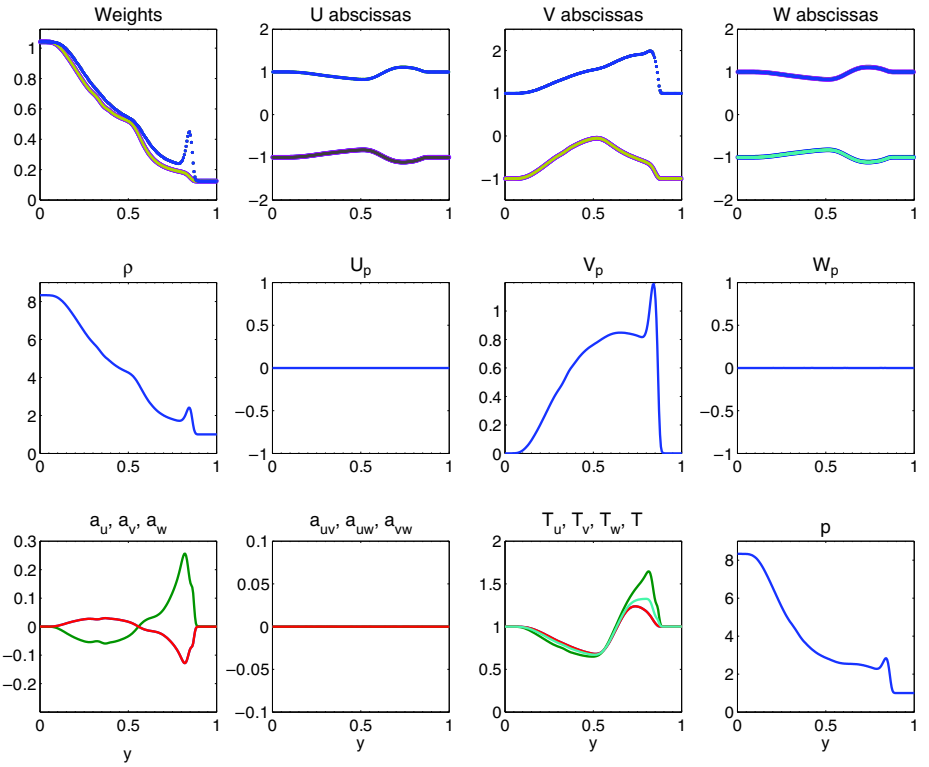


Fig. 9. The Riemann problem with density ratio $\rho_4/\rho_1 = 8.333$ and $\kappa = 0.1$ at $t = 0.23$. The density ρ is shown in the second row, first column.

The fluid-particle drag will be modeled by Stokes drag (Eq. (6)) with $\tau_p = 2$, which corresponds to a Stokes number of $St = \tau_p/\tau_f = 2$. For this Stokes number, the particles will not be able to decelerate fast enough to avoid crossing the centerline ($y = 0$) (i.e. they are underdamped). As shown in [19,20], the case with no collisions ($\kappa = \infty$) exhibits particle trajectory crossing (PTC) where the local V particle velocity is multi-valued. (See [32,33,39,40,44,45,67] for examples of this type of behavior in other non-collisional physical systems.) Essentially, in this limit, each velocity abscissa obeys its own conservation equation and the weights are advected without change. (In the numerical implementation, the weights change only due to numerical diffusion.) For finite κ, the weights and abscissas are coupled through the particle–particle collision term (making the computational problem considerably more challenging), and it is of great interest to see how the PTCs are altered (albeit in the elastic limit assumed by the BGK model).

In this example, the spatial grid and time step are handled the same way as in the Riemann problem. The initial particle velocities were set to zero mean with RMS values of 1×10^{-3} everywhere.[13] The initial weights are set to 1×10^{-10} everywhere except for $0.5 < |y| < 0.6$, where the weights are set to 1/8. This results in an $M^0(y)$ that represents two particle planes symmetric with respect to the origin and initially at rest. The particle planes are accelerated by the fluid drag towards the origin. At $t = 1.81$ the particle planes exhibit the maximum degree of overlap (i.e., if $\kappa = 0$ they are each centered at the origin). Note that because V_f is dependent on the initial particle location, all particles in the same plane are not accelerated uniformly. Thus, small amounts of particle velocity variance are created in each plane before collision. Nevertheless, this variance is negligible compared to that created when the particle planes overlap (due to the fact that they essentially have the opposite velocities).

Figs. 10–12 show example results for the number density ρ and the V_p particle velocity at two different times ($t = 1.81$ (maximum overlap) and $t = 2$). Note that by symmetry the mean velocity must be zero at $y = 0$.

[13] Non-zero RMS is needed to ensure that the velocity covariance matrix is positive definite.

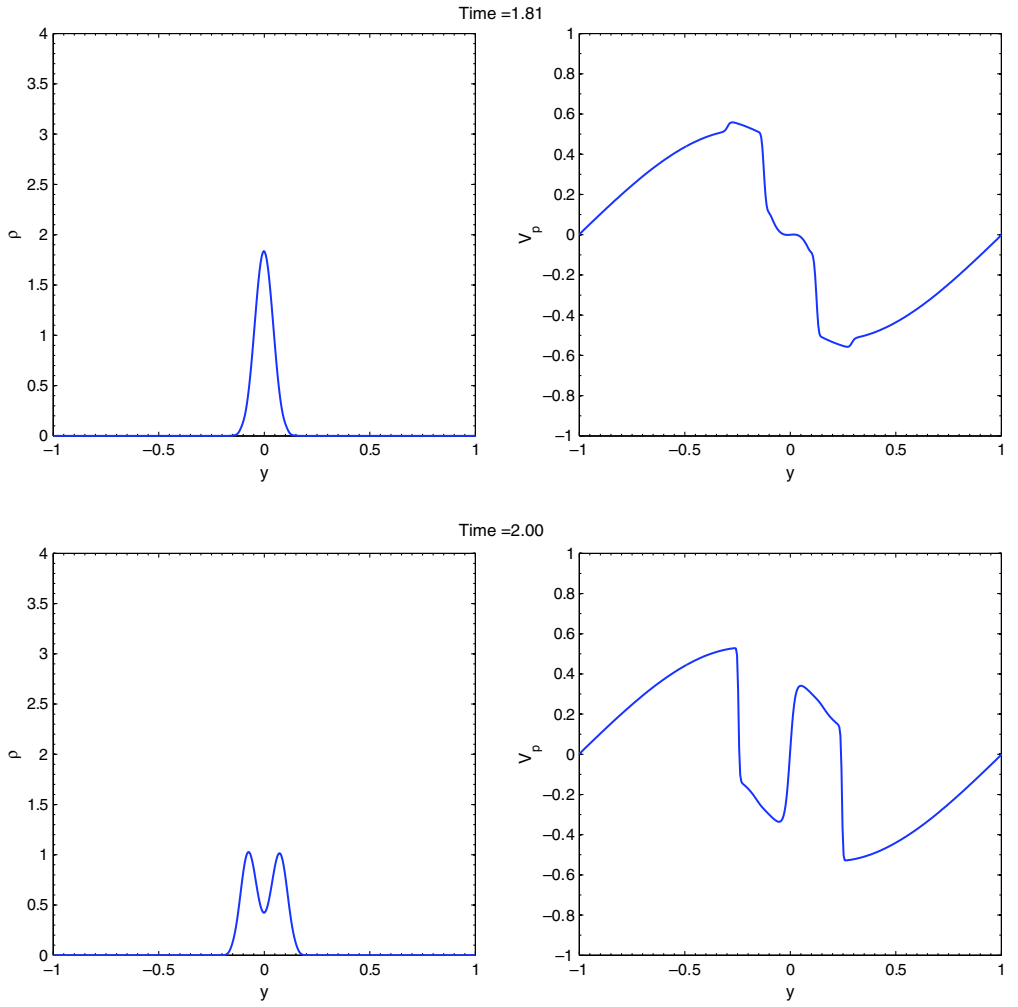


Fig. 10. Impinging particle planes with $\kappa = 1$. Left: number density. Right: V_p velocity of particles. Maximum overlap occurs at $t = 1.81$. At $t = 2$ the planes have almost completely crossed through each other for the first time.

