INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **58**:381–402
Published online 26 February 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1724

Variance-reduced Monte Carlo solutions of the Boltzmann equation for low-speed gas flows: A discontinuous Galerkin formulation

Lowell L. Baker and Nicolas G. Hadjiconstantinou*, †

Mechanical Engineering Department, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.

SUMMARY

We present and discuss an efficient, high-order numerical solution method for solving the Boltzmann equation for low-speed dilute gas flows. The method's major ingredient is a new Monte Carlo technique for evaluating the weak form of the collision integral necessary for the discontinuous Galerkin formulation used here. The Monte Carlo technique extends the variance reduction ideas first presented in Baker and Hadjiconstantinou (*Phys. Fluids* 2005; **17**, art. no. 051703) and makes evaluation of the weak form of the collision integral not only tractable but also very efficient. The variance reduction, achieved by evaluating only the deviation from equilibrium, results in very low statistical uncertainty and the ability to capture arbitrarily small deviations from equilibrium (e.g. low-flow speed) at a computational cost that is independent of the magnitude of this deviation. As a result, for low-signal flows the proposed method holds a significant computational advantage compared with traditional particle methods such as direct simulation Monte Carlo (DSMC). Copyright © 2008 John Wiley & Sons, Ltd.

Received 14 August 2007; Revised 13 November 2007; Accepted 20 November 2007

KEY WORDS: Boltzmann equation; numerical solution; dilute gas; small scale; kinetic theory; DSMC

1. INTRODUCTION

Interest in numerical solution of the original Boltzmann equation for dilute gas flow [1, 2] has recently been revived in connection with small-scale science and technology [3]. In this regime, characteristic flow scales are comparable to (or smaller than) the molecular mean free path, and as a result, modeling beyond the Navier–Stokes level of description is often required. Numerical solution of the Boltzmann equation is a formidable task due to the high dimensionality of the

*Correspondence to: Nicolas G. Hadjiconstantinou, Mechanical Engineering Department, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.
†E-mail: ngh@mit.edu

Contract/grant sponsor: Sandia National Laboratory

molecular distribution function and the difficulty associated with the evaluation of the collision term (a high-dimensional integral) of this equation [1, 2]. For these reasons, and because interest in numerical solutions of the Boltzmann equation was for a long time focused on description of high-speed flows studied in connection with flight in the upper atmosphere, Monte Carlo approaches have prevailed over direct numerical solutions. In particular, the prevalent Boltzmann solution method is currently a stochastic particle simulation method known as direct simulation Monte Carlo (DSMC) [4]. Unfortunately, DSMC becomes very inefficient for the low-signal (e.g. low-speed) flows typical of micro- and nano-scale applications [3, 5] due to the uncertainty associated with the statistical sampling of hydrodynamic properties from particle data. As an example, consider the flow velocity as quantified by the local Mach number Ma: it can be shown [6] that to retain a constant signal to noise ratio in the flow velocity as this decreases ($Ma \to 0$), the computational cost of DSMC increases as Ma^{-2}. In other words, this limitation is sufficiently severe to make noise-free simulation of low-signal flows very expensive and in some cases intractable.

Recently, Baker and Hadjiconstantinou [7] showed that this serious limitation suffered by traditional Monte Carlo approaches in the limit of small deviation from equilibrium can be addressed using variance reduction ideas for evaluating the collision integral. In particular, they showed that by simulating only the deviation from equilibrium, it is possible to construct Monte Carlo simulation methods that can accurately capture arbitrarily small deviations from equilibrium at a computational cost that is independent of the magnitude of this deviation. The benefits of variance reduction were demonstrated in [7] where it was applied to a finite volume formulation of the Boltzmann equation in which the collision integral was treated as a source term.

The present paper extends this variance reduction approach for evaluating the collision integral to the weak form of the collision integral and presents an efficient implementation that enables higher-order solutions of the Boltzmann equation through a Runge–Kutta discontinuous Galerkin (RKDG) [8] formulation. In the resulting implementation, the collision integral operator evaluation is typically of comparable computational cost to the evaluation of the advection operator, thus making the application of the discontinuous Galerkin (DG) formulation (and possibly other sophisticated numerical formulations) to the original Boltzmann equation possible. At the same time, the variance reduction employed within the formulation endows it with very low statistical uncertainty—essentially negligible as will be seen later—and the ability to capture arbitrarily small deviations from equilibrium at a cost that *does not* scale with the magnitude of this deviation. The resulting method is thus both efficient and high-order accurate in *all* dimensions: time, physical space and velocity space.

The DG method [8, 9] has become a popular finite element approach for obtaining solutions to *hyperbolic* equations (or systems of equations) due to its ability to provide high-order solutions on irregular meshes in several dimensions, while allowing for the solution of problems exhibiting discontinuities [8, 10]. In our context, it provides a robust framework that can accurately capture traveling discontinuities in the distribution function—traditionally a weak point of partial differential equation (PDE)-based methods compared with particle methods[‡] for solving the Boltzmann equation—while endowing the method with considerable efficiency through high-order accuracy and a formulation that is compatible with variance reduction. Other advantages of the PDE-based

[‡]Variance-reduced particle formulations akin to DSMC have also been developed [11–15] although in some cases [11–13] they require particle cancelation which, as shown in [12], introduces a low-order velocity space discretization that increases the cost and reduces the accuracy of the method. The method by Homolle and Hadjiconstantinou [14, 15] does not require velocity space discretization and is a viable alternative to the method discussed here.

DOI: 10.1002/fld

approach include simplified application of some boundary conditions and the possibility of direct steady-state formulations. The latter can provide large computational savings in the collision-dominated regime, where the time to steady state is very long compared with the kinetic time step imposed by the advection operator in the unsteady formulation.

The value of the DG formulation for solving the Boltzmann equation has been previously recognized both in gas dynamics and in other fields, where Boltzmann-like transport equations need to be solved [16]. However, the computational cost associated with the evaluation of the weak form of the collision operator for dilute gases has limited previous work to cases where simplifying/modeling assumptions can be made. We refer here to the work of Dai and Yu [17] who applied the DG formulation to the BGK model of the Boltzmann equation, and the work of Cale and coworkers [18] who consider the linear Boltzmann equation[§] in conjunction with a relaxation-time approximation.

The paper is organized as follows: In the next section we provide a brief background on the Boltzmann equation. In Section 3, we give a brief overview of the DG formulation of the Boltzmann equation for completeness. We also discuss the Monte Carlo method used to evaluate the weak form of the collision integral, which enables the efficient use of the DG formulation for the Boltzmann equation. In Section 4 we present a variety of validation results for both time-dependent and steady problems in zero or one spatial dimension and three velocity dimensions. Although these test problems are performed in zero or one spatial dimensions, we note that the collision integral evaluation approach presented here can be directly applied to higher-dimensional problems without any modification, while extension of the RKDG treatment of the advection operator to higher dimensions is standard; in other words, we expect the present method to be directly extendable to flows in two and three spatial dimensions.

