Low-noise Monte Carlo simulation of the variable hard sphere gas

Citation: Radtke, Gregg A., Nicolas G. Hadjiconstantinou, and Wolfgang Wagner. "Low-noise Monte Carlo simulation of the variable hard sphere gas." Physics of Fluids 23.3 (2011) : 030606.

As Published: http://dx.doi.org/10.1063/1.3558887

Publisher: American Institute of Physics

Persistent URL: http://hdl.handle.net/1721.1/64408

Version: Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

Low-noise Monte Carlo simulation of the variable hard sphere gas

Gregg A. Radtke,[1] Nicolas G. Hadjiconstantinou,[1] and Wolfgang Wagner[2]
[1]*Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*
[2]*Weierstrass Institute for Applied Analysis and Stochastics, Mohrenstrasse 39, D-10117 Berlin, Germany*

(Received 14 October 2010; accepted 4 November 2010; published online 18 March 2011)

We present an efficient particle simulation method for the Boltzmann transport equation based on the low-variance deviational simulation Monte Carlo approach to the variable-hard-sphere gas. The proposed method exhibits drastically reduced statistical uncertainty for low-signal problems compared to standard particle methods such as the direct simulation Monte Carlo method. We show that by enforcing mass conservation, accurate simulations can be performed in the transition regime requiring as few as ten particles per cell, enabling efficient simulation of multidimensional problems at arbitrarily small deviation from equilibrium. © *2011 American Institute of Physics*.
[doi:10.1063/1.3558887]

I. INTRODUCTION

Efficient simulation of low-signal small-scale gas flows, such as those occurring in microelectromechanical and nanoelectromechanical systems,[1–5] continues to represent a significant computational challenge.[6] This is because the direct simulation Monte Carlo (DSMC) method, the prevalent method for solving the Boltzmann equation,[7] is most efficient for simulating highly nonequilibrium flow conditions but suffers from high levels of statistical noise for smaller deviations from equilibrium.[8,9] Recently, stochastic particle methods that employ variance-reduction techniques[9] have demonstrated considerable efficiency improvements over the DSMC method. These approaches are based on the method of control variates[10] but can be described as falling into two broad subcategories: deviational methods,[9,11–14] where particles simulate the deviation from the equilibrium, and weight-based methods,[15,16] which exploit the correlation between an equilibrium and a nonequilibrium simulation to reduce statistical uncertainty.[16] A basic form of the correlated simulation approach, albeit without importance weights, was originally proposed in the context of the Brownian dynamics simulations.[17] Unfortunately, it was observed[18] that the correlation between the two simulations cannot be maintained indefinitely, resulting in loss of variance reduction; in the Boltzmann simulations this manifests itself in the form of diverging weights in weight-based simulations and diverging number of particles in deviational simulations. This divergence can be mitigated by numerical procedures, such as particle cancellation routines (deviational simulations) or techniques for reconstructing the distribution function (weight-based methods), albeit at the cost of numerical error and computational cost. Here, we mention that the method outlined in Ref. 16 uses kernel density estimation to ensure weight stability and results in an efficient variance-reduction procedure operating in parallel with an essentially unmodified DSMC simulation.

Resolution of the above limitations came with the development of low-variance direct simulation Monte Carlo (LVDSMC),[12,13] a deviational method which uses a form of the hard-sphere collision operator, originally obtained by Hilbert,[19] in which the angular integration within the Boltzmann collision integral is performed analytically. This has the effect of providing particle "precancellation" and leads to a stable simulation method with no numerical intervention (and associated error). The developers of the original LVDSMC method[12,13] were made aware of this special form of the collision operator via Cercignani's various expositions,[6,19,20] which provide alternative derivations as well as a historical perspective[19] on related work by Hilbert, Enskog, Carleman, and Grad.

The LVDSMC methodology has been extended to treat the Bhatnagar–Gross–Krook collision model.[21,22] These studies have also served to highlight the differences between using a global or a local (spatially variable) equilibrium distribution as a control; their findings are briefly discussed in Sec. IV. Recently, the LVDSMC methodology was put on a more precise theoretical footing.[23] In the same publication, a collision algorithm with no inherent time step error that can also treat the variable-hard-sphere (VHS) collision model[24]—which is more realistic for engineering flows—was also proposed.

In this paper we present an LVDSMC algorithm which combines the recently introduced VHS collision algorithm[23] with a highly efficient advection routine[21] within a formulation that enforces mass conservation. The latter significantly reduces the number of particles required for accurate simulations compared to previous implementations—as we show below accurate results can be obtained with as low as approximately ten particles per cell, as in DSMC—resulting in a significantly more efficient and versatile method. Since the primary application domain of this method is low-signal flows, here we treat the linearized version of the collision operator. Also, in the interest of simplicity, the present algorithm simulates the deviation from a global (constant) equilibrium distribution.

The paper is organized as follows: the overall simulation method is described in Sec. II with the collision, advection, property evaluation, and time step procedures discussed in

separate sections. A selected number of validation and demonstration cases in two-spatial dimensions are presented in Sec. III; these demonstrate that the proposed algorithm simulates the Boltzmann transport with considerable computational efficiency savings for problems with small departures from equilibrium. Finally, in Sec. IV, a summary, as well as a discussion of open issues and future research directions, is presented.

II. SIMULATION METHOD

The LVDSMC method is derived directly from the Boltzmann transport equation,

$$\frac{\partial f(\boldsymbol{c})}{\partial t} + \boldsymbol{c} \cdot \frac{\partial f(\boldsymbol{c})}{\partial \boldsymbol{x}} + \boldsymbol{a} \cdot \frac{\partial f(\boldsymbol{c})}{\partial \boldsymbol{c}} = \mathcal{Q}[f,f](\boldsymbol{c}), \tag{1}$$

which simulates. In the above, $f=f(\boldsymbol{c})$ is the velocity distribution function, t is time, $\boldsymbol{c}$ is the particle velocity, $\boldsymbol{x}$ is the spatial coordinate, and $\boldsymbol{a}$ is the body force per unit mass. In the flow regimes of interest for this work, the gravitational body force is negligible, and thus in what follows we assume $\boldsymbol{a}=0$; relaxing this assumption requires small modifications to the present algorithm.

The collision operator for the VHS model is given by

$$\mathcal{Q}[f,f](\boldsymbol{c}) = C_\beta \int_{\mathcal{S}^2} \mathrm{d}^2\boldsymbol{\Omega} \int_{\mathcal{R}^3} \mathrm{d}^3\boldsymbol{c}_* \|\boldsymbol{c}-\boldsymbol{c}_*\|^\beta \cdot [f(\boldsymbol{c}')f(\boldsymbol{c}_*') - f(\boldsymbol{c})f(\boldsymbol{c}_*)], \tag{2}$$

where primes denote postcollision velocities $\{\boldsymbol{c}',\boldsymbol{c}_*'\} = \frac{1}{2}(\boldsymbol{c}+\boldsymbol{c}_* \pm \|\boldsymbol{c}-\boldsymbol{c}_*\|\boldsymbol{\Omega})$ and the solid angle $\boldsymbol{\Omega}$ is integrated over the unit sphere $\mathcal{S}^2$. The relative velocity exponent β is related to the temperature coefficient of viscosity ω via $\beta=2(1-\omega)$; the constant prefactor is given by $C_\beta=\frac{1}{4m}d_{\mathrm{ref}}^2 c_{\mathrm{r,ref}}^{1-\beta}$, where m is the molecular mass, d_{ref} is the reference molecular diameter, and $c_{\mathrm{r,ref}}=4\sqrt{RT_{\mathrm{ref}}/\pi}$ is the mean relative molecular speed at reference temperature T_{ref}.

In the deviational approach, the velocity distribution is split into an equilibrium, f^0, and a deviational part, f^{d}. The equilibrium part is taken to be a fixed Maxwell–Boltzmann distribution,

$$f^0(\boldsymbol{c}) = \frac{\rho_0}{\pi^{3/2}c_0^3}\exp\left(-\frac{\|\boldsymbol{c}-\boldsymbol{u}_0\|^2}{c_0^2}\right), \tag{3}$$

with density ρ_0, mean velocity $\boldsymbol{u}_0=(u_{0,x},u_{0,y},u_{0,z})$, most probable molecular velocity $c_0=\sqrt{2RT_0}$, and temperature T_0. The deviational distribution is formally represented by signed particles via $f^{\mathrm{d}}(\boldsymbol{c})=mW\Sigma_{i=1}^{N} s_i\delta^3(\boldsymbol{x}-\boldsymbol{x}_i)\delta^3(\boldsymbol{c}-\boldsymbol{c}_i)$, where each particle is characterized by a sign $s_i\in\pm 1$ in addition to a position $\boldsymbol{x}_i$ and velocity $\boldsymbol{c}_i$. Here, W is a constant which relates the number of physical molecules to the number of deviational particles in the simulation. This quantity (W) plays the role of N_{eff} in DSMC simulations by allowing a numerical particle to represent a number of physical particles; however, the relationship between W and the ratio of physical to numerical particles is less direct in the deviational approach (this point will be elucidated with a numerical example in Sec. III).

The simulation domain is discretized into $N_{\Delta V}$ disjoint spatial cells, each with spatial volume ΔV_j, $j\in\{1,2,\ldots,N_{\Delta V}\}$, where the total volume is given by $V=\Sigma_{j=1}^{N_{\Delta V}}\Delta V_j$. The set of particles contained within cell j at the instantaneous state of the simulation is denoted by $\mathcal{N}_j$, where $\cup_{j=1}^{N_{\Delta V}}\mathcal{N}_j=\{1,2,\ldots,N\}$. Likewise, the boundary is discretized into $N_{\Delta A}$ surface elements, each with area ΔA_j, $j\in\{1,2,\ldots,N_{\Delta A}\}$.

Similar to the DSMC method, evolution under the Boltzmann dynamics is calculated by splitting into collision and advection steps. Using a formulation by Cercignani and Daneri,[25] streamwise pressure and temperature gradients are included using forcing terms that resemble effective "body forces," implemented here as part of the splitting algorithm. These steps are described in detail in the following sections.