Before collision, the mean velocity has the same sign as V_f for all κ, while after collision the number density and velocity depend strongly on κ. For $\kappa = 1$ (weakly collisional), the results are shown in Fig. 10. At $t = 1.81$ the mean density is roughly equivalent to two superimposed non-colliding planes, and at $t = 2$ the two planes are nearly separated. Note that the mean velocity at $t = 2$ is negative for the plane on the left and positive for the plane on the right. The two planes will continue to move apart until the drag term changes the signs of the corresponding mean velocity, after which the planes will move back and re-collide. The results for $\kappa = 0.1$ (moderately strong collisions) are shown in Fig. 11. At $t = 1.81$ one can observe that the peak in the number density is higher, indicating that the velocity has been significantly impacted by collisions. At $t = 2$ it can be seen that the two particle planes do not separate, rather they remain "stuck" together and only partially cross. Likewise, the velocity at $t = 2$ has a very different form near $y = 0$ than for $\kappa = 1$. This form suggests that the planes have partially "rebounded" off each other and reversed direction. As κ is further decreased, the tendency toward rebound becomes more and more pronounced. This can be seen from the results in Fig. 12 for $\kappa = 0.01$. Note that at $t = 1.81$ the number density in Fig. 12 has the form of two peaks smashed against a virtual wall located at the origin. In the Euler limit ($\kappa = 0$) the behavior is very similar to $\kappa = 0.01$.

Finally, note that the rebound velocity (i.e. the peak V_p velocity at the leading edge of the plane) grows very rapidly. For example, in the Euler limit (see Fig. 13), the rebound velocity is nearly $|V_p| = 4$ at $t = 2.1$ and continues to grow rapidly with time. Physically, the rebound velocity is generated by the transformation of

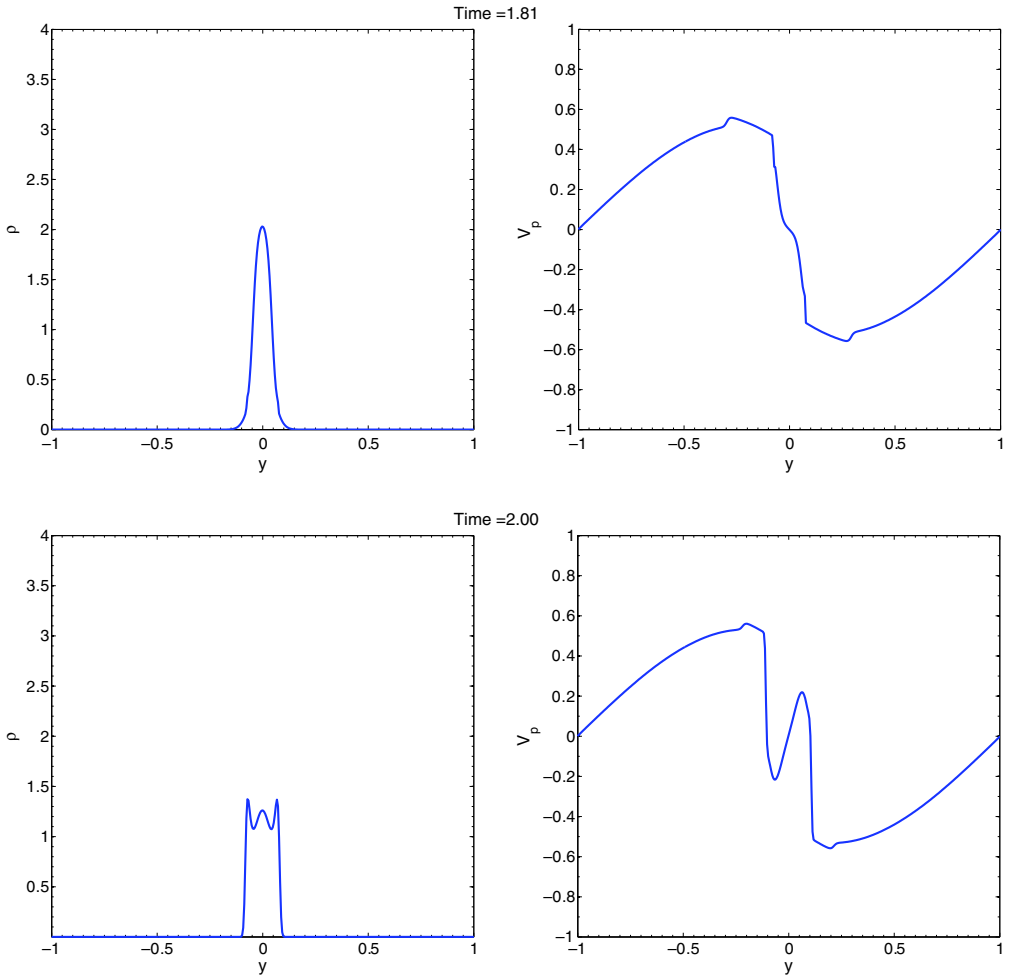


Fig. 11. Impinging particle planes with $\kappa = 0.1$. Left: number density. Right: V_p velocity of particles. Maximum overlap occurs at $t = 1.81$. At $t = 2$ the planes have only partially crossed through each other.

"thermal" energy created during the collision into translational energy. In other words, the maximum particle temperature T occurs at the peak rebound velocity, in front of which T is near zero. The rebound velocity thus corresponds to a shock wave traveling from the more dense to the more dilute particles. Because the collisions are assumed to be elastic, all of the energy in the original system must be radiated outward by the shock wave. Real particle systems have some degree of in-elasticity and thus it will be interesting to repeat these examples with a collision model that dissipates some thermal energy. Nevertheless, the numerical implementation of the quadrature-based moment method is robust and exhibits no anomalous behavior for this example.

6.3. Vertical channel flow

For this example, we consider a fully developed, wall-bounded channel flow where the statistics depend only on t and y [37,57,61]. The low solids concentration and large particle inertia allow us to neglect the fluctuating gas velocity and the influence of particles on the mean gas velocity [57]. The gas is flowing upward (positive x) against gravity with a given velocity $\mathbf{U}_f = [U_f(y), 0, 0]$ where[14]

[14] This approximation for the velocity profile was chosen to illustrate the method. A more accurate expression would be found by solving a turbulence model for the gas phase.

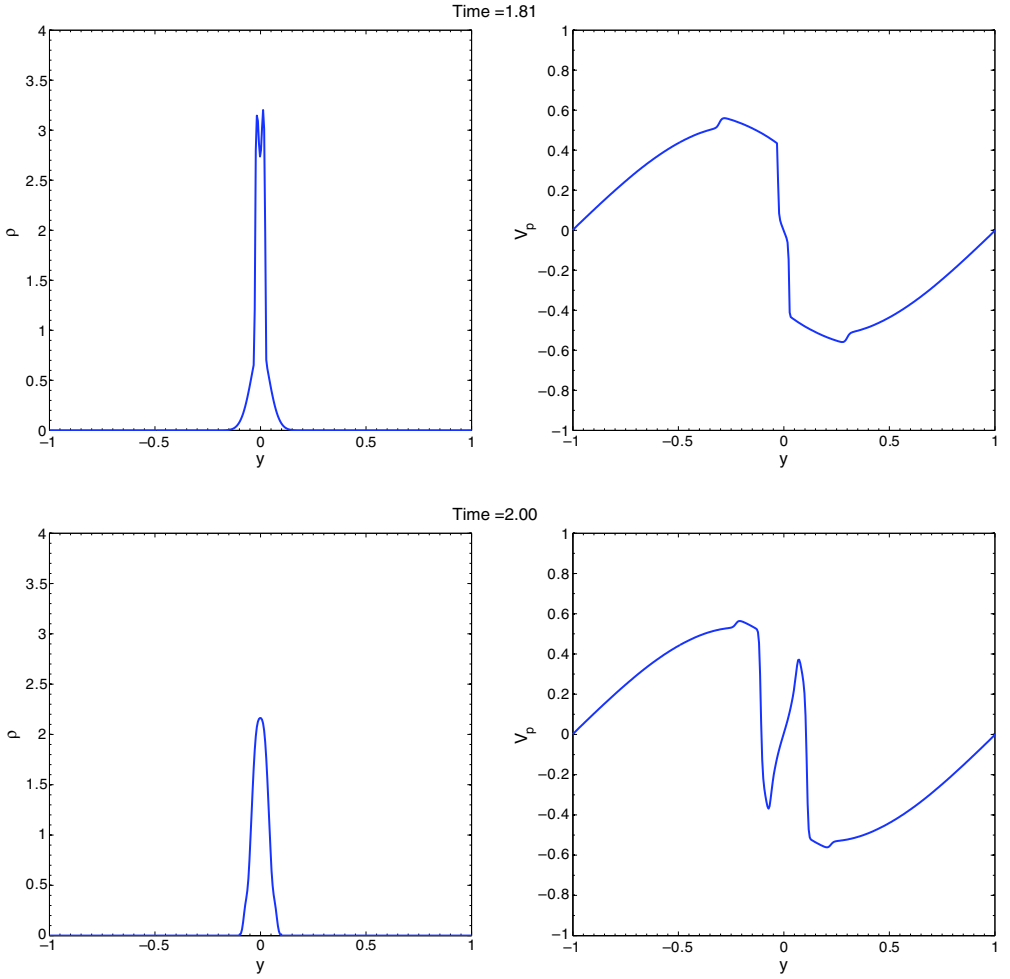


Fig. 12. Impinging particle planes with $\kappa = 0.01$. Left: number density. Right: V_p velocity of particles. Maximum overlap occurs at $t = 1.81$. At $t = 2$ the planes have begun to rebound after colliding.