2. BACKGROUND

We consider a dilute monoatomic gas of molecular mass m at a reference temperature T_0 and reference number density n_0. The most probable molecular speed is given by $c_0 = \sqrt{2k_B T_0/m}$, where k_B is Boltzmann's constant. Let $f(\mathbf{r}, \mathbf{c}, t)$ be the velocity distribution function [2], where $\mathbf{r} = (x, y, z)$ is the position vector in physical space, $\mathbf{c} = (c_x, c_y, c_z)$ is the molecular velocity vector, $\mathbf{a} = (a_x, a_y, a_z)$ is the molecular acceleration resulting from body forces and t is the time.

In the remainder of the paper, all quantities will be nondimensionalized using the molecular mean free path[¶] λ_0, most probable molecular velocity c_0 and a collision time $\tau_0 \equiv \sqrt{\pi}\lambda_0/(2c_0)$. We assume that a well-defined [2] differential collision cross section σ exists such that the (dimensionless) Boltzmann equation can be expressed as

$$\frac{\partial f}{\partial t} + \frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial f}{\partial \mathbf{r}} + \mathbf{a}\cdot\frac{\partial f}{\partial \mathbf{c}} = \left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\mathrm{coll}} \tag{1}$$

[§]The linear Boltzmann equation [2] differs significantly from the linearized Boltzmann equation, and is valid for special applications that do not include the hydrodynamics of a dilute gas of interest here.

[¶]In the particular case of a hard-sphere gas of diameter d at number density n_0, $\lambda_0 = (\sqrt{2}\pi n_0 d^2)^{-1}$.

DOI: 10.1002/fld

with

$$\left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\mathrm{coll}}(\mathbf{r},\mathbf{c},t)=\frac{\sqrt{\pi}}{2}\int_{\mathbf{c}_1}\int_{\Theta}(f'f_1'-ff_1)g\sigma\,\mathrm{d}^2\Theta\,\mathrm{d}^3\mathbf{c}_1 \tag{2}$$

where $\mathbf{c}$ and $\mathbf{c}_1$ are the pre-collision velocities, $f_1=f(\mathbf{r},\mathbf{c}_1,t)$, $f'=f(\mathbf{r},\mathbf{c}',t)$, and $f_1'=f(\mathbf{r},\mathbf{c}_1',t)$; also, $g=|\mathbf{c}_1-\mathbf{c}|$ is the magnitude of the relative velocity vector, and $\mathbf{c}',\mathbf{c}_1'$ are the post-collision velocities, related to the precollision velocities through the scattering solid angle Θ. Here, and in the remainder of the paper, integration in velocity space extends from $-\infty$ to ∞ unless otherwise stated; similarly, the solid angle integration is over the surface of the unit sphere.

Hydrodynamic quantities are extracted from moments of the distribution function [1, 2, 19]. For example, the number density is given by

$$n=\int f\,\mathrm{d}^3\mathbf{c} \tag{3}$$

while the flow velocity $\mathbf{u}=(u_x,u_y,u_z)$ is given by

$$u_i=\frac{1}{n}\int c_i f\,\mathrm{d}^3\mathbf{c} \tag{4}$$

Finally, the components of the stress tensor are given by

$$P_{ij}=\int(c_i-u_i)(c_j-u_j)f\,\mathrm{d}^3\mathbf{c} \tag{5}$$

3. FORMULATION

Following the approach laid out in [7], we begin by introducing the deviational distribution function, f^d, defined as

$$f^d\equiv f-f^{\mathrm{MB}} \tag{6}$$

where f^{MB} is an *arbitrary* equilibrium (Maxwell–Boltzmann) distribution. For simplicity, in this paper we take f^{MB} to be only a function of velocity—that is f^{MB} is chosen to be independent of physical space and time, thus representing absolute equilibrium. Substituting this definition into the dimensionless Boltzmann equation, we obtain

$$\frac{\partial f^d}{\partial t}+\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial f^d}{\partial\mathbf{r}}+\mathbf{a}\cdot\frac{\partial f^d}{\partial\mathbf{c}}=\left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\mathrm{coll}}-\mathbf{a}\cdot\frac{\partial f^{\mathrm{MB}}}{\partial\mathbf{c}} \tag{7}$$

We would like to emphasize that the particular choice of f^{MB} may affect the efficiency (by affecting the magnitude of f^d and thus [7, 20] the resulting statistical uncertainty), but does not affect the accuracy or applicability of the method. Our particular choice here, primarily motivated by algorithmic convenience, means that the magnitude of the statistical uncertainty will vary in space and time as the shape and magnitude of f^d varies. Note, however, that in the low-signal problems considered here, the deviation from absolute equilibrium is small at any point in physical space and time. As our numerical results confirm, the resulting spatial variation of statistical uncertainty is small and thus, at present, does not justify the extra algorithmic complexity associated with selecting the locally optimal $f^{\mathrm{MB}}=f^{\mathrm{MB}}(\mathbf{r},\mathbf{c},t)$.

DOI: 10.1002/fld

3.1. Discontinuous Galerkin

To obtain the RKDG formulation of the above governing equation, we closely follow the approach laid out in [8]. We will suppose that our six-dimensional computational domain[‖] is divided into elements denoted by Ω. We require that for *any*[**] test function $v(\mathbf{r},\mathbf{c})$ defined to be zero outside of the element Ω

$$\int_\Omega v\left(\frac{\partial f^d}{\partial t}+\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial f^d}{\partial \mathbf{r}}+\mathbf{a}\cdot\frac{\partial f^d}{\partial \mathbf{c}}\right)\mathrm{d}^6\Omega=\int_\Omega v\left(\left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\mathrm{coll}}-\mathbf{a}\cdot\frac{\partial f^{\mathrm{MB}}}{\partial \mathbf{c}}\right)\mathrm{d}^6\Omega \tag{8}$$

where the integrals extend over the element Ω. We note that as v is defined to be zero outside of Ω, a similar equation also holds when the integrals extend over any part of the entire domain.