A. Collision step

The collision algorithm presented here was proposed previously[23] and was presently extended to feature mass conservation. By simulating the collision process as a sequence of Markov particle creation and deletion events, this collision algorithm has no intrinsic time step error, in contrast to previous LVDSMC methods.[12,13,21,22]

In this method,[23] collision events are processed in order to simulate the Boltzmann equation [Eq. (1)] in the absence of advection,

$$\frac{\partial f(\boldsymbol{c})}{\partial t} = \mathcal{Q}[f,f](\boldsymbol{c}). \tag{4}$$

By substituting $f=f^0+f^{\mathrm{d}}$ into Eq. (2), the collision operator is represented by linear $\mathcal{L}[f^{\mathrm{d}}]$ and nonlinear $\mathcal{Q}[f^{\mathrm{d}},f^{\mathrm{d}}]$ terms,

$$\mathcal{Q}[f,f](\boldsymbol{c}) = \mathcal{L}[f^{\mathrm{d}}](\boldsymbol{c}) + \mathcal{Q}[f^{\mathrm{d}},f^{\mathrm{d}}](\boldsymbol{c}). \tag{5}$$

For this paper, we focus on the linear part of the collision operator; this is a reasonable approximation since we are interested in low-signal problems. The more general nonlinear approach has been published in a preliminary study.[26]

The key in efficiently simulating collisions while maintaining stability in the LVDSMC method lies in exploiting a special representation,[6,12,13,19,20,23]

$$\mathcal{L}[f^{\mathrm{d}}](\boldsymbol{c}) = \int_{\mathcal{R}^3} \mathrm{d}^3\boldsymbol{c}_* [2K^{(1)} - K^{(2)}](\boldsymbol{c},\boldsymbol{c}_*) f^{\mathrm{d}}(\boldsymbol{c}_*) - \nu(\boldsymbol{c}) f^{\mathrm{d}}(\boldsymbol{c}), \tag{6}$$

$$K^{(1)}(\boldsymbol{c},\boldsymbol{c}_*) = \frac{4C_\beta}{\|\boldsymbol{c}-\boldsymbol{c}_*\|}\int_{\Gamma_\perp(\boldsymbol{c}-\boldsymbol{c}_*)} \mathrm{d}^3\boldsymbol{\zeta} \frac{f^0(\boldsymbol{c}+\boldsymbol{\zeta})}{\|\boldsymbol{c}-\boldsymbol{c}_*-\boldsymbol{\zeta}\|^{1-\beta}}, \tag{7}$$

$$K^{(2)}(\boldsymbol{c},\boldsymbol{c}_*) = 4\pi C_\beta \|\boldsymbol{c}-\boldsymbol{c}_*\|^\beta f^0(\boldsymbol{c}), \tag{8}$$

$$\nu(\boldsymbol{c}) = 4\pi C_\beta \int_{\mathcal{R}^3} \mathrm{d}^3\boldsymbol{c}_* \|\boldsymbol{c}-\boldsymbol{c}_*\|^\beta f^0(\boldsymbol{c}_*), \tag{9}$$

where $\Gamma_\perp(\boldsymbol{c})$ is the plane perpendicular to $\boldsymbol{c}$ passing through the origin. This structure allows for efficiently sampling the $[2K^{(1)}-K^{(2)}]$ term as a single distribution, which produces fewer extraneous particles; moreover, particle deletion

through the term $-\nu f^{\mathrm{d}}$ lends stability to the method by countering an unbounded increase in the number of particles in the simulation. The origin of representation (6) can be explained as follows: the convolution involving $K^{(1)}(\boldsymbol{c},\boldsymbol{c}_*)$ follows from angular integration of the linearized form of the gain term of the collision integral; the other two terms in Eq. (6) originate from the linearized loss term: the convolution involving $K^{(2)}(\boldsymbol{c},\boldsymbol{c}_*)$ is obtained from integration of $f^0(\boldsymbol{c})f^{\mathrm{d}}(\boldsymbol{c}_*)$, while the term $\nu(\boldsymbol{c})f^{\mathrm{d}}(\boldsymbol{c})$ follows from integration of $f^0(\boldsymbol{c}_*)f^{\mathrm{d}}(\boldsymbol{c})$.

The collision rate and the kernel functions are related as follows:

$$\nu(\boldsymbol{c}_*) = \int_{\mathcal{R}^3} \mathrm{d}^3\boldsymbol{c} K^{(1)}(\boldsymbol{c},\boldsymbol{c}_*) = \int_{\mathcal{R}^3} \mathrm{d}^3\boldsymbol{c} K^{(2)}(\boldsymbol{c},\boldsymbol{c}_*). \tag{10}$$

Using the inequality,

$$\left(\frac{\|\boldsymbol{c}-\boldsymbol{c}_*\|}{c_0}\right)^{\beta} \le \left[\beta\frac{\|\boldsymbol{c}-\boldsymbol{c}_*\|}{c_0} + (1-\beta)\right], \quad \forall\ \boldsymbol{c},\boldsymbol{c}_* \in \mathcal{R}^3, \tag{11}$$

a tight bound on the collision rate ν can be formed,

$$\nu(\boldsymbol{c}) \le \nu_{\max}(\boldsymbol{c}) = 4\pi C_\beta \rho_0 c_0^\beta [\beta\psi(\xi) + (1-\beta)], \quad \forall\ \ \boldsymbol{c} \in \mathcal{R}^3, \tag{12}$$

where $\boldsymbol{\xi}=(\boldsymbol{c}-\boldsymbol{u}_0)/c_0$, $\xi=\|\boldsymbol{\xi}\|$, and $\psi(\xi)$ is a pure numerical function given by

$$\psi(\xi) = \frac{e^{-\xi^2}}{\sqrt{\pi}} + \left(\xi + \frac{1}{2\xi}\right)\mathrm{erf}(\xi). \tag{13}$$

Here, equality for Eqs. (11) and (12) is recovered for both the hard-sphere ($\beta=1$) and the Maxwell-molecule ($\beta=0$) limits.

Using the common bound $\nu_{\max}$ [Eqs. (10) and (12)] for all terms appearing in collision operator (6), the simulation is performed using a common (stochastic) time step δt to process both deletion and particle generation events. These stochastic time steps can be interpreted as waiting times between "arrivals" in a Poisson process, in which the state of the gas is transformed by adding or deleting a particle or remains unchanged. In this context, the time steps δt are sampled from an appropriate exponential distribution: $P(\delta t)=\Lambda e^{-\Lambda\delta t}$, $\delta t \in (0,\infty)$, which has parameter

$$\Lambda = 16\pi C_\beta \rho_0 c_0^\beta \left[\beta\sum_{k=1}^{N}\psi(\xi_k) + (1-\beta)N\right]. \tag{14}$$

The proper number of arrivals is simulated when the total collision time (total sum of all stochastic time steps for all collision steps) exceeds the simulation time $t+\Delta t_{\mathrm{col}}$, at which point the collision routine passes control over to the advection routine (cf. Sec. II E). For each time step, a trial deletion step is performed with probability 1/4 and a trial particle generation step is performed with the remaining probability. Each routine is summarized and discussed below.

1. Particle generation routine

The particle generation step is the most complex part of the VHS collision algorithm; here, we summarize the basic steps of the algorithm, while the mathematical derivation of these procedures is available elsewhere.[23] In the algorithm below, the notation $\pi_{\boldsymbol{c}-\boldsymbol{c}_i}(\boldsymbol{c}_*)$ refers to the vector projection of $\boldsymbol{c}_*$ onto the plane $\Gamma_\perp(\boldsymbol{c}-\boldsymbol{c}_i)$, while $I^0_{\Gamma_\perp}(\boldsymbol{c},\boldsymbol{c}_k)$ is given by

$$\begin{aligned} I^0_{\Gamma_\perp}(\boldsymbol{c},\boldsymbol{c}_k) &= \int_{\Gamma_\perp(\boldsymbol{c}-\boldsymbol{c}_k)} \mathrm{d}^3\boldsymbol{c}_* f^0(\boldsymbol{c}+\boldsymbol{c}_*) \\ &= \frac{\rho_0}{\sqrt{\pi}c_0}\exp\left(-\frac{|(\boldsymbol{c}-\boldsymbol{u}_0)\cdot(\boldsymbol{c}-\boldsymbol{c}_k)|^2}{c_0^2\|\boldsymbol{c}-\boldsymbol{c}_k\|^2}\right). \end{aligned} \tag{15}$$

We also introduce two additional notations: an acceptance probability $P^0_\beta(\boldsymbol{c},\boldsymbol{c}_*)$ and particle sign for the accepted particles $s^0_\beta(\boldsymbol{c},\boldsymbol{c}_*)$ given by

$$P^0_\beta(\boldsymbol{c},\boldsymbol{c}_*) = \frac{\left|\sum_{k\in\mathcal{N}_j} s_k\left[\frac{2I^0_{\Gamma_\perp}(\boldsymbol{c},\boldsymbol{c}_k)/c_0^\beta}{\|\boldsymbol{c}-\boldsymbol{c}_k\|\,\|\boldsymbol{c}-\boldsymbol{c}_k-\pi_{\boldsymbol{c}-\boldsymbol{c}_k}(\boldsymbol{c}_*)\|^{1-\beta}} - \pi\left(\frac{\|\boldsymbol{c}-\boldsymbol{c}_k\|}{c_0}\right)^\beta f^0(\boldsymbol{c})\right]\right|}{\sum_{k\in\mathcal{N}_j}\left\{\frac{2I^0_{\Gamma_\perp}(\boldsymbol{c},\boldsymbol{c}_k)}{\|\boldsymbol{c}-\boldsymbol{c}_k\|}\left[\frac{\beta}{c_0}+\frac{1-\beta}{\|\boldsymbol{c}-\boldsymbol{c}_k-\pi_{\boldsymbol{c}-\boldsymbol{c}_k}(\boldsymbol{c}_*)\|}\right] + \pi\left[\beta\frac{\|\boldsymbol{c}-\boldsymbol{c}_k\|}{c_0}+(1-\beta)\right]f^0(\boldsymbol{c})\right\}} \tag{16}$$

and

$$s^0_\beta(\boldsymbol{c},\boldsymbol{c}_*) = \mathrm{sgn}\left(\sum_{k\in\mathcal{N}_j} s_k\left[\frac{2I^0_{\Gamma_\perp}(\boldsymbol{c},\boldsymbol{c}_k)}{\|\boldsymbol{c}-\boldsymbol{c}_k\|\,\|\boldsymbol{c}-\boldsymbol{c}_k-\pi_{\boldsymbol{c}-\boldsymbol{c}_k}(\boldsymbol{c}_*)\|^{1-\beta}} - \pi\|\boldsymbol{c}-\boldsymbol{c}_k\|^\beta f^0(\boldsymbol{c})\right]\right). \tag{17}$$

Algorithm 1. Particle generation routine for VHS collisions.