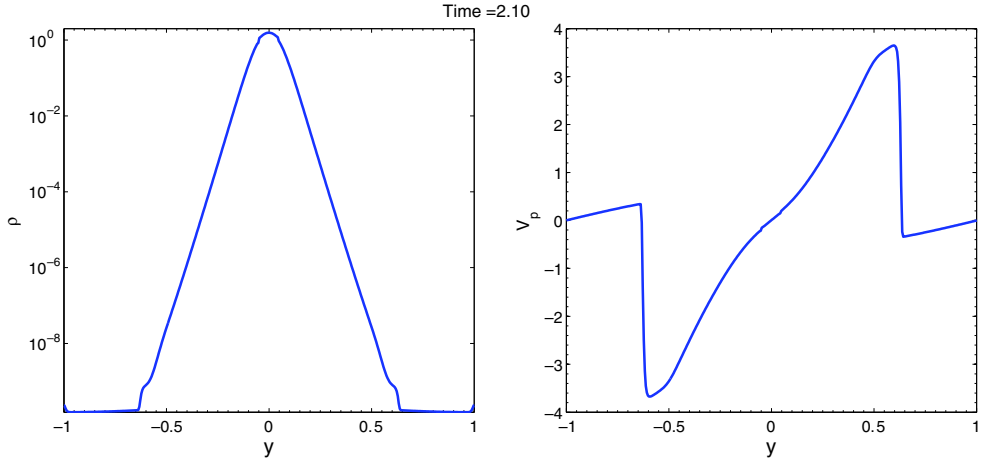


Fig. 13. Impinging particle planes in Euler limit at $t = 2.1$. Left: number density. Right: V_p velocity of particles.

$$U_f(y) = U_{max}[1 - (2y/W)^2]^{0.2}. \quad (63)$$

The channel width is $W = 4 \times 10^{-2}$ m. The maximum gas velocity is $U_{max} = 18$ m/s. We will be interested in solutions with reflective boundary conditions ($U_{1\alpha} \to -U_{1\alpha}$) at $y = -W/2$ and $y = W/2$, which correspond to purely elastic walls ($e_w = 1$). The physical properties of the gas phase are kinematic viscosity $\nu_f = 1.515 \times 10^{-5}$ m^2/s and density $\rho_f = 1.205$ kg/m^3. For the solid phase, two types of particles are used: (1) $\rho_p = 1032$ kg/m^3, $d_p = 1.5 \times 10^{-3}$ m; (2) $\rho_p = 1038$ kg/m^3, $d_p = 0.406 \times 10^{-3}$ m. These correspond to (1) particle volume $\mathrm{Vol}_p = 1.77 \times 10^{-9}$ m^3, particle mass $m_p = 1.82 \times 10^{-6}$ kg; and (2) $\mathrm{Vol}_p = 3.50 \times 10^{-11}$ m^3, $m_p = 3.64 \times 10^{-8}$ kg. The average volume fractions $\langle\alpha\rangle$ of interest are 10^{-3} and 4×10^{-2} [57]. The average number density is related to the average volume fraction by $\langle M^0\rangle = \langle\alpha\rangle/\mathrm{Vol}_p$.

For this example, the collision rate will be modeled by the product of the collision cross-section, the particle RMS velocity, and the number density:

$$\frac{1}{\tau} = \pi d_p^2 (3T)^{1/2} M^0 = 10.392 \frac{\alpha\sqrt{T}}{d_p},$$

where $\alpha = \pi M^0 d_p^3/6$ is the particle volume fraction and T is the particle temperature. In the following, we will make the moment transport equations dimensionless using $W/2$, U_{max}, and $\langle\alpha\rangle$. The dimensionless parameters appearing in the moment transport equations are $g^* = gW/(2U_{max}^2) = 6.06 \times 10^{-4}$, $\tau_p^* = 2\tau_p U_{max}/W = 900\tau_p$, $Re_p^* = d_p U_{max}/\nu_f = 1.188 \times 10^6 d_p$, and $\tau^* = 2\tau U_{max} V_p/(W\langle\alpha\rangle) = 4.811 d_p/(\sqrt{T^*}\langle\alpha\rangle)$, where $T^* = T/U_{max}^2$. For the two types of particles, the characteristic times can be written as (1) $\tau_p^* = 6344$, $Re_p^* = 1782$, $\tau^* = 7.217 \times 10^{-3}/(\sqrt{T^*}\langle\alpha\rangle)$ and (2) $\tau^*_p = 468.6$, $Re_p^* = 482.4$, $\tau^* = 1.953 \times 10^{-3}/(\sqrt{T^*}\langle\alpha\rangle)$. In dimensionless form, the flow domain of interest is $-1 \leqslant y \leqslant 1$ and $\mathbf{U}_f = [(1-y^2)^{0.2}, 0, 0]$. Because the terms on the right-hand sides of the moment transport equations are small relative to the left-hand sides, we can expect that this flow will be dominated by the transport terms. In particular, the quadrature-based closure of the fourth-order moments will have a strong influence on model predictions.

Note that by symmetry the following seven moments are null for this example:

$$M_3^1, \quad M_{13}^2, \quad M_{23}^2, \quad M_{113}^3, \quad M_{123}^3, \quad M_{223}^3, \quad M_{333}^3.$$

Thus it would suffice to consider only the 13 remaining moment equations. Nevertheless, we will solve for all 20 moments and use the symmetry condition to verify the code. Note also that the mean particle velocity and the velocity covariance matrix simplify to

$$\mathbf{U}_p = \begin{bmatrix} U_p \\ V_p \\ 0 \end{bmatrix} \quad \text{and} \quad \sigma_U = \begin{bmatrix} \sigma_u & \sigma_{uv} & 0 \\ \sigma_{uv} & \sigma_v & 0 \\ 0 & 0 & \sigma_w \end{bmatrix}. \quad (64)$$

The Cholesky decomposition matrix $\mathbf{L}$ will thus contain only one off-diagonal component (L_{12}). Finally, we should note that a particle Stokes number can be defined as

$$St = \tau_p^*/(1 + 0.15(Re_p^*)^{0.687}).$$

Thus, the Stokes numbers for the two particles sizes are (1) $St = 238$ and (2) $St = 40.9$. (Hereinafter we will refer to the two cases by their Stokes numbers.) While these large Stokes numbers justify neglecting the gas-phase velocity fluctuations, they also imply that the particles will behave as a rarefied (elastic) granular gas. Hence, non-equilibrium effects will be significant in the dilute limit (i.e. small $\langle\alpha\rangle$).

For this flow, a trivial steady state exists where the particle temperature T is null and the mean particle velocity is equal to the local slip velocity. For this steady state M^0 is time invariant. The quadrature-based moment closure predicts this steady state if the particle temperature is initially set to zero. On the other hand, if the simulation is initialized with $0 < T$, then the temperature will evolve to a non-zero state for T where U_p differs significantly from the slip velocity. Physically, a non-zero T will cause particles to collide with the walls. The reflective boundary condition then causes the temperature near the wall to increase rapidly (causing more wall collisions). In this system, the temperature is dissipated by the fluid drag whose characteristic dimension-

less time is St. Velocity fluctuations in the U component are generated by fluctuations in the V component acting on the U_p velocity gradient [57]. Thus, the collisions are important because they transfer σ_u into σ_v, which ensures the production of σ_u by the source term $\sigma_{uv}\mathrm{d}U_\mathrm{p}/\mathrm{d}y$. In other words, without collisions σ_{uv} would decay to zero, as would the particle temperature. As we shall see below, in dilute flows collisions are relatively infrequent. Thus, the final state will be reached relatively slowly (many characteristic times), making the problem relatively expensive to solve using time marching. Moreover, due to the large Stokes number (i.e. slow energy decay rate), the initial transients can be quite sensitive to the initial conditions. Essentially, the initial behavior is made up of traveling waves that reflect off of the walls until they are slowly damped by the drag.