We proceed with integration by parts to obtain

$$\begin{aligned}&\int_\Omega v\frac{\partial f^d}{\partial t}\mathrm{d}^6\Omega+\int_\Gamma v f^d\left(\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\mathbf{n}_\mathbf{r}+\mathbf{a}\cdot\mathbf{n}_\mathbf{c}\right)\mathrm{d}^5\Gamma-\int_\Omega f^d\left(\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial v}{\partial \mathbf{r}}-\mathbf{a}\cdot\frac{\partial v}{\partial \mathbf{c}}\right)\mathrm{d}^6\Omega\\&\quad=\int_\Omega v\left(\left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\mathrm{coll}}-\mathbf{a}\cdot\frac{\partial f^{\mathrm{MB}}}{\partial \mathbf{c}}\right)\mathrm{d}^6\Omega\end{aligned} \tag{9}$$

Here, $\mathbf{n}_\mathbf{r}$ and $\mathbf{n}_\mathbf{c}$ are, respectively, outward-normal vectors of the element in physical and velocity space and $|\mathbf{n}_\mathbf{r}|^2+|\mathbf{n}_\mathbf{c}|^2=1$; Γ denotes the surface of the element Ω.

We then define h to be a flux function[††] that approximates $f^d\times[(\sqrt{\pi}/2)\mathbf{c}\cdot\mathbf{n}_\mathbf{r}+\mathbf{a}\cdot\mathbf{n}_\mathbf{c}]$. Additionally, in the present work, our interest lies in cases where body forces are negligible or not present; we will thus set $\mathbf{a}=0$ to obtain

$$\int_\Omega v\frac{\partial f^d}{\partial t}\mathrm{d}^6\Omega+\int_\Gamma vh\,\mathrm{d}^5\Gamma-\int_\Omega f^d\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial v}{\partial \mathbf{r}}\mathrm{d}^6\Omega=\int_\Omega v\left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\mathrm{coll}}\mathrm{d}^6\Omega \tag{10}$$

We expand f^d (specifically, f^d within a single element) and v in terms of our (now taken to be finite) set of basis functions $\phi_i(\mathbf{r},\mathbf{c})$.

$$\hat{f}^d=\sum_i \hat{f}_i^d(t)\phi_i(\mathbf{r},\mathbf{c}) \tag{11}$$

$$\hat{v}=\sum_j \hat{v}_j\phi_j(\mathbf{r},\mathbf{c}) \tag{12}$$

The functions ϕ_i are defined to be nonzero only within a single element. No continuity requirements exist between elements, allowing the formulation to capture discontinuities in the solution. Substituting these expressions into (10) and requiring the equation to hold for any set of coefficients $\hat{v}_j$ (i.e. any $\hat{v}$), we obtain

$$\sum_i\frac{\partial \hat{f}_i^d}{\partial t}\int_\Omega\phi_i\phi_j\,\mathrm{d}^6\Omega+\int_\Gamma\phi_j h\,\mathrm{d}^5\Gamma-\frac{\sqrt{\pi}}{2}\sum_i\hat{f}_i^d\int_\Omega\phi_i\mathbf{c}\cdot\frac{\partial\phi_j}{\partial\mathbf{r}}\mathrm{d}^6\Omega=\int_\Omega\phi_j\left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\mathrm{coll}}\mathrm{d}^6\Omega \tag{13}$$

[‖]In practice, the computational domain will be of dimension $D+3$, where D is the problem dimensionality in physical space.

[**]Provided integrability requirements are met, see [8].

[††]This will simply be an upwind flux as our convection term is linear.

DOI: 10.1002/fld

which must hold for all j. Equation (13) defines a system of ordinary equations for $\hat{f}_i^d$; integrating this in time using a (strong stability preserving [8]) Runge–Kutta method[‡‡] gives the time evolution of $\hat{f}_i^d$, and thus $\hat{f}^d$.

All integrals on the left side of (13) are evaluated using Gaussian quadrature using standard methods [10]. The shape functions used in this work are tensor products of Legendre polynomials. We will use the notation p_i to indicate that we use polynomials up to (and including) order i in this tensor product. The use of tensor product shape functions means that a sum-factorization technique [10] can be used to efficiently evaluate the sums arising from this integration.

3.2. Collision integral formulation

To evaluate the term involving the collision integral in Equation (13) we will use a Monte Carlo integration technique, which extends the variance reduction ideas proposed in [7] to the present DG formulation. Specifically, the variance reduction arises from considering only the deviational distribution function f^d (Equation (6)) and the use of importance sampling (see below).

Let us define

$$I(v) \equiv \int_\Omega v \left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\text{coll}} \mathrm{d}^6\Omega = \int_{\mathbf{r}}\int_{\mathbf{c}} v \left[\frac{\mathrm{d}f}{\mathrm{d}t}\right]_{\text{coll}} \mathrm{d}^3\mathbf{c}\,\mathrm{d}^3\mathbf{r} \tag{14}$$

The second equality follows by using the fact that v is taken to be zero outside the element in question and thus integration over both $\mathbf{r}$ and $\mathbf{c}$ can extend over all space. Using the properties of the collision operator [1, 2], and the fact that integration extends over all velocity space we can write

$$I(v) = \frac{\sqrt{\pi}}{4}\int_{\mathbf{r}}\int_{\mathbf{c}}\int_{\mathbf{c}_1}\int_\Theta (v_1' + v' - v_1 - v) f f_1 g\sigma \,\mathrm{d}^2\Theta\,\mathrm{d}^3\mathbf{c}_1\,\mathrm{d}^3\mathbf{c}\,\mathrm{d}^3\mathbf{r} \tag{15}$$

where $v_1 = v(\mathbf{r}, \mathbf{c}_1)$, $v' = v(\mathbf{r}, \mathbf{c}')$ and $v_1' = v(\mathbf{r}, \mathbf{c}_1')$. It is clear that the evaluation of the above 11-dimensional integral by direct quadrature is prohibitive, especially when one considers that $I(\phi_j)$ needs to be evaluated for every shape function ϕ_j. On the other hand, the Monte Carlo approach developed below makes this evaluation not only possible but also efficient when coupled with variance reduction and importance sampling.