1. Choose a particle index i according to the probabilities,
$$\frac{\beta\psi(\xi_i)+(1-\beta)}{\beta\Sigma_{k=1}^{N}\psi(\xi_k)+(1-\beta)N}. \tag{18}$$
Determine the cell index j to which particle i belongs.
2. Based on index i, generate a velocity $\boldsymbol{c}$ from distribution,
$$\frac{4\pi C_\beta c_0^\beta}{\nu_{\max}(\boldsymbol{c}_i)}\left[\beta\frac{\|\boldsymbol{c}-\boldsymbol{c}_i\|}{c_0}+(1-\beta)\right]f^0(\boldsymbol{c}). \tag{19}$$
3. With probability 2/3, perform step 4. Otherwise, perform step 5.
4. Continue to step 4.1.

4.1. Replace $\boldsymbol{c}$ with a postcollision velocity $\boldsymbol{c}\rightarrow\boldsymbol{c}'$ via $\boldsymbol{c}'=\frac{1}{2}(\boldsymbol{c}+\boldsymbol{c}_i+\|\boldsymbol{c}-\boldsymbol{c}_i\|\boldsymbol{\Omega})$, where $\boldsymbol{\Omega}$ is sampled from $\mathcal{S}^2$ uniformly.
4.2. Produce a sample $\boldsymbol{c}_*$ from distribution $f^0(\boldsymbol{c}+\boldsymbol{c}_*)/\rho_0$.
4.3. With probability,
$$\frac{\beta+(1-\beta)c_0\|\boldsymbol{c}-\boldsymbol{c}_i-\pi_{\boldsymbol{c}-\boldsymbol{c}_i}(\boldsymbol{c}_*)\|^{-1}}{\beta+(1-\beta)c_0\|\boldsymbol{c}-\boldsymbol{c}_i\|^{-1}}, \tag{20}$$
go to step 6. Otherwise, return to step 4.2.

5. Produce a sample $\boldsymbol{c}_*$ from $f^0(\boldsymbol{c}+\boldsymbol{c}_*)/\rho_0$.
6. Acceptance/rejection step: with probability $P_\beta^0(\boldsymbol{c},\boldsymbol{c}_*)$ (16) accept generated particle by adding it to the simulation with velocity $\boldsymbol{c}$, position $\boldsymbol{x}$ sampled uniformly from cell j, and sign $s_\beta^0(\boldsymbol{c},\boldsymbol{c}_*)$ (17). Otherwise, the procedure finishes without generating a particle.

Here, acceptance probability (16) entails summing over all particles in the cell. In the original,[23] the possibility of performing this summation over a fraction of the particles in the cell is discussed. Because in our current implementation a small number of particles are required, we have limited our approach to performing the summation over the entire cell. Some discussion of the appropriate number of particles to average over can be found in Ref. 23, but this issue merits future investigation.

2. Particle deletion routine

In the particle deletion routine, the first two steps of the particle generation routine are performed by choosing a particle (i) from index distribution (18) and sampling a velocity $\boldsymbol{c}$ from distribution (19). The particle is deleted with probability

$$\frac{(\|\boldsymbol{c}-\boldsymbol{c}_i\|/c_0)^\beta}{\beta\|\boldsymbol{c}-\boldsymbol{c}_i\|/c_0+(1-\beta)}. \tag{21}$$

Otherwise, the simulation remains unchanged.

3. Mass conservation

The LVDSMC collision algorithm conserves mass, momentum, and energy only on average; this is a weaker sense of conservation compared to DSMC, which conserves these quantities for individual collision events. Here, we are able to achieve conservation of mass by appropriate stochastic steps that correct the total mass residual. This is performed at the end of each collision step by resampling particles from the set $\mathcal{G}$ of particles which were generated during the previous collision step but were not subsequently deleted. The mass conservation algorithm makes use of the stochastic particle creation routine (above) from the collision algorithm.

The mass residual ΔS is defined as the total sign of all generated particles minus the total sign of all deleted particles for all previous collision steps. This is continuously tracked: first by initializing $\Delta S=0$ at the start of the simulation (unless ΔS is available from a restart file) and by updating $\Delta S=\Delta S+s_{\text{gen}}$ for each generated particle with sign s_{gen} and $\Delta S=\Delta S-s_{\text{del}}$ for each deleted particle with sign s_{del}. Following each collision step, ΔS is reduced to its minimum possible absolute value (typically to zero). In the event that the residual is not eliminated, it is carried over to be addressed during the next time step.

When ΔS is an odd number, it cannot be reduced to zero by resampling processes; thus, the initial step in the mass conservation process involves correcting the parity of the mass residual. This step consists of repeating the particle generation routine (above) until a single particle is accepted, with probability 1/2, or deleting a random particle (uniformly) from $\mathcal{G}$ (by removing it from the simulation), with probability 1/2. This step can only be performed if the number of particles $N_{\mathcal{G}}$ in $\mathcal{G}$ is nonzero; otherwise ΔS cannot be changed in the current time step, and the parity correction step (as well as the resampling step) will be skipped entirely.

Following the parity correction step, resampling events are performed until the optimal mass residual ΔS_{opt} is attained. Here, the optimal mass residual is defined as the minimum absolute mass residual obtainable by resampling from set $\mathcal{G}$, initially containing $N_{\mathcal{G}}^+$ positive and $N_{\mathcal{G}}^-$ negative particles,

$$\Delta S_{\text{opt}}=\begin{cases}0 & \text{if } N_{\mathcal{G}}^{\mp}\geq\frac{1}{2}|\Delta S| \text{ and } \Delta S\lessgtr 0\\ \Delta S\pm 2N_{\mathcal{G}}^{\mp} & \text{if } N_{\mathcal{G}}^{\mp}<\frac{1}{2}|\Delta S| \text{ and } \Delta S\lessgtr 0.\end{cases} \tag{22}$$

For the trivial case, $\Delta S=\Delta S_{\text{opt}}=0$ and no resampling is needed.

Here, we introduce the partition $\mathcal{G}=\mathcal{G}^+\cup\mathcal{G}^-$, where $\mathcal{G}^\pm$ are the subsets of $\mathcal{G}$ with positive and negative signs, respectively. The resampling procedure consists of performing the following two steps in random order: (i) delete a random particle (uniformly) from $\mathcal{G}^{\text{sgn}(\Delta S)}$ and (ii) generate a particle with sign $-\text{sgn}(\Delta S)$. In step (ii), we use the particle generation step used in the collision routine, repeating the routine automatically rejecting all particles with sign $\text{sgn}(\Delta S)$ until a single particle is generated with the correct sign, which is added to the simulation. This procedure is repeated until $\Delta S=\Delta S_{\text{opt}}$.

B. Advection step

The advection procedure is based on a previous method,[21] with a few key differences in the present treatment. First, in this paper, we are simulating deviation from a

fixed equilibrium distribution f^0 rather than a spatially variable equilibrium distribution. Although the former method has a clear efficiency advantage for one-dimensional flow in the Navier–Stokes limit ($Kn \to 0$), it becomes significantly more expensive as the number of dimensions increases as it requires particle generation at all cell interfaces. The second key difference is the inclusion of mass conservation to complement the mass-conservative collision routine. Finally, while the previous method[21] is strictly valid only for small perturbations from equilibrium, here we generalize the method for all regimes; a version without mass conservation was presented in a previous paper.[26]

The advection step simulates the left-hand side of the Boltzmann equation [Eq. (1)], i.e.,

$$\frac{\partial f(\boldsymbol{c})}{\partial t} + \boldsymbol{c} \cdot \frac{\partial f(\boldsymbol{c})}{\partial \boldsymbol{x}} = 0. \tag{23}$$

By introducing $f = f^0 + f^{\mathrm{d}}$ into Eq. (23), the following deviational advection equation is obtained:

$$\frac{\partial f(\boldsymbol{c})}{\partial t} + \boldsymbol{c} \cdot \frac{\partial f(\boldsymbol{c})}{\partial \boldsymbol{x}} = \frac{\partial f^{\mathrm{d}}(\boldsymbol{c})}{\partial t} + \boldsymbol{c} \cdot \frac{\partial f^{\mathrm{d}}(\boldsymbol{c})}{\partial \boldsymbol{x}} = 0, \tag{24}$$

which shows that deviational particles advect identically to physical particles. Thus, in the absence of boundary interactions, particles are advected according to usual DSMC rule: $\{\boldsymbol{x}_k(t+\Delta t_{\mathrm{adv}}) = \boldsymbol{x}_k(t) + \boldsymbol{c}_k(t)\Delta t_{\mathrm{adv}}\}_{k=1}^{N}$ for the advective time step Δt_{adv}.

For boundary interactions, the standard DSMC rules are extended. When the particle strikes a boundary, it is reflected according to the standard DSMC rules (e.g., by redrawing the velocity from the appropriate fluxal boundary distribution). However, when a pair of particles of opposite signs strike the same boundary element and diffusively reflect in their first wall collision during an advective time step, they can both be removed from the simulation. This step is necessary to stabilize simulations in the collisionless ($Kn \to \infty$) limit by preventing an unbounded increase in the number of particles.