In order to achieve an initial steady-state solution without excessively small time steps, we have first started the simulation with U_p constant, equal to the slip velocity at $y = 0$, and $T = 1 \times 10^{-4}$ (i.e. non-zero but smaller than the steady-state temperature). In addition, we have enhanced the collisions by setting $\langle\alpha\rangle = 0.5$, which stabilizes the initial wave dynamics. The simulation is then run to $t = 1000$ (i.e. many characteristic drag times) to achieve a steady-state solution that is reasonably close to desired one. A relatively good indicator of stationary is when $V_\mathrm{p}(y)$ is nearly zero. (As seen in the Riemann problem, waves cause transient behavior in V_p that is dissipated in this system by drag.) The desired solution is then found be lowering $\langle\alpha\rangle$ to the target value (i.e. increasing the collision time τ^*) and running until the final state is attained. Note that there is no guarantee that a steady-state solution exists for this problem. As described previously, the simulations are found to be grid independent with 400 equally-spaced grid cells. We should note, however, that the stationary solution (when it exists) can be adequately represented on a coarser grid.

We begin by considering the case of $St = 40.9$. For this case, a stationary solution was observed for $\langle\alpha\rangle = 0.02$, but not at $\langle\alpha\rangle = 0.01$. The latter yields a highly non-equilibrium, time-dependent solution with traveling density and V velocity waves. Results for the 20 moments are shown in Fig. 14 for $\langle\alpha\rangle = 0.02$.

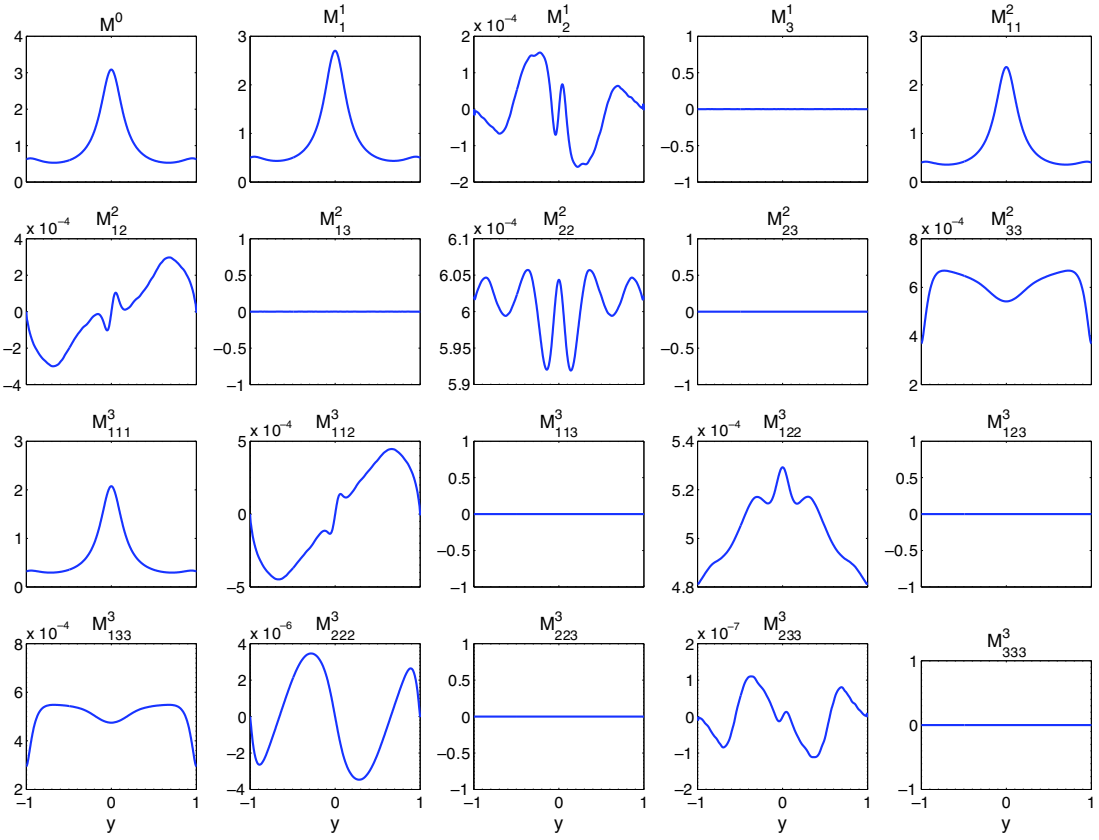
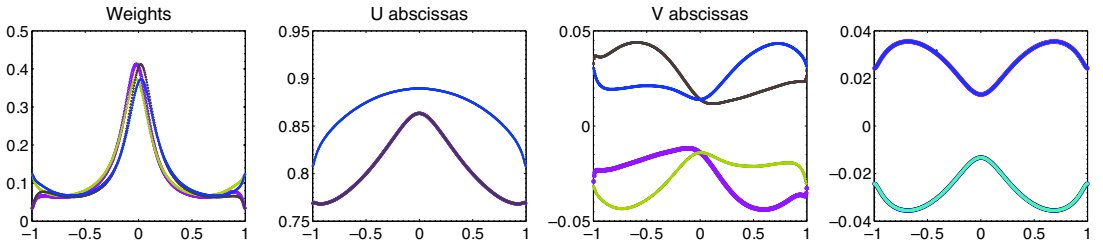


Fig. 14. Twenty transported moments in vertical channel flow with $St = 40.9$ and $\langle\alpha\rangle = 0.02$. Note that due to symmetry seven of the moments are null.



The seven moments that are null by symmetry are zero to machine precision. The other 13 moments are (anti)symmetric with respect to $y = 0$ and have non-trivial spatial dependencies. The corresponding weights, abscissas, and related flow statistics are shown in Fig. 15. Looking first at the weights, it can be seen that they are not all equal anywhere across the channel. As noted earlier, equal weights is a necessary condition for equilibrium (Maxwellian) behavior. For this flow, the unequal weights first appear near the walls due to reflections of the V velocity between incoming and outgoing particles. At larger $\langle\alpha\rangle$, the collisions are strong enough to force the weights to equilibrium in the center of the channel. However, at $\langle\alpha\rangle = 0.02$ the non-equilibrium behavior has reached the centerline as collisions are not frequent enough to overcome transport effects. Using the relative velocity and centerline temperature from Fig. 15, we can compute a Knudsen number[15] for this flow as $Kn \approx 4$. A key point from these observations is that even at 2% volumetric solids loading (which is not usually considered to be very dilute), the particle collisions are not frequent enough to keep the flow in equilibrium, and this is because the RMS velocity is low relative to molecular systems at the same holdup. Therefore, two-fluid models based on Chapman–Enskog-like expansions (i.e. $Kn \ll 1$) will not be valid for these particles until the holdup is considerably larger (at least 10% for this case). Moreover, even with $\langle\alpha\rangle = 0.1$, the non-equilibrium behavior near the walls is still significant.

The plots for the abscissas in Fig. 15 indicate that the moment-inversion method provides physically realistic values. In particular, the abscissas vary smoothly with y, which would not be the case if the transformation matrix $\mathbf{L}$ were defined using the eigenvectors of the covariance matrix. The correlation between U and V is evident from a_{uv} and from the V abscissas. The number density ($\rho = \alpha/\langle\alpha\rangle$) has a peak value at the centerline. We should note that the peak value of ρ decreases with decreasing $\langle\alpha\rangle$. The mean vertical solids velocity U_p is near

0.8 at the walls and peaks at the center (i.e. the particle wall-slip velocity is significant). The mean horizontal solids velocity V_p is very small, but non-zero. Since at steady state one would expect V_p to be null, this non-zero value suggests that the system is still (slowly) approaching the stationary state. The anisotropy coefficients and the temperatures all show significant non-equilibrium values. Finally, the pressure ($p = \rho T$) has a form very similar to M_{33}^2 and suggests that ρ is nearly proportional to $1/T$, except near the walls and the centerline.