Using expression (6), noting that the collision integral for a Maxwell–Boltzmann distribution is identically zero, and taking advantage of the symmetry in the collision operator [2], we can express

$$\begin{aligned} I(v) &= \frac{\sqrt{\pi}}{4}\int_{\mathbf{r}}\int_{\mathbf{c}}\int_{\mathbf{c}_1}\int_\Theta (v_1' + v' - v_1 - v)\,(2 f^d f_1^{\mathrm{MB}} + f^d f_1^d) g\sigma \,\mathrm{d}^2\Theta\,\mathrm{d}^3\mathbf{c}_1\,\mathrm{d}^3\mathbf{c}\,\mathrm{d}^3\mathbf{r} \\ &= \frac{\sqrt{\pi}}{2}\chi_{\mathrm{MB},d}\int_{\mathbf{r}}\int_{\mathbf{c}}\int_{\mathbf{c}_1}\int_\Theta (v_1' + v' - v_1 - v)\frac{f^d f_1^{\mathrm{MB}}}{\chi_{\mathrm{MB},d}} g\sigma \,\mathrm{d}^2\Theta\,\mathrm{d}^3\mathbf{c}_1\,\mathrm{d}^3\mathbf{c}\,\mathrm{d}^3\mathbf{r} \\ &\quad + \frac{\sqrt{\pi}}{4}\chi_{d,d}\int_{\mathbf{r}}\int_{\mathbf{c}}\int_{\mathbf{c}_1}\int_\Theta (v_1' + v' - v_1 - v)\frac{f^d f_1^d}{\chi_{d,d}} g\sigma \,\mathrm{d}^2\Theta\,\mathrm{d}^3\mathbf{c}_1\,\mathrm{d}^3\mathbf{c}\,\mathrm{d}^3\mathbf{r} \end{aligned} \tag{16}$$

[‡‡]All work presented in this paper uses the three-stage Runge–Kutta method given in [8].

DOI: 10.1002/fld

where we have defined

$$\chi_{\mathrm{MB},d} \equiv \int_{\mathbf{r}}\int_{\mathbf{c}}\int_{\mathbf{c}_1}\int_{\Theta} |f^d| f_1^{\mathrm{MB}}\, \mathrm{d}^2\Theta\, \mathrm{d}^3\mathbf{c}_1\, \mathrm{d}^3\mathbf{c}\, \mathrm{d}^3\mathbf{r} = 4\pi \left(\int f_1^{\mathrm{MB}}\, \mathrm{d}^3\mathbf{c}_1\right) \int_{\mathbf{r}}\int_{\mathbf{c}} |f^d|\, \mathrm{d}^3\mathbf{c}\, \mathrm{d}^3\mathbf{r} \tag{17}$$

$$\chi_{d,d} \equiv \int_{\mathbf{r}}\int_{\mathbf{c}}\int_{\mathbf{c}_1}\int_{\Theta} |f^d||f_1^d|\, \mathrm{d}^2\Theta\, \mathrm{d}^3\mathbf{c}_1\, \mathrm{d}^3\mathbf{c}\, \mathrm{d}^3\mathbf{r} = 4\pi \int_{\mathbf{r}} \left\{ \left(\int_{\mathbf{c}} |f^d|\, \mathrm{d}^3\mathbf{c}\right) \left(\int_{\mathbf{c}_1} |f_1^d|\, \mathrm{d}^3\mathbf{c}_1\right) \right\} \mathrm{d}^3\mathbf{r} \tag{18}$$

with 4π being the surface area of the unit sphere.

Noting that $f^d f_1^d/\chi_{d,d}$ and $f_1^{\mathrm{MB}} f^d/\chi_{\mathrm{MB},d}$ in Equation (16) are *normalized* probability distribution functions, we can perform *importance sampling* [20, 21], obtaining

$$\begin{aligned} I(v) \approx{}& \frac{\sqrt{\pi}}{2}\chi_{\mathrm{MB},d}\frac{1}{N_{\mathrm{MB}}}\sum_{i=1}^{N_{\mathrm{MB}}}[v'_{1,i}+v'_i-v_{1,i}-v_i]g_i\sigma_i\,\mathrm{sgn}(f_i^d) \\ &+\frac{\sqrt{\pi}}{4}\chi_{d,d}\frac{1}{N_d}\sum_{j=1}^{N_d}[v'_{1,j}+v'_j-v_{1,j}-v_j]g_j\sigma_j\,\mathrm{sgn}(f_j^d)\,\mathrm{sgn}(f_{1,j}^d) \end{aligned} \tag{19}$$

where the set of collision parameters $\{\mathbf{r}, \mathbf{c}_1, \mathbf{c}_2, \Theta\}$ are chosen with a probability $f_1^{\mathrm{MB}}|f^d|/\chi_{\mathrm{MB},d}$ in the first sum and $|f^d||f_1^d|/\chi_{d,d}$ in the second (this implies Θ is chosen with a uniform probability on the unit sphere in each case).

To implement (13), we must evaluate $I(\phi_j)$ for all shape functions in all elements. To do this more efficiently, we note that the cost of evaluating (19) (and thus the cost of this DG approach) can be reduced significantly because this equation can be evaluated for *all* shape functions in *all* elements using the *same* set of samples, namely the N_{MB} collision events in the first term and N_d collision events in the second term. This is achieved by updating the sum only for the cells containing pre- or post-collision velocities for each collision event. This is possible because the shape functions are zero outside their associated element, and thus only the elements containing a pre- or post-collision velocity for a given collision event will be affected by that collision event. When f^d is small, $\chi_{\mathrm{MB},d} \gg \chi_{d,d}$; in such cases, for improved computational efficiency, one could choose N_{MB} to be much larger than N_d.

We re-emphasize that the above derivation holds for an arbitrary f^{MB} ('underlying Maxwell–Boltzmann distribution'). The choice of f^{MB} does not affect the accuracy or applicability of the method (no approximation has been made), only its efficiency: in general, the smaller the magnitude of the resulting f^d, the greater the degree of variance reduction and the higher the efficiency.

We also note that (19) shows that the action of the collision operator is calculated by sampling a representative number of 'collision events', as in DSMC. One small difference is that, unlike DSMC where the collision cross section enters as a weighing factor for the collision probability, in our formulation the collision cross section value directly enters into the numerical value of $I(\phi_j)$. Modifying this formulation to mirror the DSMC collision process, i.e. by including $g\sigma$ in the processes of choosing the N_d and N_{MB} collision events (and thus removing $g_i\sigma_i$ and $g_j\sigma_j$ from the sums) is also possible. The similarity between the two approaches means that a number of collisional processes developed for DSMC (e.g. inelastic collisions for internal degrees of freedom), will likely be applicable to our formulation after a reasonable amount of modifications.

DOI: 10.1002/fld

3.3. Boundary conditions

In this formulation, boundary conditions are imposed by specifying the (upwind) numerical fluxes at the walls. In this work, we use diffuse wall boundary conditions [2, 19]; this model is by far the most widely used [3, 19], primarily because it appears to capture the behavior of engineering surfaces of practical interest quite well.