In addition to reflecting escaping particles back into the simulation domain, additional particles must be generated at the boundary to account for the difference in fluxes between the equilibrium and boundary distributions.[12,13,21–23] The boundary generation procedure for Maxwell's accommodation model has been derived previously for the special case of the no-flux boundary condition with $\boldsymbol{u}_{\mathrm{B},j} \cdot \boldsymbol{n}_j = 0$,[21,22] where j indices the boundary surface element with (inward) surface normal $\boldsymbol{n}_j$, velocity $\boldsymbol{u}_{\mathrm{B},j}$, temperature $T_{\mathrm{B},j}$, and accommodation coefficient α_j. In this case, the particle generation term is

$$\frac{\partial f(\boldsymbol{c})}{\partial t} \Delta A_j = \Delta A_j \alpha_j \boldsymbol{c} \cdot \boldsymbol{n}_j [\rho_{\mathrm{B},j} \phi_j^{\mathrm{B}}(\boldsymbol{c}) - f^0(\boldsymbol{c})], \tag{25}$$

where

$$\phi_j^{\mathrm{B}}(\boldsymbol{c}) = \frac{1}{\pi^{3/2} c_{\mathrm{B},j}^3} \exp\left(- \frac{\|\boldsymbol{c} - \boldsymbol{u}_{\mathrm{B},j}\|^2}{c_{\mathrm{B},j}^2} \right), \quad c_{\mathrm{B},j} = \sqrt{2RT_{\mathrm{B},j}}, \tag{26}$$

and the "boundary density" ($\rho_{\mathrm{B},j}$) is evaluated via mass conservation at the boundary

$$\rho_{\mathrm{B},j} \int_{\boldsymbol{c}\cdot\boldsymbol{n}_j > 0} \mathrm{d}^3\boldsymbol{c} (\boldsymbol{c} \cdot \boldsymbol{n}_j) \phi_j^{\mathrm{B}}(\boldsymbol{c}) = \int_{\boldsymbol{c}\cdot\boldsymbol{n}_j < 0} \mathrm{d}^3\boldsymbol{c} (-\boldsymbol{c} \cdot \boldsymbol{n}_j) f^0(\boldsymbol{c}), \tag{27}$$

which can be analytically solved for $\rho_{\mathrm{B},j}$. For $\boldsymbol{u}_0 \cdot \boldsymbol{n}_j = 0$, distribution (25) is conveniently sampled in terms of a dimensionless velocity $\boldsymbol{\xi} = (\boldsymbol{c} - \boldsymbol{u}_0)/c_0$,

$$\frac{\partial f(\boldsymbol{c})}{\partial t} \Delta A_j \mathrm{d}^3\boldsymbol{c} = \Delta A_j \alpha_j \rho_0 c_0 F_j^{\mathrm{B}}(\boldsymbol{c}) \mathrm{d}^3\boldsymbol{\xi},$$
$$F_j^{\mathrm{B}}(\boldsymbol{c}) = \frac{c_0^2}{\rho_0} \boldsymbol{c} \cdot \boldsymbol{n}_j [\rho_{\mathrm{B},j} \phi_j^{\mathrm{B}}(\boldsymbol{c}) - f^0(\boldsymbol{c})]. \tag{28}$$

Without loss of generality, we will assume that $\boldsymbol{n}_j$ is in the $+x$ direction; the more general case can be handled by the appropriate vector transformations. Generation term (28) is efficiently sampled by using the ratio-of-uniforms method,[27] as implemented in a previous publication.[21] The ratio-of-uniforms method produces samples from a transformed distribution $H(\boldsymbol{\eta})$, which is related to the original distribution via $|F_j^{\mathrm{B}}| = H^{5/2}$ and $\boldsymbol{\xi} = \boldsymbol{\eta}/\sqrt{H}$. An important advantage of this formulation is that the transformed variables are all bounded quantities,

$$0 \le H \le a_j^{\mathrm{B}}, \tag{29}$$

$$0 \le \eta_x \le b_{j,x}^{\mathrm{B}}, \tag{30}$$

$$-b_{j,y}^{\mathrm{B}} \le \eta_y \le b_{j,y}^{\mathrm{B}}, \tag{31}$$

$$-b_{j,z}^{\mathrm{B}} \le \eta_z \le b_{j,z}^{\mathrm{B}}. \tag{32}$$

For small perturbations from equilibrium, tight bounds can be obtained via a Taylor expansion of F_j^{B} about f^0, as was done in a previous paper.[21] These are listed below as functions of the boundary properties,

$$\begin{bmatrix} (a_j^{\mathrm{B},0})^{5/2} \\ (b_{j,x}^{\mathrm{B},0})^5 \\ (b_{j,y}^{\mathrm{B},0})^5 \\ (b_{j,z}^{\mathrm{B},0})^5 \end{bmatrix} = \mathbf{M}_{\mathrm{B}} \cdot \begin{bmatrix} \left| \frac{\rho_{\mathrm{B},j} - \rho_0}{\rho_0} - 3\frac{c_{\mathrm{B},j} - c_0}{c_0} \right| \\ 2\frac{|u_{\mathrm{B},j,x} - u_{0,x}|}{c_0} \\ 2\frac{|u_{\mathrm{B},j,y} - u_{0,y}|}{c_0} \\ 2\frac{|u_{\mathrm{B},j,z} - u_{0,z}|}{c_0} \\ 2\frac{|c_{\mathrm{B},j} - c_0|}{c_0} \end{bmatrix}, \tag{33}$$

where $\mathbf{M}_{\mathrm{B}}$ is a constant matrix given by

$$\mathbf{M}_{\mathrm{B}} = \frac{1}{\pi^{3/2}} \begin{bmatrix} 1/\sqrt{2e} & 1/e & 1/(2e) & 1/(2e) & [3/(2e)]^{3/2} \\ (3/e)^3 & [7/(2e)]^{7/2} & 27e^{-7/2}/\sqrt{2} & 27e^{-7/2}/\sqrt{2} & (4/e)^4 \\ 5^{5/2}/(2e)^3 & (5/2)^{5/2}e^{-7/2} & 27e^{-7/2}/\sqrt{2} & 5^{5/2}/(2e)^{7/2} & 20^{5/2}/(27e^4) \\ 5^{5/2}/(2e)^3 & (5/2)^{5/2}e^{-7/2} & 5^{5/2}/(2e)^{7/2} & 27e^{-7/2}/\sqrt{2} & 20^{5/2}/(27e^4) \end{bmatrix}. \tag{34}$$

These bounds are extended to more general conditions by introducing numerical factors (Y), which are dynamically updated during the simulation,

$$a_j^{\mathrm{B}} = Y_a^{\mathrm{B}} a_j^{\mathrm{B},0}, \tag{35}$$

$$b_{j,x}^{\mathrm{B}} = Y_{b,x}^{\mathrm{B}} b_{j,x}^{\mathrm{B},0}, \tag{36}$$

$$b_{j,y}^{\mathrm{B}} = Y_{b,y}^{\mathrm{B}} b_{j,y}^{\mathrm{B},0}, \tag{37}$$

$$b_{j,z}^{\mathrm{B}} = Y_{b,z}^{\mathrm{B}} b_{j,z}^{\mathrm{B},0}. \tag{38}$$

The number of trial samples (in $H, \boldsymbol{\eta}$ space) is calculated based on these bounds and the Jacobian of the transformation (=5/2), yielding

$$N_{\mathrm{B},j}^{\mathrm{trial}} = \frac{5}{2} \frac{A_j \alpha_j \rho_0 c_0 \Delta t_{\mathrm{adv}}}{mW} a_j^{\mathrm{B}} b_{j,x}^{\mathrm{B}} (2b_{j,y}^{\mathrm{B}})(2b_{j,z}^{\mathrm{B}}). \tag{39}$$

Here, Δt_{adv} is the advective time step. For each trial step, a sample $(H, \boldsymbol{\eta})$ is generated (uniformly) utilizing bounds (35)–(38). Using $\boldsymbol{c} = c_0 \boldsymbol{\eta}/\sqrt{H} + \boldsymbol{u}_0$, F_j^{B} is evaluated from Eq. (28), and the trial particle is accepted if $H < |F_j^{\mathrm{B}}|^{2/5}$. Accepted particles are advected a random fraction of the advective time step away (performing standard DSMC procedures for any boundary interactions) from a uniformly distributed random position on the boundary surface element and added to the simulation with sign $\mathrm{sgn}(F_j^{\mathrm{B}})$.

The ratio-of-uniforms sampling bounds are dynamically updated using the following procedure. At the start of the advection step, bounds (35)–(38) are fixed based on current values. During the advection step, when accepted particles are added with H or $\boldsymbol{\eta}$ values very close to one of the sampling bounds, the numerical factor is increased for the next advection step. For example, when a particle is accepted with $H > (1-\chi) a_j^{\mathrm{B}}$, the numerical factor is updated to

$$Y_a^{\mathrm{B}} = \max\left\{ \frac{H}{(1-\chi) a_j^{\mathrm{B},0}}, Y_a^{\mathrm{B}} \right\}. \tag{40}$$

Here, the sampling margin χ is a small positive numerical parameter which controls the responsiveness of the dynamic update; typically, χ is chosen to be a few percent. The updated Y_a^{B} value does not take effect until subsequent time steps, when sampling bounds are reevaluated via Eq. (35). A similar procedure is followed for dynamically updating the remaining bounds. For typical simulations, the bounds are well-characterized by their approximate values (33) and (34), and the numerical factors represent only small corrections.

1. Mass conservation

Mass conservation requires only a simple modification to the advection step, accomplished by applying a stratified sampling approach.[10] The particle generation routine is split into two separate processes. First, only half $N_{\mathrm{B},j}^{\mathrm{trial}}/2$ of the trial generation steps are performed, keeping track of the actual number of positive and negative particles $N_{\mathrm{B},j}^{\pm}$ which are accepted and added to the simulation. Finally, particle generation steps are repeated until precisely $N_{\mathrm{B},j}^{-}$ positive and $N_{\mathrm{B},j}^{+}$ negative particles are added to the simulation, rejecting all accepted particles of unneeded sign.