The flow behavior observed for $\langle\alpha\rangle < 0.02$ is time dependent and takes the form of density waves (or delta-shocks) that traverse back and forth across the channel. These waves appear to be correlated with V velocity "shocks" similar to the one observed in Fig. 5 for the Riemann problem. For the channel flow, the reflective wall boundary conditions send the shock back into the channel, and eventually the entire channel is filled with such structures. We should note that similar behavior has been reported in 1D two-fluid model simulations (e.g. [3]) that use a lower-order moment closure (i.e. equivalent to the compressible flow equation). In the present context, we know that the transient behavior starts when the flow is significantly non-equilibrium at the centerline. Because our moment closure stops at third order, we are unable to capture strongly non-equilibrium behavior (unless we extend the closure to higher order by increasing the number of nodes). We are thus inclined to interpret the unsteady behavior as unphysical (although mathematically correct). This interpretation is consistent with earlier work [20] where a two-node quadrature model was compared to Lagrangian simulations for a particle-laden Taylor–Green flow. For that flow, delta-shocks were observed in the Eulerian model at moderately large Stokes numbers that were not present in the Lagrangian simulations.

We next consider the case of $St = 238$. For this flow, unsteady behavior is observed for $\langle\alpha\rangle < 0.20$, presumably because the larger Stokes number decreases the energy dissipation due to drag. Example results for $\langle\alpha\rangle = 0.19$ are shown in Fig. 16. The large Stokes number creates a significant slip velocity ($|U_p - U_f| \approx 0.38$) at the centerline, as well as a large wall-slip velocity. The small amplitude traveling waves are clearly seen in V_p. The amplitude of these waves rapidly increase with decreasing $\langle\alpha\rangle$. For this example, the weights are more nearly equal, indicating that the flow is not too far from equilibrium. This can also be seen from the diagonal anisotropy coefficients whose magnitudes are approximately 10 times smaller than in Fig. 15. Likewise, the temperatures are nearly equal and the pressure is almost constant. In other words, the density ρ is inversely

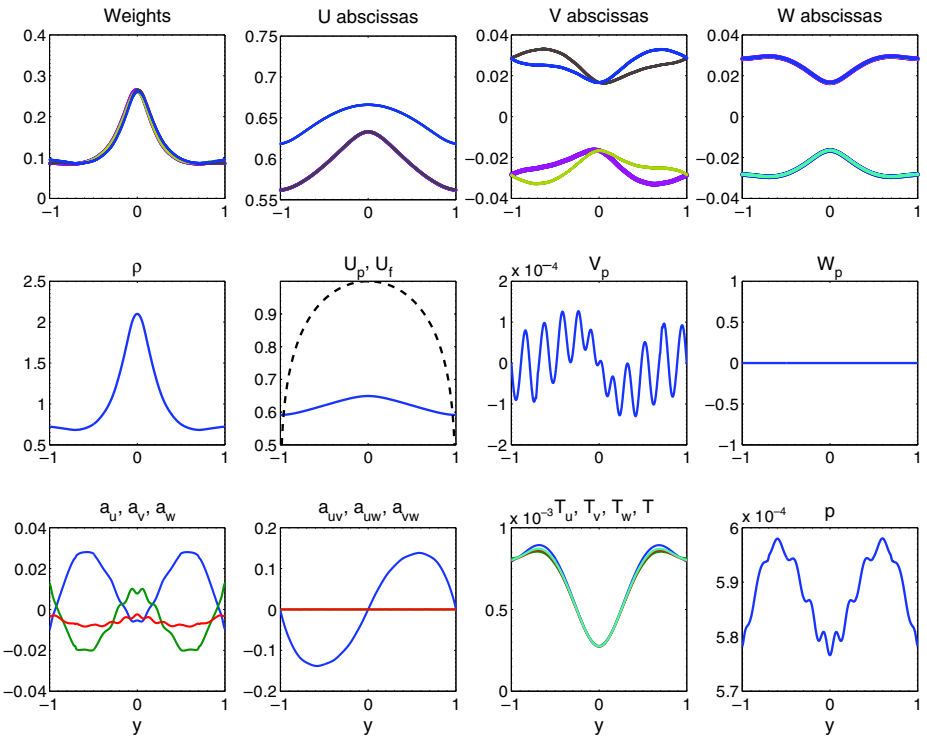


Fig. 16. Dimensionless variables in vertical channel flow with $St = 238$ and $\langle\alpha\rangle = 0.19$.

proportional to T. Finally, we should note that this case would suggest that the unsteady behavior is more related to the degree of energy dissipation through drag than to non-equilibrium moments. In real particulate systems, particle–particle and particle-wall collisions are inelastic, thereby leading to other routes for energy dissipation. We will explore this subject in more detail by using a dissipative collision term and dissipative wall boundary conditions in future work.

7. Conclusions

We have derived a quadrature-based third-order moment method for computing dilute gas-particle flows governed by the kinetic equation with gravity, fluid drag and particle–particle collisions. Overall, the proposed numerical scheme appears to be robust and yields physically realistic results for the examples considered (Riemann shock problem, impinging jets, and channel flow). The key characteristics of the proposed algorithm are as follows.

(1) Unlike previous work on moment methods [33,39,44,56,67] and on level sets [40,45] for representing multi-valued solutions to the kinetic equation, the quadrature-based method provides a consistent approximation of the density function and the collision term for continuous distributions in velocity phase space. In other words, much like DSMC [6], quadrature methods represent a continuous density function by a finite sets of "particles" (i.e. nodes). The weights and locations (abscissas) of each node are determined by forcing them to agree with as many moments as possible. (In this work, a maximum of 14 moments up to third order.) While we have only considered closure at third order (i.e. two quadrature nodes in each direction of velocity phase space), extension to higher order is possible and will be investigated in future work.

(2) The closure of nonlinear terms in the moment transport equations is based on a quadrature representation of the density function. This feature allows us to provide a consistent closure for the fluid-particle drag term and the collision term [64]. However, any other nonlinear term in the kinetic equation could be closed in a similar manner.

(3) A moment-inversion algorithm was developed based on Cholesky decomposition of the velocity covariance matrix and repeated application of one-dimensional quadrature [49]. Because the latter is well defined and computationally efficient to high order,[16] the extension to higher-order moment closures should be possible. The Cholesky decomposition was found to be preferable to the eigenvectors [70] because the latter do not vary continuously with the components of the covariance matrix, leading to non-continuous fluxes in physical space. At this point, the only open question concerning extension of the inversion algorithm to more nodes appears to be what is the best choice of the third- and higher-order moments needed to define the linear system used to solve for the weights. We should note that this is not a trivial question, and that the response may be problem dependent. For example, for non-colliding particles a set of moments consistent with the symmetry of traveling waves might be appropriate [56], while for colliding particles another set of moments that better captures deviations from Maxwellian distribution might be more accurate [28].

(4) The spatial transport terms in the moment equations are treated using a kinetic description, and hence a gradient-diffusion closure is not invoked to close these terms. This feature makes quadrature-based moment closures distinctly different from all other moment-based closures valid for small perturbations from equilibrium. In fact, because the weights are always non-negative, the quadrature-based moment closure remains realizable for arbitrary Knudsen number. This is an important property because it allows us to define the kinetic-based fluxes for arbitrary-order moments.

(5) Because quadrature-based closures use directly the moment transport equations, and not a discretized version of the Boltzmann equation, they allow us to conserve mass, momentum and energy for arbitrary Knudsen number (including $Kn = 0$). Any conservation errors will result from the numerical method used to discretize the moment equations. Moreover, unlike DSMC and other statistical methods [36],

[16] We have gone as high as 8-node quadrature for uni-variate cases in previous work, which requires moments up to 15th order.

quadrature-based closure do not suffer from statistical noise (even in the limit of one node). In this work we have used a first-order kinetic scheme to compute the moment fluxes, but higher-order schemes should also be applicable [20]. An important consideration when advecting moments with higher-order schemes is that the moments must remain realizable [50,68]. This property must be demonstrated before higher-order schemes can be used with the velocity moment equations.