The diffuse wall boundary condition can be expressed as

$$f^{\mathrm{MB}} + f^{d} = n^{\mathrm{wall}} f^{\mathrm{wall}} \quad \text{for } \mathbf{c}\cdot\hat{n} > 0 \tag{20}$$

where f^{wall} is a (normalized) distribution at equilibrium with the wall, $\hat{n}$ is the unit normal pointing into the gas, and the constant n^{wall} is determined by the mass conservation requirement

$$\int_{\mathbf{c}\cdot\hat{n}<0} (\mathbf{c}\cdot\hat{n})(f^{\mathrm{MB}} + f^{d})\,\mathrm{d}^3\mathbf{c} = \int_{\mathbf{c}\cdot\hat{n}>0} (\mathbf{c}\cdot\hat{n})(n^{\mathrm{wall}} f^{\mathrm{wall}})\,\mathrm{d}^3\mathbf{c} \tag{21}$$

In our implementation, we took advantage of the fact that for the low-speed, isothermal flows presented here, $n^{\mathrm{wall}} = n_0$, and did not fully implement (21). For flows were $n^{\mathrm{wall}} \neq n_0$, implementation of (21) is straightforward.

Imposition of other typical boundary conditions is also straightforward. For boundary conditions that are more complex than diffuse or specular walls, application of boundary conditions is an area where PDE-based approaches arguably have an advantage over particle-based approaches; in both cases a fluxal quantity is required, however, in particle approaches, random *samples* from this distribution must be generated, while here only the numerical value of the flux at the Gaussian quadrature points is required.

3.4. Collision integral implementation details

The implementation of the advection terms follows standard approaches [8]. We will thus focus on the numerical evaluation of the collision integral using Equation (19).

As usual [22] the infinite velocity space is truncated to a finite volume; this truncation does not cause any problems [7, 22] in low-speed flows provided the volume is sufficiently large. In our numerical implementation we used a cut-off of 4, i.e. the non-dimensional molecular velocity ranges from -4 to 4 in each dimension of velocity space.

To simplify the implementation, we will assume that our mesh is a 'tensor product' of an (arbitrary) mesh in physical space and an (arbitrary) mesh in velocity space[§§]—in other words we assume that every element is part of a set of elements *having an identical extent in physical space* that, as a set, span our truncated velocity space. This simplifies the procedure for finding the element containing the post-collision velocities in Equation (19) as well as for picking the pre-collision velocities (because $\mathbf{r}$ is the same for the two pre- and two post-collision velocities).

We also note that Equation (16) holds when the integral is taken over an arbitrary region in *physical* space (although the integrals must still extend over all velocity space). In particular, this means that we can independently evaluate (19) for each set of elements sharing a common extent in physical space; this greatly simplifies the implementation of a parallel code.[¶¶]

[§§] In the authors' current implementation, a tensor product grid in all dimensions is used.

[¶¶] The trade-off in this implementation is a smaller degree of variance reduction in cases where the magnitude of f^{d} varies significantly in physical space.

DOI: 10.1002/fld

Implementing Equation (19) is straightforward, provided an efficient method for generating the collision parameters $\{\mathbf{r}, \mathbf{c}, \mathbf{c}_1, \Theta\}$ exists. The method used in the present work is outlined below. As mentioned above, for simplicity, this algorithm assumes that we are computing the collision integral for a set of cells with the *same* extent in physical space that, as a set, span velocity space. We also assume that f^{MB} is only a function of velocity.

We will use a combination of the alias method [23, 24], which is a method for generating a sample from a discrete distribution in *constant* time (with a linear setup time), and the acceptance–rejection method [20] to correctly generate points from our continuous probability distribution.[‖‖] We use the alias method to generate samples from a function that bounds our desired distribution from above, and then perform acceptance–rejection on these samples to obtain the correct distribution.

We begin with the distribution $f_1^{\mathrm{MB}}|f^d|/\chi_{\mathrm{MB},d}$.

1. We first generate an upper bound for $|f^d|$ in each element; let us denote this by $f_{\max,k}$ where k indexes the element. As we use tensor products of Legendre polynomials, each shape function has a maximum magnitude of 1; hence, $f^d_{\max,k} = \sum_i |(\hat{f}_i^d)_k|$ is an upper bound. (In general, a tighter upper bound would be preferable.)
2. Loop:
   (a) We then use the alias method [23] to randomly pick a cell with a probability proportional to $\mathscr{C}_k f_{\max,k}$ where $\mathscr{C}_k$ is the volume of the cell k in *velocity* space.
   (b) A random point in phase space $\{\mathbf{r}, \mathbf{c}\}$ within the cell k is chosen using a uniform probability distribution.
   (c) This point is either accepted with a probability $|f^d(\mathbf{r}, \mathbf{c})|/f^d_{\max,k}$ or rejected and a new cell is chosen and the process repeats.
3. Generate three Gaussian random numbers to find $\mathbf{c}_1$ [20].
4. Generate Θ on the unit sphere [25].

The distribution $|f_1^d||f^d|/\chi_{d,d}$ is slightly more complex because both f_1^d and f^d have a dependence on $\mathbf{r}$.

1. We use the same upper bound for $|f^d|$ in each element as in the previous algorithm.
2. Loop:
   (a) We then use the alias method to randomly pick two cells independently with a probability $\mathscr{C}_k f_{\max,k}$. Let us denote the indices of the two cells selected by k and ℓ.
   (b) The vector $\{\mathbf{r}, \mathbf{c}, \mathbf{c}_1\}$ is chosen from a uniform distribution such that $\mathbf{c}$ is in cell k, $\mathbf{c}_1$ is in cell ℓ and $\mathbf{r}$ is in both cells k and ℓ (recall that we have assumed that all cells have the same spatial extent).
   (c) This (entire vector) $\{\mathbf{r}, \mathbf{c}, \mathbf{c}_1\}$ is accepted with a probability $|f^d(\mathbf{r}, \mathbf{c})|/f^d_{\max,k} \times |f^d(\mathbf{r}, \mathbf{c}_1)|/f^d_{\max,\ell}$ or (the entire vector is) rejected and a new *pair* of cells is chosen and the process repeats.
3. Generate Θ on the unit sphere.

[‖‖] A more efficient algorithm for this process could lead to a significant improvement of the overall speed of the code.

DOI: 10.1002/fld

4. NUMERICAL RESULTS

In this section, we present a sample of the test cases that have been performed to verify the accuracy (and performance) of the present method. In this work, the test cases are performed in zero and one spatial dimensions; however, we note that all tests use the (full) three-dimensional velocity space, meaning that the collision integral formulation is directly applicable to the more general case. Extension of the DG formulation to higher-dimensional cases is standard. Additionally, modifying the collision integral formulation to other interaction models (for which the collision cross section σ can be computed, e.g. variable hard-sphere gas [4]) is straightforward.

4.1. Spatially homogeneous case

The convergence properties of the RKDG method have been studied extensively [8]. For this reason, we will focus on the convergence results for the collision integral formulation of Section 3.2 as this represents the primary difference between the present work and existing RKDG formulations.