C. Effective body force step

In the linearized regime, it is possible to simulate streamwise pressure and temperature gradients in a long duct without simulating the streamwise direction (z) in physical space by introducing an effective "body force" term into the simulation.[21,25,28–30] This approach was pioneered by Cercignani and Daneri[25] as a mathematical formulation of pressure-driven flow in small capillaries; using this formulation, Cercignani and Daneri[25] proceeded to solve the Boltzmann equation in the relaxation approximation numerically for a two-dimensional channel geometry and thus theoretically verify, for the first time, the existence of a Knudsen minimum in the scaled flow rate as a function of nondimensional channel height (see also Fig. 2). This minimum was originally experimentally observed by Knudsen.[31]

If we let $\kappa_{\mathrm{P}} = -\frac{1}{P}\frac{dP}{dz}$ and $\kappa_{\mathrm{T}} = \frac{1}{T}\frac{dT}{dz}$ denote the scaled pressure and temperature gradients, the change in the distribution function due to these two effects is given by

$$\frac{\partial f(\boldsymbol{c})}{\partial t} V = V c_z \left[\kappa_{\mathrm{P}} + \left(\frac{5}{2} - \frac{\|\boldsymbol{c} - \boldsymbol{u}_0\|^2}{c_0^2} \right) \kappa_{\mathrm{T}} \right] f^0(\boldsymbol{c}). \tag{41}$$

For the choice of $f^0(\boldsymbol{c})$ considered here, namely, $\boldsymbol{u}_0 = \mathbf{0}$, we perform the sampling in terms of $\boldsymbol{\xi}$, as was done for advection as shown below,

$$\frac{\partial f(\boldsymbol{c})}{\partial t} V \mathrm{d}^3\boldsymbol{c} = \frac{V \rho_0 c_0}{L} F^{\mathrm{F}}(\boldsymbol{c}) \mathrm{d}^3\boldsymbol{\xi},$$

$$F^{\mathrm{F}}(\boldsymbol{c}) = \frac{c_0^2 L}{\rho_0} c_z \left[\kappa_{\mathrm{P}} + \left(\frac{5}{2} - \xi^2 \right) \kappa_{\mathrm{T}} \right] f^0(\boldsymbol{c}). \tag{42}$$

Here, L is a physical length scale used to make distribution F^{F} dimensionless. This distribution is sampled using the ratio-of-uniforms method[21] in the transformed variable space: $|F^{\mathrm{F}}| = H^{5/2}$, $\boldsymbol{\xi} = \boldsymbol{\eta}/\sqrt{H}$ (cf. Sec. II B), with bounds

$$0 \le H \le a^{\mathrm{F}}, \tag{43}$$

$$-b_x^{\mathrm{F}} \le \eta_x \le b_x^{\mathrm{F}}, \tag{44}$$

$$-b_y^{\mathrm{F}} \le \eta_y \le b_y^{\mathrm{F}}, \tag{45}$$

$$-b_z^{\mathrm{F}} \le \eta_z \le b_z^{\mathrm{F}}. \tag{46}$$

Analytical bounds were derived for small perturbations from equilibrium,[21]

$$\begin{bmatrix} (a^{\mathrm{F},0})^{5/2} \\ (b_x^{\mathrm{F},0})^5 \\ (b_y^{\mathrm{F},0})^5 \\ (b_z^{\mathrm{F},0})^5 \end{bmatrix} = \mathbf{M}_{\mathrm{F}} \cdot \begin{bmatrix} \left|\kappa_{\mathrm{P}} + \frac{5}{2}\kappa_{\mathrm{T}}\right| \\ |\kappa_{\mathrm{T}}| \end{bmatrix} L, \tag{47}$$

where

$$\mathbf{M}_{\mathrm{F}} = \frac{1}{\pi^{3/2}} \begin{bmatrix} 1/\sqrt{2e} & [3/(2e)]^{3/2} \\ 5^{5/2}/(2e)^3 & 20^{5/2}/(27e^4) \\ 5^{5/2}/(2e)^3 & 20^{5/2}/(27e^4) \\ (3/e)^3 & (4/e)^4 \end{bmatrix}. \tag{48}$$

Since the effective body force approach is only valid for small perturbations from equilibrium, there is no reason to extend these bounds to larger deviations from equilibrium as was done for the advection routine (Sec. II B). Thus, we shall use $a^{\mathrm{F}}=a^{\mathrm{F},0}$ and $\boldsymbol{b}^{\mathrm{F}}=\boldsymbol{b}^{\mathrm{F},0}$ for the bounds in the simulation.

Based on the advective time step, the number of trial samples to generate is computed as

$$N_{\mathrm{F}}^{\mathrm{trial}} = \frac{5}{2}\frac{V\rho_0 c_0 \Delta t_{\mathrm{adv}}}{mWL} a^{\mathrm{F}}(2b_x^{\mathrm{F}})(2b_y^{\mathrm{F}})(2b_z^{\mathrm{F}}). \tag{49}$$

As before, a sample $(H, \boldsymbol{\eta})$ is generated (uniformly) utilizing bounds (43)–(46) for each trial step. Using $\boldsymbol{c}=c_0\boldsymbol{\eta}/\sqrt{H}$, F^{F} is evaluated from Eq. (42), and the trial particle generation is accepted if $H<|F^{\mathrm{F}}|^{2/5}$. Accepted particles are added to the simulation with sign $\mathrm{sgn}(F^{\mathrm{F}})$ and with a position $\boldsymbol{x}$ sampled uniformly from V.

1. Mass conservation

Mass conservation is again enforced via stratified sampling (cf. Sec. II B). In the first step, $N_{\mathrm{F}}^{\mathrm{trial}}/2$ trial generation steps are performed, which adds $N_{\mathrm{F}}^{\pm}$ positive and negative particles to the simulation. Additional particle generation steps are performed to produce $N_{\mathrm{F}}^{\mp}$ additional positive and negative particles.

D. Property evaluation

Hydrodynamic properties are evaluated by simple extensions to the rules developed for evaluating DSMC properties.[12,13,21,22] For example, the DSMC estimate of temperature in the jth cell is obtained from

$$\rho_j(3RT_j + u_j^2) = \frac{mN_{\mathrm{eff}}}{V_j}\sum_{k\in\mathcal{N}_j} c_k^2. \tag{50}$$

In the LVDSMC approach, the above summation now includes the sign and corresponds to the difference between the group of cell properties $\rho_j(3RT_j+u_j^2)$ and the corresponding equilibrium values [Eq. (54), below]. These results are summarized below for the density, mean velocity, pressure tensor $(\mathbf{P})$, temperature, and heat flux $(\boldsymbol{q})$, where $\mathbf{P}_0=\rho_0 RT_0\mathbf{I}$ and $\mathbf{I}$ is the identity tensor,

$$\rho_j = \rho_0 + \frac{mW}{\Delta V_j}\sum_{k\in\mathcal{N}_j} s_k, \tag{51}$$

$$\rho_j \boldsymbol{u}_j = \rho_0 \boldsymbol{u}_0 + \frac{mW}{\Delta V_j}\sum_{k\in\mathcal{N}_j} s_k \boldsymbol{c}_k, \tag{52}$$

$$\mathbf{P}_j + \rho \boldsymbol{u}_j \boldsymbol{u}_j = \mathbf{P}_0 + \rho_0 \boldsymbol{u}_0 \boldsymbol{u}_0 + \frac{mW}{\Delta V_j}\sum_{k\in\mathcal{N}_j} s_k \boldsymbol{c}_k \boldsymbol{c}_k, \tag{53}$$

$$\rho_j(3RT_j + u_j^2) = \rho_0(3RT_0 + u_0^2) + \frac{mW}{\Delta V_j}\sum_{k\in\mathcal{N}_j} s_k c_k^2, \tag{54}$$

$$2(\boldsymbol{q}_j + \mathbf{P}_j \cdot \boldsymbol{u}_j) + \rho_j(3RT_j + u_j^2)\boldsymbol{u}_j = 2\mathbf{P}_0 \cdot \mathbf{u}_0 + \rho_0(3RT_0 + u_0^2)\boldsymbol{u}_0 + \frac{mW}{\Delta V_j}\sum_{k\in\mathcal{N}_j} s_k \boldsymbol{c}_k c_k^2. \tag{55}$$

E. Time step

In order to improve the overall rate of time convergence, a symmetrized version of the algorithm was implemented. Previous convergence studies[32] have shown that Strang's method[33] achieves second-order time convergence for DSMC; we adopt this approach, with the additional effective body force generation terms (not appearing in DSMC) symmetrically split around collision step as shown below based on an overall time step of Δt.

Algorithm 2. Symmetrized time stepping algorithm:

1. Half advection $\left(\Delta t_{\mathrm{adv}}=\frac{1}{2}\Delta t\right)$;
2. Half body force $\left(\Delta t_{\mathrm{adv}}=\frac{1}{2}\Delta t\right)$;
3. Full collision $(\Delta t_{\mathrm{col}}=\Delta t)$;
4. Half body force $\left(\Delta t_{\mathrm{adv}}=\frac{1}{2}\Delta t\right)$;
5. Half advection $\left(\Delta t_{\mathrm{adv}}=\frac{1}{2}\Delta t\right)$; and
6. Sample properties.

III. RESULTS

We performed a number of two-dimensional simulations in order to highlight the key features of the method and to showcase its ability to efficiently simulate problems with arbitrarily small deviations from equilibrium with drastically reduced levels of statistical noise. In all cases presented here, we simulate the deviation from a global equilibrium distribution (with $\boldsymbol{u}_0=\mathbf{0}$). The normalized characteristic deviation from equilibrium is quantified by ϵ, which is typically related to the characteristic temperature difference $(\epsilon=\Delta T/T_0)$ or velocity (e.g., $\epsilon=u_x/c_0$) of the problem. We note that, in contrast to DSMC, the cost of the proposed method does not increase as ϵ decreases. For this reason, all results presented here are scaled by ϵ.