(6) In principle, quadrature-based moment closures can be used to solve any flow problem that is modeled by a "Boltzmann-like" equation for the velocity distribution function. The only requirement is that the equation be closed in terms of the density function (i.e., the collision term involving the two-point density function must first be closed in terms of f). Although we have employed the BGK approximation (which is closed at the level of the moments), any one-point collision term could be used with quadrature-based closures. For example, the Boltzmann hard-sphere collision term has been used by Vedula and Fox [64].

(7) While direct Boltzmann solvers [4,12,13,52,53] should be more accurate than quadrature-based moment closures for difficult problems, the degree of accuracy attained at equivalent computational cost will likely favor quadrature-based methods. (All simulation results presented here can be performed on a laptop computer.) Nevertheless, further work is needed to investigate the range of applicability of quadrature-based moment closures (e.g. higher Mach numbers in the Riemann problem, two-way coupling in gas-particle flows, etc.) and to optimize the numerical methods.

Acknowledgments

This work was supported by a Grant (DE-FC26-07NT43098) from the National Energy Technology Laboratory (NETL) of the US Department of Energy.

Appendix A. Linear transformations and translations of moment sets

Consider a non-singular linear transformation $\mathbf{L}$ of a vector $\boldsymbol{\phi}$:

$$\boldsymbol{\phi}^* = \mathbf{L}\boldsymbol{\phi} \quad \Longleftrightarrow \quad \begin{bmatrix} \phi_1^* \\ \phi_2^* \\ \phi_3^* \end{bmatrix} = \begin{bmatrix} L_{11} & L_{12} & L_{13} \\ L_{21} & L_{22} & L_{23} \\ L_{31} & L_{32} & L_{33} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}. \tag{65}$$

Let $m^*(\mathbf{k})$ denote the moments of $\boldsymbol{\phi}^*$ and $m(\mathbf{k})$ denote the moments of $\boldsymbol{\phi}$ for a particular set of exponents $\mathbf{k} = (k_1, k_2, k_3)$. Using multinomial expansions, it can be easily shown that m^* is related to m by

$$\begin{aligned} m^*(k_1^*, k_2^*, k_3^*) = & \sum_{j_1=0}^{k_1^*} \sum_{i_1=0}^{j_1} \sum_{j_2=0}^{k_2^*} \sum_{i_2=0}^{j_2} \sum_{j_3=0}^{k_3^*} \sum_{i_3=0}^{j_3} \binom{k_1^*}{j_1} \binom{j_1}{i_1} \binom{k_2^*}{j_2} \binom{j_2}{i_2} \binom{k_3^*}{j_3} \binom{j_3}{i_3} L_{11}^{k_1^*-j_1} L_{12}^{j_1-i_1} L_{13}^{i_1} L_{21}^{k_2^*-j_2} L_{22}^{j_2-i_2} \\ & \times L_{23}^{i_2} L_{31}^{k_3^*-j_3} L_{32}^{j_3-i_3} L_{33}^{i_3} \delta_{k_1, k_1^*+k_2^*+k_3^*-j_1-j_2-j_3} \delta_{k_2, j_1+j_2+j_3-i_1-i_2-i_3} \delta_{k_3, i_1+i_2+i_3} m(k_1, k_2, k_3), \end{aligned} \tag{66}$$

where $\delta_{k,j}$ is the Kronecker delta. Letting $\mathbf{m}^*$ and $\mathbf{m}$ denote column vectors containing the distinct moments in a given moment set:

$$\mathbf{m}^* = \begin{bmatrix} m^*(0,0,0) \\ m^*(1,0,0) \\ m^*(0,1,0) \\ \vdots \end{bmatrix}, \quad \mathbf{m} = \begin{bmatrix} m(0,0,0) \\ m(1,0,0) \\ m(0,1,0) \\ \vdots \end{bmatrix}, \tag{67}$$

we can observe that Eq. (66) defines a square transformation matrix ($\mathbf{m}^* = \mathbf{M}\mathbf{m}$) with the following properties:

(1) $\mathbf{M}$ is full rank.
(2) $\mathbf{M} = \mathrm{diag}(\mathbf{M}_0, \mathbf{M}_1, \ldots)$ is block diagonal where the size of the square block $\mathbf{M}_\gamma$ equals the number of moments of order γ.
(3) $\mathbf{M}_\gamma$ will be diagonal if and only if $\mathbf{L}$ is diagonal.

Letting $\mathbf{m}^*_\gamma$ ($\mathbf{m}_\gamma$) denote of the components of $\mathbf{m}^*$ ($\mathbf{m}$) corresponding to moments of order γ, it then follows that $\mathbf{m}^*_\gamma = \mathbf{M}_\gamma \mathbf{m}_\gamma$. In other words, the moments of $\boldsymbol{\phi}^*$ of order γ are a linear combination of the moments of $\boldsymbol{\phi}$ of order γ. In general, unless $\mathbf{L}$ is diagonal, a *particular* moment of $\boldsymbol{\phi}^*$ of order γ will be a linear combination of *all* moments of $\boldsymbol{\phi}$ of order γ.

Finally, we can note that a simple translation:

$$\boldsymbol{\phi}^\dagger = \boldsymbol{\phi} + \boldsymbol{\mu} \quad \Longleftrightarrow \quad \begin{bmatrix} \phi_1^\dagger \\ \phi_2^\dagger \\ \phi_3^\dagger \end{bmatrix} = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix} + \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix}, \tag{68}$$

relates the moments sets by

$$m^\dagger(k_1^\dagger, k_2^\dagger, k_3^\dagger) = \sum_{k_1=0}^{k_1^\dagger} \sum_{k_2=0}^{k_2^\dagger} \sum_{k_3=0}^{k_3^\dagger} \binom{k_1^\dagger}{k_1} \binom{k_2^\dagger}{k_2} \binom{k_3^\dagger}{k_3} \mu_1^{k_1^\dagger - k_1} \mu_2^{k_2^\dagger - k_2} \mu_3^{k_3^\dagger - k_3} m(k_1, k_2, k_3). \tag{69}$$

Thus a moment of $\boldsymbol{\phi}^\dagger$ of order γ will be a linear combination of *only one* moment of $\boldsymbol{\phi}$ of order γ. However, in general, it will depend on lower-order moments of $\boldsymbol{\phi}$. Hence, a linear transformation combined with a translation will couple all moments of order γ and smaller. The moment sets used to define the quadrature formulas in the main text can be computed by successive application of Eqs. (69) and (66).

Appendix B. Relationship between the quadrature-based moment method and off-lattice Boltzmann methods

As discussed in the main text, quadrature-based moment methods determine the weights and velocity abscissas by forcing agreement with lower-order velocity moments. In off-lattice Boltzmann methods (oLBM) [1] the velocity abscissas are *fixed* (e.g., roots of a Hermite polynomial) and only the weights are allowed to vary. Note that the examples considered in the main text correspond to *non-isothermal* flows, which are difficult to treat with LBM [41]. Isothermal (or more accurately low Mach number) cases can be treated by fixing the particle temperature in the equilibrium distribution f_{eq}. For simplicity, we will consider only the one-dimensional case here.[17] In order to distinguish between the two methods, we will denote the oLBM velocities by c_α and the weights by f_α, where $\alpha \in (1, \ldots, N)$ denotes the nodes. We can then define the moment of order k by

$$m_k(x,t) = \sum_{\alpha=1}^{N} c_\alpha^k f_\alpha(x,t), \tag{70}$$

where the c_α are fixed. Note that there are N weights and hence exactly N moments can be reproduced by oLBM (i.e., $k \in (0, \ldots, N-1)$). A *discrete* Boltzmann equation must then be derived for computing the unknown weights. This is most easily accomplished using DQMOM [46,29] as described next.[18]

Starting from the one-dimensional Boltzmann equation with BGK collisions:

$$\partial_t f + \partial_x (cf) = \frac{1}{\tau}(f_{\text{eq}} - f), \tag{71}$$

the corresponding moment equations are

$$\partial_t m_k + \partial_x m_{k+1} + \frac{1}{\tau} m_k = \frac{1}{\tau} m_k^{\text{eq}}, \tag{72}$$

where m_k^{eq} are the known moments of the Maxwellian distribution. Substituting Eq. (70) into (72) yields

[17] The extension to higher-dimensions is straightforward [46], but the notation is cumbersome.