We use the analytical solution [26] of the Boltzmann equation for spatially homogenous relaxation with Maxwellian molecules*** as a test case. Figure 1 shows the convergence for evaluating $[\mathrm{d}f/\mathrm{d}t]_{\mathrm{coll}}$ as a function of the mesh size for zeroth-order (p_0) and first-order (p_1) elements. The error reported in this case is the root-mean-square difference between the calculated and exact

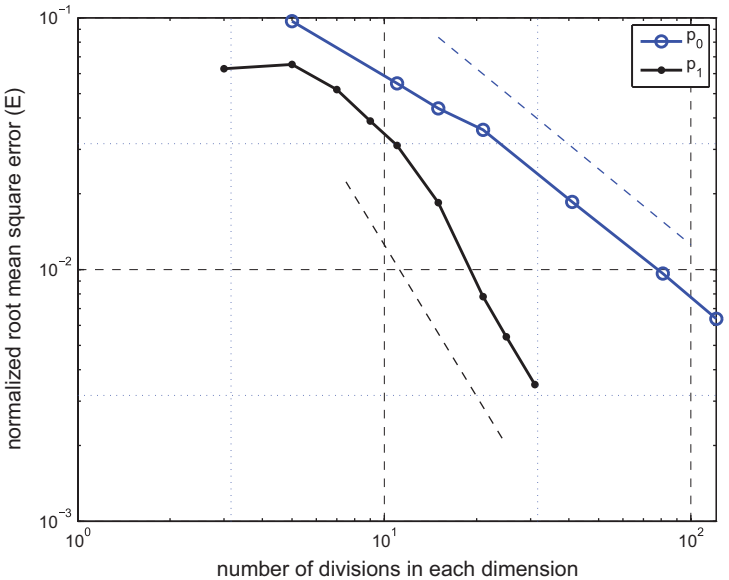

Figure 1. Error levels for collision integral as a function of number of elements used in each dimension. The dashed lines illustrate linear and quadratic convergence rates.

***Maxwellian molecules are defined such that $\sigma \propto 1/g$ (the constant of proportionality is immaterial for the present test case).

DOI: 10.1002/fld

value of the collision integral at a lattice of points in velocity space (approximating an L_2 error norm). This error is normalized by the exact value of the collision integral at $\mathbf{c}=0$.

As expected, with p_0 elements we observe a linear convergence rate, while the convergence rate is approximately quadratic with p_1 elements. This suggests that, when higher levels of accuracy are needed, it will be advantageous to use higher-order elements, even when taking into account the fact that higher-order elements have more degrees of freedom and a higher computational cost per element.

Although calculations using p_2 elements are certainly possible (as will be shown in the next section), we were unable to observe the expected asymptotic convergence rate for p_2 elements. This is primarily because the associated error levels are too low and decrease too quickly to accurately resolve with the Monte Carlo integration method used: for example, with only 11 elements in each dimension we obtain normalized error levels on the order of 5×10^{-3}; the error levels in the asymptotic regime will be *much* lower. Our inability to observe the asymptotic convergence rate does not imply that there is no advantage in using p_2 or higher-order elements; to the contrary, this low error level is precisely what one desires. However, the choice of the optimum polynomial order will also depend on statistical uncertainty considerations; this is discussed further in Section 4.3. Additionally, note that the difficulty in resolving exceptionally low-noise levels does not diminish the effectiveness of our Monte Carlo method for its intended purpose—obtaining low-noise solutions for low-signal flows. As will be shown later, our method is very effective in obtaining relative statistical uncertainties of significantly less than 1% for arbitrarily small deviations from equilibrium.

4.2. Flow in a channel

In this section we present comparisons of our results with DSMC solutions of transient shear and steady pressure-driven flows of a hard-sphere gas of molecular diameter d [dimensional collision cross section $d^2/4$, dimensional mean free path $\lambda_0=1/(\sqrt{2}\pi n_0 d^2)$]. The test cases presented here span the transition regime $0.1\lesssim Kn\equiv\lambda/L\lesssim10$ where solution of the Boltzmann equation is required.[†††]

In the shear flow, the system walls (at $x=-L/2$ and $x=L/2$) are impulsively accelerated at $t=0$ to a velocity of ±0.1 in the y-direction;[‡‡‡] a schematic illustrating the geometry is given in Figure 2. This flow represents a stringent test of the method, because it involves a discontinuity in the distribution function propagating into the computational domain.

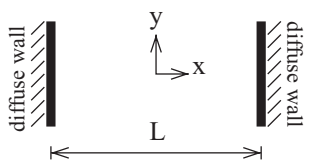

Figure 2. One-dimensional flow geometry.

[†††] The Navier–Stokes description (albeit subject to slip boundary conditions) is valid for $Kn\lesssim0.1$ while for $Kn\gtrsim10$, the effect of collisions is small and the collisionless Boltzmann equation may be used [2, 3].

[‡‡‡] This velocity was chosen to make comparison with DSMC possible; our method can solve this problem at arbitrary speeds for the same computational cost.

DOI: 10.1002/fld

All calculations presented in this section were performed using *nonuniform* elements; refinement in velocity space near $\mathbf{c}=0$ was used in all cases. For $Kn=0.1$, refinement in physical space near the walls was also applied. Additionally, due to the discontinuity propagating in the x-direction, in these transient flows a finer discretization is used in the c_x direction (compared with the c_y and

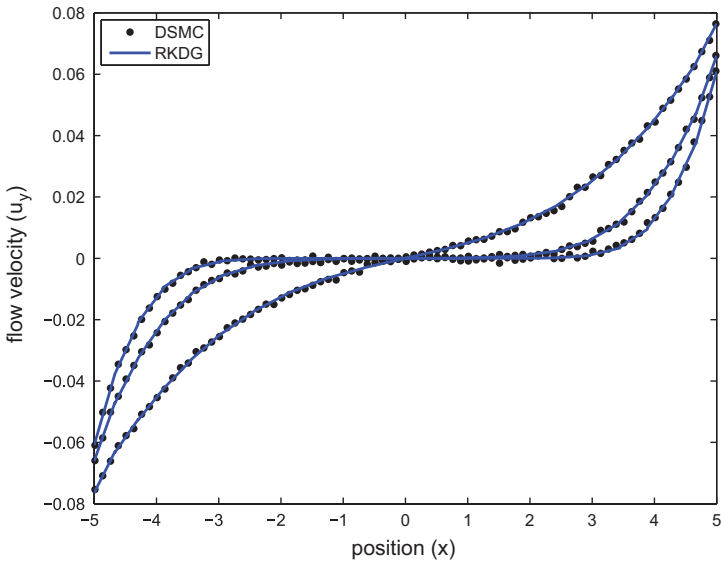

Figure 3. Velocity profiles for transient shear flow at $Kn=10^{-1}$ at times $t\approx 1.7, 2.8, 7.3$.