In order to verify correct representation of the VHS collision operator, viscosities were obtained for hard-sphere, he-

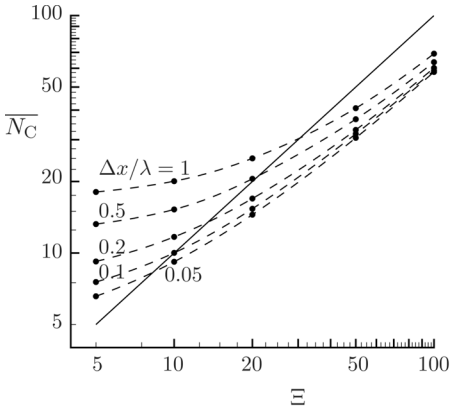

FIG. 1. Average number of deviational particles per cell for a heat flux through a layer of argon gas confined between parallel plates with $Kn=0.1$. The data (symbols) are shown in terms of the computational parameters $(\Delta x, \Xi)$, while the line indicates $\overline{N_C}=\Xi$ for comparison.

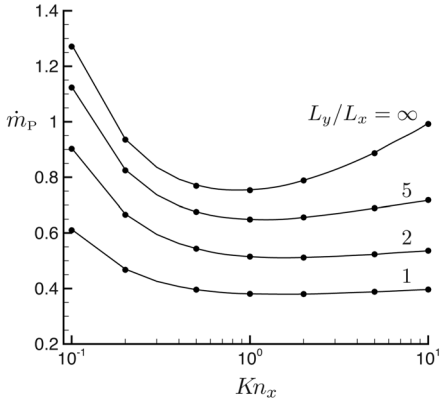

FIG. 2. Flow rate for Poiseuille flow through a rectangular microchannel for various Knudsen numbers and aspect ratios. The LVDSMC results (symbols) are compared with data from Doi (Ref. 34) (lines).

lium, argon, and Maxwell molecules ($\omega=0.5$, 0.66, 0.81, and 1, respectively) and compared to results for the DSMC method in a $Kn=0.05$ shear flow. Excellent agreement was observed.

The accuracy of the LVDSMC method, like the DSMC method on which it is based, depends on the spatial cell size Δx, the overall time step Δt, and the average number of computational particles per cell $\overline{N_C}$. The LVDSMC simulation approach utilizes Δx and Δt in similar ways to DSMC; however, the average number of computational particles per cell $\overline{N_C}$ has a dramatically different behavior in each method and merits further discussion.

In the DSMC method, the number of simulation particles is well defined in terms of problem and discretization parameters. For example, the average number of particles per cell for a simulation with zero-mass-flux boundary conditions is given by

$$\overline{N_C} = \frac{\rho_0 \overline{\Delta V}}{m N_{\text{eff}}}, \tag{56}$$

where $\overline{\Delta V}=V/N_{\Delta V}$ is the average cell volume; here, we have taken ρ_0 to be the density of the initial state. However, for the LVDSMC method, the local number of particles depends on the local deviation from equilibrium, and as a result, $\overline{N_C}$ depends on the "average" degree of deviation from equilibrium ϵ, for which no established measure exists.

For a suitably defined ϵ and for simulations in the transition regime $(0.1 \le Kn \le 10)$, we have observed that the number of particles per cell, at a nontrivial steady state, can be approximately scaled using

$$\Xi = \frac{\epsilon \rho_0 \overline{\Delta V}}{mW} \tag{57}$$

in the sense that $\overline{N_C} \sim \Xi$. In other words, instead of using $\overline{N_C}$ as a separate convergence parameter (as in DSMC), the parameter Ξ is used. This is illustrated in Fig. 1 which shows the actual number of particles for various values of Δx and Ξ for heat transfer between parallel plates at $Kn=0.1$; the gas is argon $(\omega=0.81)$ and the boundary conditions are diffusely reflecting $(\alpha=1)$ with temperatures $T(0)=(1+\epsilon)T_0$ and $T(L)=(1-\epsilon)T_0$, where $\epsilon \ll 1$. The figure shows that for large Ξ, there is a direct relationship between $\overline{N_C}$ and Ξ, while for small Ξ, the smaller number of particles makes the "particle cancellation effect" in Eq. (16) less effective (see Sec. II A and discussion in Ref. 23) leading to a larger number of particles. For all the simulation results presented in this section, excellent results were achieved using $\Xi=10$. Such results are significant because they demonstrate that simulations with approximately ten particles per cell are achievable in mass-conservative LVDSMC simulations *without apparent random walks in any non-negligible hydrodynamic variables*, a substantial improvement over the previous implementation.[26] This dramatic improvement enables efficient simulation in multiple spatial dimensions, as we show below.

Here, the time step was chosen as $\Delta t=\Delta x/c_0$, where Δx is the smallest cell dimension, which effectively treats the effect of Δx and Δt as a single convergence parameter (Δx). More rigorous convergence studies for the method are needed, which are left to future studies. For all simulations with $Kn>1$, the boundary cancellation procedure (see discussion in Sec. II B) was used. While the simulations performed in this work (up to $Kn=10$) remained stable without boundary cancellation, the number of simulated particles per cell tended to scale with ΞKn (rather than Ξ) and the overall computational efficiency of the method was noticeably degraded. For $Kn \le 1$, the boundary cancellation procedure was unnecessary and was not used.

A. Poiseuille and thermal creep flow in a rectangular microchannel

As a validation of the overall method in two-dimensional geometries, as well as to highlight an application of the effective body force term (Sec. II C), Poiseuille $(\epsilon=\kappa_P L_x \ll 1)$ and thermal creep $(\epsilon=\kappa_T L_x \ll 1)$ flows of a hard-sphere gas were simulated for a rectangular microchannel geometry with cross section $L_x \times L_y$. The Knudsen number Kn_x is defined as $Kn_x=\lambda/L_x$, where $\lambda^{-1}=\sqrt{2}\pi(\rho_0/m)d_{\text{ref}}^2$. Shown in Figs. 2 and 3 are the dimensionless flow rates $\dot{m}_{P,T}=\overline{u_z}/(\epsilon c_0)$ for various aspect ratios as a function of Kn_x, where the overbar denotes the spatial average in x and y.

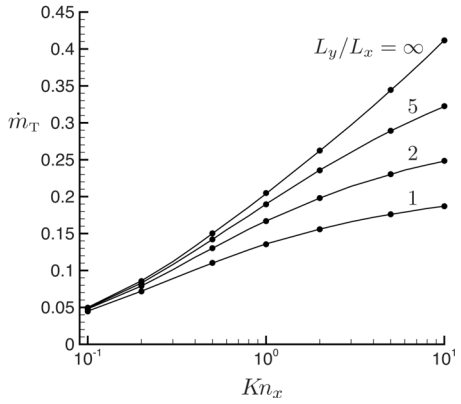

FIG. 3. Flow rate for thermal creep flow through a rectangular microchannel for various Knudsen numbers and aspect ratios. The LVDSMC results (symbols) are compared with data from Doi (Ref. 34) (lines).

Due to twofold symmetry, only a quarter of the channel cross section was simulated. For most cases, a cell size of $\Delta x/L_x=\Delta y/L_x=0.02$ was used to obtain better than 1% agreement in the total mass flow rates compared to the results of Doi.[34] For many cases with $Kn=0.1$, further refinement was required to obtain the same level of agreement, and $\Delta x/L_x=\Delta y/L_x=0.01$ was used.[35]

Figure 4 shows the velocity field for Poiseuille flow through a square microchannel ($L=L_x=L_y$) for $Kn=0.1$ using 50×50 spatial cells. By performing steady-state averaging over 10^6 time steps ($\Delta t=\Delta x/c_0$) after steady state was reached, this simulation resulted in a velocity field with a relative statistical uncertainty[8] of ~0.1%. In order to obtain a 0.1% statistical uncertainty in $\dot{m}_P$, 5×10^5 time steps are required, using approximately 16 h on a single core of an Intel Q9650 (3.0 GHz Core 2 Quad) processor. For square channels with $Kn=1$ and 10, with a 25×25 cell mesh, 1.1 and 0.2 h of computational times, respectively, were required to achieve the same level of relative statistical uncertainty in $\dot{m}_P$. Given that (even in variance-reduced guise) Monte Carlo approaches will always perform worse when *very* low noise is required, this performance is very encouraging for *highly resolved two-dimensional* calculations.

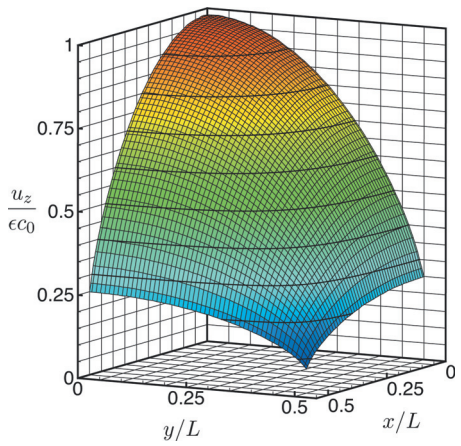

FIG. 4. (Color) Streamwise velocity for the Poiseuille flow through a square microchannel with $Kn=0.1$.

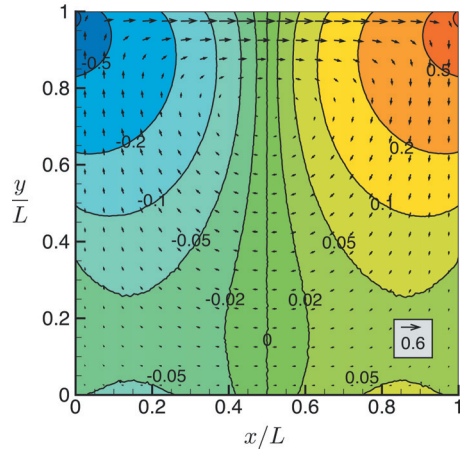

FIG. 5. (Color) Lid-driven flow of argon gas at $Kn=0.1$. The contour lines show the density $\epsilon^{-1}(\rho/\rho_0-1)$, while the velocity field $\epsilon^{-1}\boldsymbol{u}/c_0$ is shown as a vector plot.