[18] DQMOM was derived to find transport equations for both the weights and abscissas. However, it is straightforward to modify the procedure for fixed abscissas.

$$\sum_{\alpha=1}^{N} c_\alpha^k \left(\partial_t f_\alpha + c_\alpha \partial_x f_\alpha + \frac{1}{\tau} f_\alpha \right) = \frac{1}{\tau} m_k^{\mathrm{eq}}, \tag{73}$$

which must hold for $k \in (0, \ldots, N-1)$. Defining the discrete Boltzmann equation by

$$\partial_t f_\alpha + c_\alpha \partial_x f_\alpha = \frac{1}{\tau} (f_\alpha^{\mathrm{eq}} - f_\alpha), \tag{74}$$

we see that the equilibrium velocity weights must satisfy

$$\sum_{\alpha=1}^{N} c_\alpha^k f_\alpha^{\mathrm{eq}} = m_k^{\mathrm{eq}}, \tag{75}$$

which is a linear system of the form $\mathbf{V}_N [f_\alpha^{\mathrm{eq}}] = [m_k^{\mathrm{eq}}]$ where $\mathbf{V}_N$ is an $N \times N$ Vandermonde matrix. For example,

$$\mathbf{V}_2 = \begin{bmatrix} 1 & 1 \\ c_1 & c_2 \end{bmatrix} \quad \text{and} \quad \mathbf{V}_3 = \begin{bmatrix} 1 & 1 & 1 \\ c_1 & c_2 & c_3 \\ c_1^2 & c_2^2 & c_3^2 \end{bmatrix}. \tag{76}$$

In the linear system, $[m_k^{\mathrm{eq}}]$ is the vector of equilibrium moments for $k \in (0, \ldots, N-1)$. If the velocities are distinct, then $\mathbf{V}_N$ is nonsingular. It then follows that the equilibrium velocity weights are uniquely determined by

$$[f_\alpha^{\mathrm{eq}}] = \mathbf{V}_N^{-1} [m_k^{\mathrm{eq}}]. \tag{77}$$

Thus, for any distinct set of velocities c_α, Eq. (74) can be solved numerically using the kinetic-based algorithm discussed in the main text (i.e., the velocities need not lie on a regular grid in velocity space). The resulting oLBM will be well defined as long as the f_α remain non-negative. Note that there is no guarantee that the weights will remain non-negative for an arbitrary choice of c_α (or even for a highly isotropic choice like the roots of Hermite polynomial). In contrast, the quadrature-based moment method introduced in the main text is guaranteed to produce non-negative weights [49].[19] In short, a key feature of the quadrature-based method is that the velocity abscissas adapt to the underlying velocity distribution function, including cases were the mean velocity is much greater than the speed of sound. This is not a feature shared by oLBM where the velocities are fixed.

It is well known [41] that one-dimensional LBM with $N = 2$ does not work. From Eq. (77) we see that[20]

$$\begin{bmatrix} f_1^{\mathrm{eq}} \\ f_2^{\mathrm{eq}} \end{bmatrix} = \frac{1}{c_2 - c_1} \begin{bmatrix} c_2 & -1 \\ -c_1 & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \rho u \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}; \tag{78}$$

hence the collision term will not affect the discrete Boltzmann equation. This is, of course, entirely consistent with the fact that momentum is conserved. In contrast, with $N = 2$ the quadrature-based moment method used in the main text is well defined because, by letting the velocity abscissas vary with space and time, the velocity moments up to third order $(2N-1)$ can be controlled. In order to achieve a well-defined discrete Boltzmann equation with oLBM, we must have $N \geqslant 3$ so that moments up to at least second order are controlled [1].[21] With $N = 3$, the quadrature-based moment method controls velocity moments up to fifth order.

References

[1] A. Bardow, I.V. Karlin, A.A. Guzev, Multispeed models in off-lattice Boltzmann simulations, Physical Review E 77 (2008) 025701(R).
[2] J.-D. Benamou, Big ray tracing: multivalued travel time field computation using viscosity solutions of the Eikonal equation, Journal of Computational Physics 128 (4) (1996) 463–474.

[19] The abscissas found with QMOM using the moments from a Gaussian distribution centered at zero with unit variance correspond to the roots of a Hermite polynomial of order $N+1$. As shown in the main text, for cases not centered at zero and with non-unit variance, the abscissas are simply translated by the mean and rescaled.

[20] The first two moments are $\rho = f_1 + f_2$ and $\rho u = f_1 c_1 + f_2 c_2$.

[21] It is relatively straightforward to find analytical expressions for f_α^{eq} for small values of N, especially for highly isotropic choices [1]. However, in practice, $\mathbf{V}_N^{-1}$ can be precomputed and $f_\alpha{}^{\mathrm{eq}}$ can be found from Eq. (77).

[3] S. Benyahia, M. Syamlal, T.J. O'Brien, Study of the ability of multiphase continuum models to predict core-annulus flow, AIChE Journal 94 (2007) 2459–2568.
[4] A.E. Beylich, Solving the kinetic equation for all Knudsen numbers, Physics of Fluids 12 (2000) 444–465.
[5] P.L. Bhatnagar, E.P. Gross, M. Krook, A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems, Physical Reviews 94 (1954) 511–525.
[6] G.A. Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Clarendon, Oxford, 1994.
[7] F. Bouchut, On zero pressure gas dynamics, Advances in Kinetic Theory and Computing. Series on Advances in Mathematics for Applied Sciences 22 (1994) 171–190.
[8] F. Bouchut, S. Jin, X.T. Li, Numerical approximations of pressureless gas and isothermal gas dynamics, SIAM Journal of Numerical Analysis 41 (2003) 135–158.
[9] Y. Brenier, L. Corrias, A kinetic formulation for multibranch entropy solutions of scalar conservation laws, Annales de l'Institut Henri Poincaré 15 (1998) 169–190.
[10] J.E. Broadwell, Shock structure in a simple discrete velocity gas, The Physics of Fluids 7 (8) (1964) 1243–1247.
[11] N. Caraman, J. Borée, O. Simonin, Effect of collisions on the dispersed phase fluctuation in a dilute tube flow: experimental and theoretical analysis, Physics of Fluids 15 (2003) 3602–3612.
[12] J.A. Carrillo, A. Majorana, F. Vecil, A semi-Lagrangian deterministic solver for the semiconductor Boltzmann–Poisson system, Communications in Computational Physics 2 (5) (2007) 1027–1054.
[13] C. Cercignani, The Boltzmann Equation and its Applications, Springer, New York, 1988.
[14] S. Chapman, T.G. Cowling, The Mathematical Theory of Nonuniform Gases, Cambridge University Press, Cambridge, 1970.
[15] G.-Q. Chen, H. Liu, Formation of δ-shocks and vacuum states in the vanishing pressure limit of solutions to the Euler equations for isentropic fluids, SIAM Journal of Mathematical Analysis 34 (2003) 925–938.
[16] C.K. Chu, Kinetic-theoretic description of the formation of a shock wave, The Physics of Fluids 8 (1965) 12–22.
[17] P. Degond, S. Jin, L. Mieussens, A smooth transition model between kinetic and hydrodynamic equations, Journal of Computational Physics 209 (2005) 665–694.
[18] S. Deshpande, A second order accurate, kinetic theory based, method for inviscid compressible flows. Technical Report NASA Langley, vol. 2613, 1986.
[19] O. Desjardins, R.O. Fox, P. Villedieu, A quadrature-based moment closure for the Williams spray equation, Proceedings of the Summer Program 2006, Center for Turbulence Research, Stanford, 2006, pp. 223–234.
[20] O. Desjardins, R.O. Fox, P. Villedieu, A quadrature-based moment method for dilute fluid-particle flows, Journal of Computational Physics 227 (2008) 2514–2539.
[21] J.K. Dukowicz, A particle-fluid numerical model for liquid sprays, Journal of Computational Physics 35 (2) (1980) 229–253.
[22] B. Engquist, O. Runborg, Multiphase computations in geometrical optics, Journal of Computational and Applied Mathematics 74 (1996) 175–192.
[23] R. Fan, D.L. Marchisio, R.O. Fox, Application of the direct quadrature method of moments to polydisperse gas–solid fluidized beds, Powder Technology 139 (2004) 7–20.
[24] P. Fede, O. Simonin, Numerical study of the subgrid fluid turbulence effects on the statistics of heavy colliding particles, Physics of Fluids 18 (2006) 045103.
[25] P. Février, O. Simonin, K.D. Squires, Partitioning of particle velocities in gas–solid turbulent flows into a continuous field and a spatially uncorrelated random distribution: theoretical formalism and numerical study, Journal of Fluid Mechanics 533 (2005) 1–46.
[26] R.O. Fox, Computational Models for Turbulent Reacting Flows, Cambridge University Press, Cambridge, 2003.
[27] R.O. Fox, Bivariate direct quadrature method of moments for coagulation and sintering of particle populations, Journal of Aerosol Science 37 (2006) 1562–1580.
[28] R.O. Fox, Optimal moment sets for multivariate quadrature-based moment methods, in preparation.
[29] R.O. Fox, F. Laurent, M. Massot, Numerical simulation of spray coalescence in an Eulerian framework: direct quadrature method of moments and multi-fluid method, Journal of Computational Physics 227 (2008) 3058–3088.
[30] U. Frisch, D. d'Humières, B. Hasslacher, P. Lallemand, Y. Pomeau, J.-P. River, Lattice gas hydrodynamics in two and three dimensions, Complex Systems 1 (4) (1987) 649–707.
[31] R. Gatignol, Théorie cinétique d'un gaz à répartition discrète de vitesses, Lecture Notes in Physics, vol. 36, Springer, Berlin, 1975.
[32] L. Gosse, Using K-branch entropy solutions for multivalued geometric optics computations, Journal of Computational Physics 180 (2002) 155–182.
[33] L. Gosse, S. Jin, X.T. Li, On two moment systems for computing multiphase semiclassical limits of the Schrödinger equation, Mathematical Models and Methods in Applied Science 13 (2003) 1689–1723.
[34] H. Grad, On the kinetic theory of rarefied gases, Communications on Pure and Applied Mathematics 2 (1949) 331–407.
[35] N.G. Hadjiconstantinou, The limits of Navier–Stokes theory and kinetic extensions for describing small-scale gaseous hydrodynamics, Physics of Fluids 18 (2006) 111301.
[36] N.G. Hadjiconstantinou, A.L. Garcia, M.Z. Bazant, G. He, Statistical error in particle simulations of hydrodynamic phenomena, Journal of Computational Physics 187 (2003) 274–297.
[37] J. He, O. Simonin, Non-equilibrium prediction of the particle-phase stress tensor in vertical pneumatic conveying, in: Fifth International Symposium on Gas–Solid Flows, ASME FED, vol. 166, 1993, pp. 253–263.
[38] J.T. Jenkins, S.B. Savage, A theory for the rapid flow of identical, smooth, nearly elastic spherical particles, Journal of Fluid Mechanics 130 (1983) 187–202.