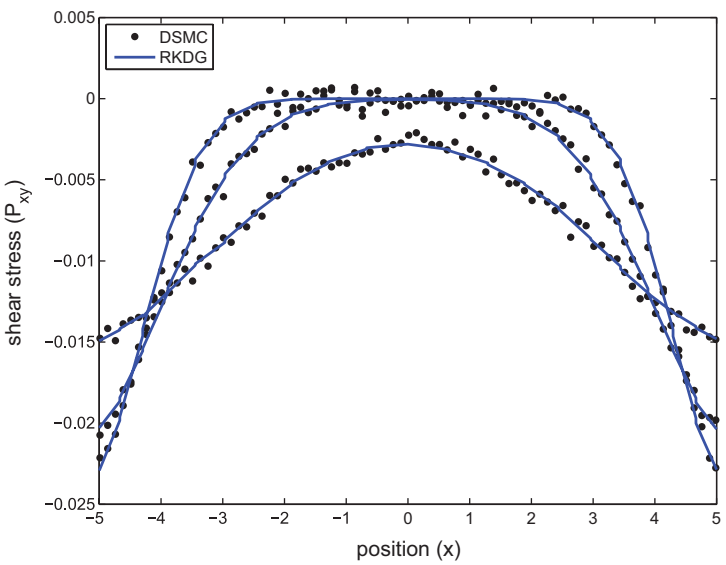

Figure 4. Shear stress profiles for transient shear flow at $Kn=10^{-1}$ at times $t\approx 1.7, 2.8, 7.3$.

DOI: 10.1002/fld

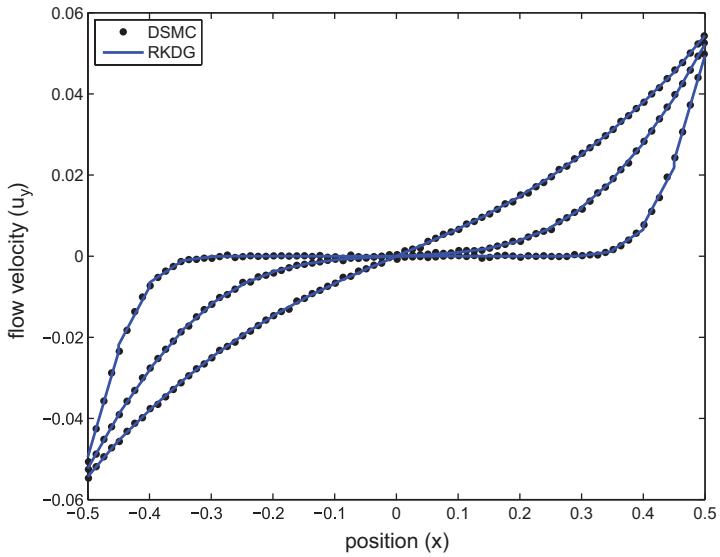

Figure 5. Velocity profiles for transient shear flow at $Kn = 1$ at times $t \approx 0.1, 0.3, 0.5$.

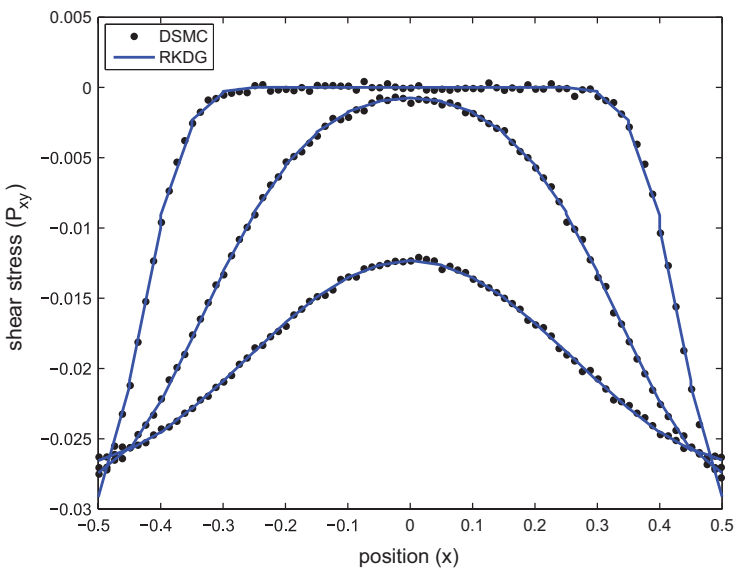

Figure 6. Shear stress profiles for transient shear flow at $Kn = 1$ at times $t \approx 0.1, 0.3, 0.5$.

c_z directions). These mesh refinements significantly improved the quality of the results, although no attempt was made to optimize the mesh used.

Figures 3–8 show the velocity and shear stress profiles of transient shear flows at various times for various Knudsen numbers (Kn). The discretization uses $20\,p_1$ elements in each dimension,

DOI: 10.1002/fld

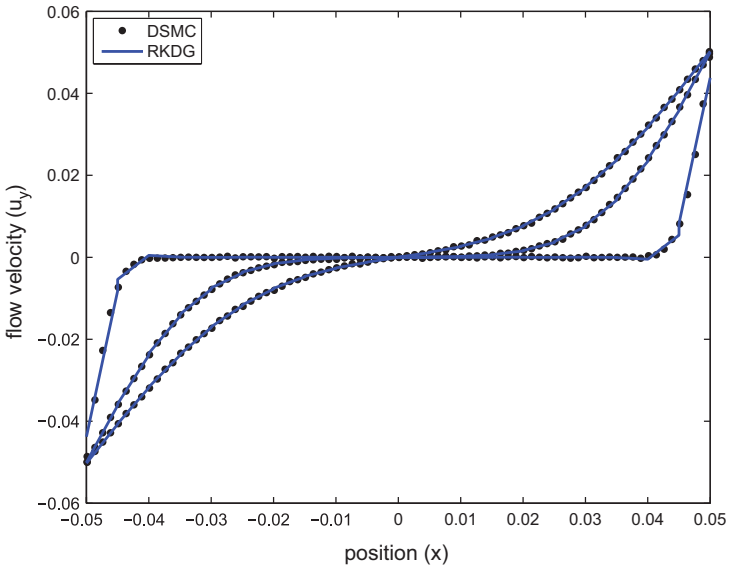

Figure 7. Velocity profiles for transient shear flow at $Kn=10$ at times $t\approx 0.006, 0.02, 0.03$.

except for c_x which uses 40 p_1 elements. The results are compared against fine-resolution DSMC calculations, and show an excellent level of agreement.[§§§] Figures 9 and 10 show velocity and shear stress profiles for the $Kn=1$ case using p_2 elements. This discretization uses 10 elements in each dimension, (20 for c_x), and in overall uses $\frac{1}{16}$ as many elements to obtain a similar degree of accuracy as in the p_1 case.[¶¶¶]

Figure 11 shows the steady-state results for pressure-driven flow (using the linearized approach of [27]) for a Knudsen number of $Kn=2/\sqrt{\pi}$ compared with the previously published results of [22]. A total of 24^4 nonuniform p_1 elements were used.