As an indication of the relative efficiency compared to DSMC, we compare the simulation time required to achieve 0.1% statistical uncertainty in the velocity field for Poiseuille flow through a square channel. Assuming a Mach number of $Ma\simeq0.02$, DSMC simulations would require approximately 500, 100, and 100 h for $Kn=0.1$, 1, and 10, respectively, compared to 30, 4, and 2 h for the LVDSMC simulations. For this problem (Poiseuille flow), DSMC simulates the pressure force as an equivalent gravitation force, which is a valid approach for small deviations from equilibrium. However, there is no obvious way to use DSMC to simulate thermal creep without resorting to very expensive three-dimensional simulations.

B. Lid-driven flow of argon gas

Next, we simulate a two-dimensional lid-driven flow of argon ($\omega=0.81$) gas in a square enclosure with side length L. The boundary conditions are diffusely reflecting walls, all of which are stationary except the top ($y=L$) which is moving in the x-direction with velocity ϵc_0, where $\epsilon\ll1$. We performed simulations for $Kn=\lambda/L=0.1$, 1, and 10; 100×100 cells were used for $Kn=0.1$, while 50×50 cells were used for $Kn=1$ and 10. Here, the mean free path is given by the VHS value: $\lambda^{-1}=\sqrt{2}\pi(\rho_0/m)d_{\mathrm{ref}}^2(T_{\mathrm{ref}}/T_0)^{\omega-1/2}$. Each simulation was repeated with a doubling of the number of cells in each coordinate direction (to 200×200 and 100×100, respectively), which showed less than 1% difference in ρ, u_x, and u_y; this was taken as evidence of convergence. Shown in Figs. 5–7 are the velocity and density fields corresponding to the finer-meshed solutions.

C. Response of a gas to a spatially varying boundary temperature

Finally, we simulate the response of argon gas to a boundary temperature with a sinusoidal spatial variation. Here, the lower boundary ($y=0$) is diffusely reflecting with a temperature given by $T_B=T_0(1-\epsilon\cos2\pi x/L)$; an identical boundary is located at $y=L$, and the Knudsen number based on the separation between the two boundaries (L) is $Kn=1$.

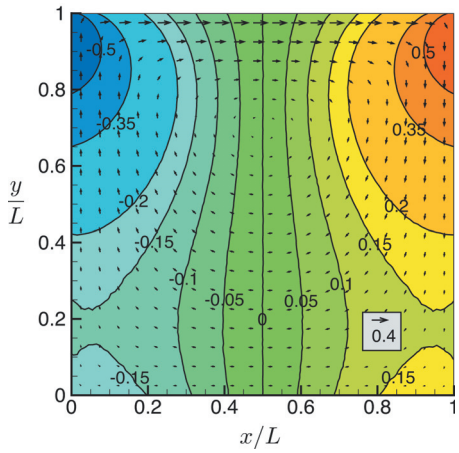

FIG. 6. (Color) Lid-driven flow of argon gas at $Kn=1$. The contour lines show the density $\epsilon^{-1}(\rho/\rho_0-1)$, while the velocity field $\epsilon^{-1}\boldsymbol{u}/c_0$ is shown as a vector plot.

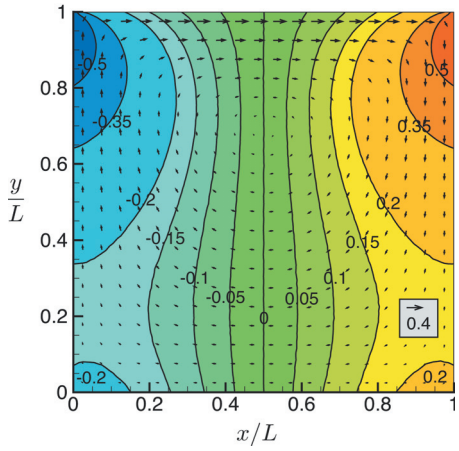

FIG. 7. (Color) Lid-driven flow of argon gas at $Kn=10$. The contour lines show the density $\epsilon^{-1}(\rho/\rho_0-1)$, while the velocity field $\epsilon^{-1}\boldsymbol{u}/c_0$ is shown as a vector plot.

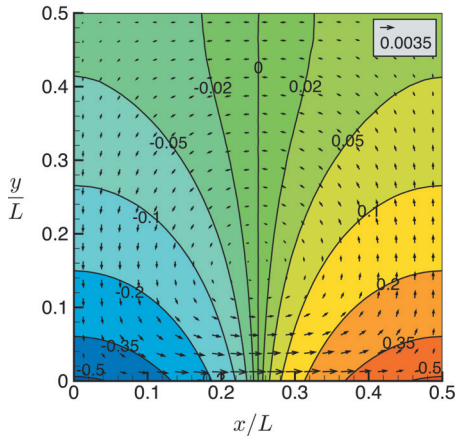

FIG. 8. (Color) Response of argon gas to spatially varying boundary temperature with $Kn=1$ and $\epsilon\ll 1$. The contour lines are isotherms [dimensionless temperature: $\epsilon^{-1}(T/T_0-1)$], while the velocity field $\epsilon^{-1}\boldsymbol{u}/c_0$ is shown as a vector plot.

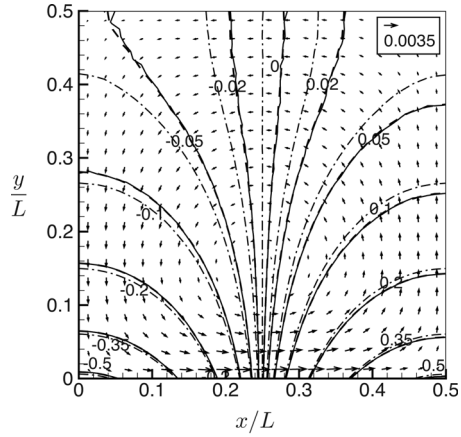

FIG. 9. Response of argon gas to spatially varying boundary temperature with $Kn=1$ and $\epsilon=0.05$. Contour plot of the dimensionless temperature $[\epsilon^{-1}(T/T_0-1)]$ as obtained by LVDSMC (dashed) and DSMC (solid); the $\epsilon\to 0$ limit (dashed-dotted) as obtained by LVDSMC is also shown for comparison. The velocity field $\epsilon^{-1}\boldsymbol{u}/c_0$ is the LVDSMC solution for $\epsilon=0.05$; the DSMC velocity field was noticeably noisier.

Due to the underlying symmetries in the x and y directions, the simulation domain is chosen as $0\le x,y\le L/2$. Unlike the previous examples, here we show results for several choices of ϵ (even though the collision operator is linearized). Shown in Fig. 8 are the temperature and velocity fields for the limit of small departure from equilibrium $\epsilon\ll 1$. In Figs. 9 and 10, the isotherms for the LVDSMC and DSMC methods are compared for $\epsilon=0.05$ and 0.5, respectively. For $\epsilon=0.05$, there is no noticeable difference between the LVDSMC and DSMC temperature fields even though the temperature field is noticeably perturbed from the $\epsilon\to 0$ solution. For $\epsilon=0.5$, which is no longer a near-equilibrium case, there is only a slight discrepancy between the temperature fields obtained from LVDSMC and DSMC.

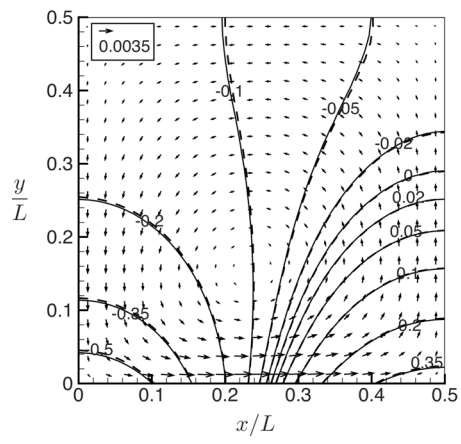

FIG. 10. Response of argon gas to spatially varying boundary temperature with $Kn=1$ and $\epsilon=0.5$. Contour plot of the dimensionless temperature $[\epsilon^{-1}(T/T_0-1)]$ as obtained by LVDSMC (dashed) and DSMC (solid). The velocity field $\epsilon^{-1}\boldsymbol{u}/c_0$ is the LVDSMC solution for $\epsilon=0.5$.

IV. DISCUSSION

We have presented a low-variance stochastic particle method which is capable of efficiently simulating kinetic flows in the linear regime. This method simulates the VHS collision operator, which is a more general and realistic model than the hard-sphere collision operator covered by previous methods. By incorporating mass conservation into the LVDSMC methodology, the number of simulation particles per cell required to produce accurate results was reduced to approximately 10, which is comparable to that required for the DSMC method.

Future research directions include further analysis and extension of this methodology. In particular, the nonlinear version of this algorithm[26] is currently being extended to include mass conservation. Likewise, additional flow simulations and validations are needed, as well as a rigorous convergence study to investigate convergence behavior in Δx, Δt, and Ξ (as has been done for DSMC[32]).

The present formulation provides variance reduction by simulating the deviation from a (constant) global equilibrium. A formulation using a spatially variable (cell based) equilibrium distribution can be achieved using a number of implementations;[13,21,23] the implementation complementing the algorithm presented here has been outlined previously.[23] As shown in previous publications,[13,21] variable equilibrium formulations provide superior variance reduction, especially in the collision-dominated limit where the local equilibrium assumption becomes very reasonable. In fact, because as $Kn \to 0$ the distribution function is increasingly better approximated by a local equilibrium distribution,[20] such formulations can capture arbitrarily small values of Kn without becoming prohibitively expensive (e.g., see Ref. 36). In other words, such methods alleviate the stiffness associated with recovering the continuum limit with molecular simulations and offer promising avenues for developing multiscale methods that can seamlessly connect the continuum and molecular descriptions. Despite this potential, a spatially variable equilibrium distribution was not used here because it requires particle generation at cell interfaces (where the equilibrium distribution changes discontinuously[13,21]), making the method cumbersome in high number of dimensions. Perhaps a continuously varying equilibrium distribution (requiring volumetric particle generation) will reduce the complexity associated with this approach.