[39] S. Jin, X.T. Li, Multi-phase computations of the semiclassical limit of the Schrödinger equation and related problems: Whitham vs Wigner, Physica D 182 (2003) 46–85.
[40] S. Jin, H. Liu, S. Osher, R. Tsai, Computing multi-valued physical observables for the semiclassical limit of the Schrödinger equation, Journal of Computational Physics 205 (2005) 222–241.
[41] P. Lallemand, L.-S. Luo, Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability, Physical Review E 68 (3) (2003) 036706.
[42] R. Leveque, Finite Volume Methods for Hyperbolic Problems, Cambridge University Press, Cambridge, 2002.
[43] C.D. Levermore, Moment closure hierarchies for kinetic theories, Journal of Statistical Physics 83 (1996) 1021–1065.
[44] X.T. Li, J.G. Wöhlbier, S. Jin, J.H. Booske, Eulerian method for computing multivalued solutions of the Euler–Poisson equations and applications to wave breaking in klystrons, Physical Review E 70 (2004) 016502.
[45] H. Liu, S. Osher, R. Tsai, Multi-valued solution and level set methods in computational high frequency wave propagation, Communications in Computational Physics 1 (5) (2006) 765–804.
[46] D.L. Marchisio, R.O. Fox, Solution of population balance equations using the direct quadrature method of moments, Journal of Aerosol Science 36 (2005) 43–73.
[47] D.L. Marchisio, J.T. Pikturna, R.O. Fox, R.D. Vigil, A.A. Barresi, Quadrature method of moments for population balances with nucleation, growth and aggregation, AIChE Journal 49 (2003) 1266–1276.
[48] D.L. Marchisio, R.D. Vigil, R.O. Fox, Quadrature method of moments for aggregation-breakage processes, Journal of Colloid and Interface Science 258 (2) (2003) 322–334.
[49] R. McGraw, Description of aerosol dynamics by the quadrature method of moments, Aerosol Science and Technology 27 (1997) 255–265.
[50] R. McGraw, Numerical advection of correlated tracers: preserving particle size/composition moment sequences during transport of aerosol mixtures, Journal of Physics: Conference Series 78 (2007) 012045.
[51] I. Nicodin, R. Gatignol, Unsteady half-space evaporation and condensation problems on the basis of the discrete kinetic theory, Physics of Fluids 18 (2006) 127105.
[52] Y. Ogata, H.-N. Im, T. Yabe, Numerical method for Boltzmann equation with Soroban-grid CIP method, Communications in Computational Physics 2 (4) (2007) 760–782.
[53] B. Perthame, Boltzmann type schemes for compressible Euler equations in one and two space dimensions, SIAM Journal of Numerical Analysis 29 (1) (1990) 1–19.
[54] D.I. Pullin, Direct simulation methods for compressible inviscid ideal gas-flows, Journal of Computational Physics 34 (1980) 53–66.
[55] O. Runborg, Some new results in multiphase geometrical optics, Mathematical Modelling and Numerical Analysis 34 (2000) 1203–1231.
[56] O. Runborg, Mathematical models and numerical methods for high frequency waves, Communications in Computational Physics 2 (5) (2007) 827–880.
[57] M. Sakiz, O. Simonin, Numerical experiments and modelling of non-equilibrium effects in dilute granular flows, Proceedings of the 21st International Symposium on Rarefied Gas Dynamics, Cépaduès-Éditions, Toulouse, France, 1998.
[58] L. Schiller, A. Nauman, A drag coefficient correlation, V.D.I. Zeitung 77 (1935) 318–320.
[59] A.H. Stroud, Approximate Calculation of Multiple Integrals, Prentice-Hall, Englewood Cliffs, 1971.
[60] S. Sundaram, L.R. Collins, Collision statistics in an isotropic particle-laden turbulent suspension. I. Direct numerical simulations, Journal of Fluid Mechanics 335 (1997) 75–109.
[61] T. Tanaka, Y. Tsuji, Numerical simulation of gas–solid two-phase flow in a vertical pipe: on the effect of inter-particle collision, in: Fourth International Symposium on Gas–Solid Flows, ASME FED, vol. 121, 1991, pp. 123–128.
[62] M. Torrilhon, H. Struchtrup, Regularized 13-moment equations: shock structure calculations and comparison to Burnett models, Journal of Fluid Mechanics 513 (2004) 171–198.
[63] M.W. Vance, K.D. Squires, O. Simonin, Properties of the particle velocity field in gas–solid turbulent channel flow, Physics of Fluids 18 (2006) 063302.
[64] P. Vedula, R.O. Fox, A quadrature-based method of moments for solution of the collisional Boltzmann equation, Journal of Statistical Physics, submitted for publication.
[65] L. Wang, D.L. Marchisio, R.D. Vigil, R.O. Fox, CFD simulation of aggregation and breakage processes in laminar Taylor–Couette flow, Journal of Colloid and Interface Science 282 (2005) 380–396.
[66] F.A. Williams, Spray combustion and atomization, Physics of Fluids 1 (1958) 541–545.
[67] J.G. Wöhlbier, S. Jin, S. Sengele, Eulerian calculations of wave breaking and multivalued solutions in a traveling wave tube, Physics of Plasmas 12 (2005) 023106.
[68] D.L. Wright, Numerical advection of moments of the particle size distribution in Eulerian models, Journal of Aerosol Science 38 (2007) 352–369.
[69] Y. Yamamoto, M. Potthoff, T. Tanaka, Y. Tsuji, Large-eddy simulation of turbulent gas-particle flow in a vertical channel: effect of considering inter-particle collisions, Journal of Fluid Mechanics 442 (2001) 303–334.
[70] C. Yoon, R. McGraw, Representation of generally mixed multivariate aerosols by the quadrature method of moments: I. Statistical foundation, Journal of Aerosol Science 35 (2004) 561–576.
[71] A. Zucca, D.L. Marchisio, A.A. Barresi, R.O. Fox, Implementation of the population balance equation in CFD codes for modelling soot formation in turbulent flames, Chemical Engineering Science 61 (2006) 87–95.