The multiscale implications of decomposing the distribution were noticed by Cheremisin,[37] who proposed simulating the deviation from equilibrium (using a discrete velocity method) as a means of removing the stiffness associated with time integration in the Navier–Stokes limit ($Kn \to 0$). Also, motivated by their interest in using particle methods for approaching the fluid-dynamic limit for high Mach-number flows, Caflisch and Pareschi[38] proposed a convex decomposition of the distribution function into a time-dependent Maxwellian and a nonequilibrium distribution; unfortunately, the requirement that the equilibrium distribution is time dependent and the nonequilibrium distribution is represented by positive particles only results in a complex algorithm which requires that the equilibrium distribution be reconstructed (from its samples) every time step.

ACKNOWLEDGMENTS

This paper is dedicated to Professor Carlo Cercignani whose contributions to the field of Rarefied Gas Dynamics are too numerous to list. Professor Cercignani has made contributions both to the mathematical theory (e.g., in stochastic models[39]) and the physics of rarefied gases. Despite his strong mathematical inclination, he has made contributions that continue to have enormous impact in engineering applications: for example, using the relaxation-time approximation, he was the first to obtain results for slip flow coefficients;[40,41] he was also the first to provide a mathematical formulation and complete (numerical) solution of the pressure-driven flow (Knudsen minimum) problem.[25] Moreover, his books on the Boltzmann equation and kinetic theory are invaluable and beautifully written resources.

The authors gratefully acknowledge the support of Singapore-MIT alliance. N.G.H. would like to thank the Weierstrass Institute for Applied Analysis and Stochastics for facilitating his visit in July 2009.

[1] N. G. Hadjiconstantinou and O. Simek, "Constant-wall-temperature Nusselt number in micro and nano-channels," J. Heat Transfer **124**, 356 (2002).
[2] N. G. Hadjiconstantinou, "Dissipation in small scale gaseous flows," J. Heat Transfer **125**, 944 (2003).
[3] A. A. Alexeenko, S. F. Gimelshein, E. P. Muntz, and A. D. Ketsdever, "Kinetic modeling of temperature driven flows in short microchannels," Int. J. Therm. Sci. **45**, 1045 (2006).
[4] N. G. Hadjiconstantinou, "The limits of Navier-Stokes theory and kinetic extensions for describing small-scale gaseous hydrodynamics," Phys. Fluids **18**, 111301 (2006).
[5] Y. L. Han, E. P. Muntz, A. Alexeenko, and Y. Markus, "Experimental and computational studies of temperature gradient-driven molecular transport in gas flows through nano/microscale channels," Nanoscale Microscale Thermophys. Eng. **11**, 151 (2007).
[6] C. Cercignani, *Slow Rarefied Flows: Theory and Application to Micro-Electro-Mechanical Systems* (Birkhauser-Verlag, Basel, 2006).
[7] G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows* (Clarendon, Oxford, 1994).
[8] N. G. Hadjiconstantinou, A. L. Garcia, M. Z. Bazant, and G. He, "Statistical error in particle simulations of hydrodynamic phenomena," J. Comput. Phys. **187**, 274 (2003).
[9] L. L. Baker and N. G. Hadjiconstantinou, "Variance reduction for Monte Carlo solutions of the Boltzmann equation," Phys. Fluids **17**, 051703 (2005).
[10] S. Asmussen and P. W. Glynn, *Stochastic Simulation: Algorithms and Analysis* (Springer, New York, 2007).
[11] L. L. Baker and N. G. Hadjiconstantinou, "Variance reduction in particle methods for solving the Boltzmann equation," *Proceedings of the 4th International Conference on Nanochannels, Microchannels, and Minichannels*, Limerick, Ireland (ASME, New York, 2006), pp. 377–383.
[12] T. M. M. Homolle and N. G. Hadjiconstantinou, "Low-variance deviational simulation Monte Carlo," Phys. Fluids **19**, 041701 (2007).
[13] T. M. M. Homolle and N. G. Hadjiconstantinou, "A low-variance deviational simulation Monte Carlo for the Boltzmann equation," J. Comput. Phys. **226**, 2341 (2007).
[14] L. L. Baker and N. G. Hadjiconstantinou, "Variance-reduced particle methods for solving the Boltzmann equation," J. Comput. Theor. Nanosci. **5**, 165 (2008).
[15] J. Chun and D. L. Koch, "A direct simulation Monte Carlo method for rarefied gas flows in the limit of small Mach number," Phys. Fluids **17**, 107107 (2005).
[16] H. A. Al-Mohssen and N. G. Hadjiconstantinou, "Low-variance direct Monte Carlo using importance weights," Math. Modell. Numer. Anal. **44**, 1069 (2010).

[17]H. C. Öttinger, *Stochastic Processes in Polymeric Fluids* (Springer-Verlag, New York, 1995).
[18]N. J. Wagner and H. C. Öttinger, "Accurate simulation of linear viscoelastic properties by variance reduction through the use of control variates," J. Rheol. **41**, 757 (1997).
[19]C. Cercignani, *Mathematical Methods in Kinetic Theory*, 2nd ed. (Plenum, New York, 1990).
[20]C. Cercignani, *The Boltzmann Equation and Its Applications* (Springer Verlag, New York, 1988).
[21]G. A. Radtke and N. G. Hadjiconstantinou, "Variance-reduced particle simulation of the Boltzmann transport equation in the relaxation-time approximation," Phys. Rev. E **79**, 056711 (2009).
[22]N. G. Hadjiconstantinou, G. A. Radtke, and L. L. Baker, "On variance reduced simulations of the Boltzmann transport equation for small-scale heat transfer applications," J. Heat Transfer **132**, 112401 (2010).
[23]W. Wagner, "Deviational particle Monte Carlo for the Boltzmann equation," Monte Carlo Meth. Appl. **14**, 191 (2008).
[24]G. A. Bird, "Monte Carlo simulation in an engineering context," Prog. Astronaut. Aeronaut. **74**, 239 (1981).
[25]C. Cercignani and A. Daneri, "Flow of a rarefied gas between two parallel plates," J. Appl. Phys. **34**, 3509 (1963).
[26]G. A. Radtke, N. G. Hadjiconstantinou, and W. Wagner, "Low variance particle simulations of the Boltzmann transport equation for the variable hard sphere collision model," in *Proceedings of the 27th International Symposium on Rarefied Gas Dynamics*, Pacific Grove, CA, edited by D. A. Levin, I. J. Wysong, and A. L. Garcia (AIP, Melville, NY, 2011).
[27]J. C. Wakefield, A. E. Gelfand, and A. F. M. Smith, "Efficient generation of random variates via the ratio-of-uniforms method," Stat. Comput. **1**, 129 (1991).
[28]S. K. Loyalka and J. W. Cipolla, Jr., "Thermal creep slip with arbitrary accommodation at the surface," Phys. Fluids **14**, 1656 (1971).
[29]S. K. Loyalka, N. Petrellis, and T. S. Storvick, "Some numerical results for the BGK model: Thermal creep and viscous slip problems with arbitrary accommodation at the surface," Phys. Fluids **18**, 1094 (1975).
[30]M. R. Allshouse and N. G. Hadjiconstantinou, "Low-variance deviational Monte Carlo simulations of pressure driven flow in micro- and nanoscale channels," in *Proceedings of the 26th International Symposium on Rarefied Gas Dynamics*, Kyoto, Japan, edited by T. Abe (AIP, Melville, NY, 2008), pp. 1015–1020.
[31]M. Knudsen, "Die Gesetze der Molekularströmung und der inneren Reibungsströmung der Gase durch Röhren," Ann. Phys. **333**, 75 (1909).
[32]D. J. Rader, M. A. Gallis, J. R. Torczynski, and W. Wagner, "DSMC convergence behavior of the hard-sphere-gas thermal conductivity for Fourier heat flow," Phys. Fluids **18**, 077102 (2006).
[33]T. Ohwada, "Higher order approximation methods," J. Comput. Phys. **139**, 1 (1998).
[34]T. Doi, "Numerical analysis of the Poiseuille flow and the thermal transpiration of a rarefied gas through a pipe with a rectangular cross section based on the linearized Boltzmann equation for a hard sphere molecular gas," J. Vac. Sci. Technol. A **28**, 603 (2010).
[35]For thermal creep with $Kn=0.1$ and $L_y/L_x=2$, Doi (Ref. 34) reported a value of $\dot{m}_T=0.048$ compared to 0.0473 for the LVDSMC method. Due to the small number of digits in the reported value, it was not possible to determine if 1% agreement was attained for this specific case.
[36]A. Manela and N. G. Hadjiconstantinou, "Gas-flow animation by unsteady heating in a microchannel," Phys. Fluids **22**, 062001 (2010).
[37]F. G. Cheremisin, "Solving the Boltzmann equation in the case of passing to the hydrodynamic flow regime," Dokl. Phys. **45**, 401 (2000).
[38]R. E. Caflisch and L. Pareschi, in *Transport in Transition Regimes*, edited by N. Ben Abdallah, A. Arnold, P. Degond, I. M. Gamba, R. T. Glassey, C. D. Levermore, and C. Ringhofer (Springer-Verlag, New York, 2004), Vol. 135, pp. 57–73.
[39]W. Wagner, "Stochastic models in kinetic theory," Phys. Fluids **23**, 030602 (2011).
[40]C. Cercignani and A. Daneri, "Elementary solutions of the linearized gas-dynamics Boltzmann equation and their application to the slip-flow problem," Ann. Phys. **20**, 219 (1962).
[41]S. Albertoni, C. Cercignani, and L. Gotusso, "Numerical evaluation of the slip coefficient," Phys. Fluids **6**, 993 (1963).