4.3. Effect of number of Monte Carlo samples

The use of a Monte Carlo integration method means that there will be some level of statistical uncertainty inherent in our evaluation of the collision integral. To find the effect of this uncertainty on the solution we have measured the effect of the number of Monte Carlo samples[‖‖‖] used on the flow velocity and the shear stress uncertainty. To quantify the effect of noise on a hydrodynamic

[§§§] To ensure accurate results for comparison, the DSMC runs used 400 cells in physical space and a time step such that a particle with normal velocity c_0 would take 10 time steps to cross a cell. While we used a highly refined DSMC calculation to ensure that our results are correct, we base our performance comparison on a coarser discretization for DSMC.

[¶¶¶] Note that a tensor product p_2 element has 3^4 shape functions per element compared with 2^4 for a p_1 element; this means that in this example the p_2 case has about $\frac{1}{3}$ the number of degrees of freedom of the p_1 case.

[‖‖‖] Here, the number of samples refers to the total number of updates done in Equation (19) (per Runge–Kutta step). When using order p elements, this will be $4(N_{\mathrm{MB}}+N_d)(p+1)^4$; there are 4 pre- or post-collision velocities for each of the $(N_{\mathrm{MB}}+N_d)$ collision events, and the sum in (19) is updated for each of the $(p+1)^4$ shape functions in the corresponding elements.

DOI: 10.1002/fld

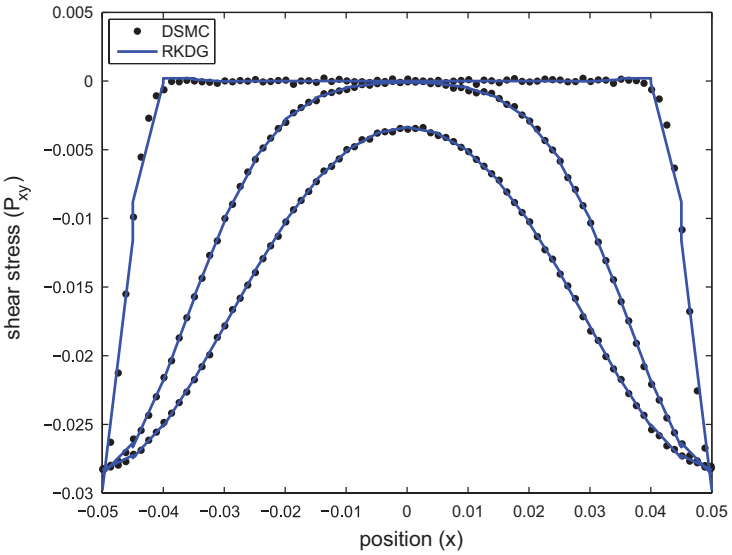

Figure 8. Shear stress profiles for transient shear flow at $Kn = 10$ at times $t \approx 0.006, 0.02, 0.03$.

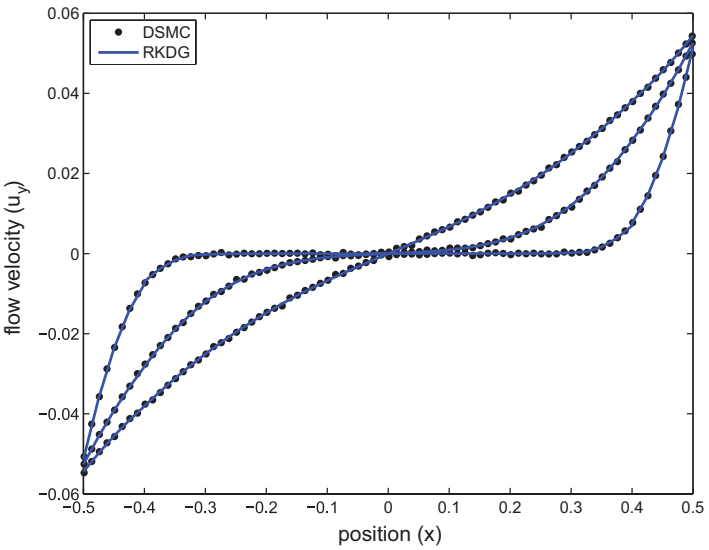

Figure 9. Velocity profiles for transient shear flow at $Kn = 1$ using p_2 elements at times $t \approx 0.1, 0.3, 0.5$.

quantity Q we use the relative statistical uncertainty $E_Q = \sigma_Q / \bar{Q}$, where σ_Q is the standard deviation characterizing the statistical distribution of Q, and $\bar{Q}$ is a characteristic value. (In the following figures, an average of E_Q over the domain is typically given.)

DOI: 10.1002/fld

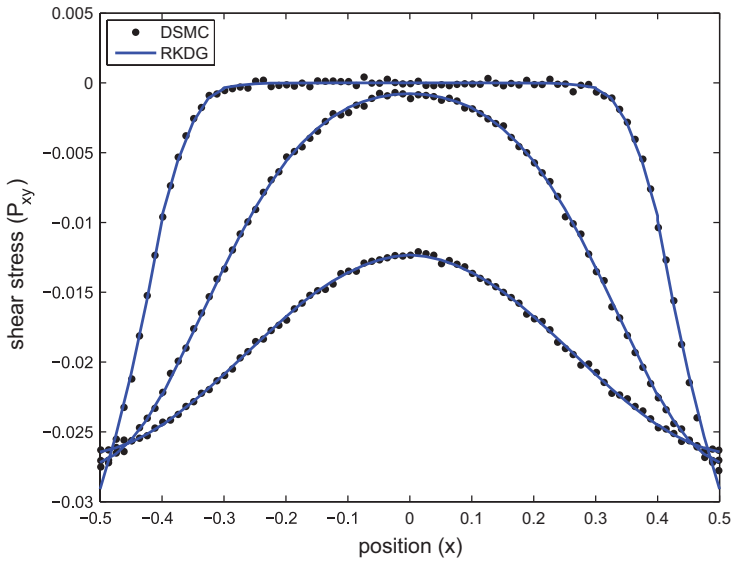

Figure 10. Shear stress profiles for transient shear flow at $Kn = 1$ using p_2 elements at times $t \approx 0.1, 0.3, 0.5$.

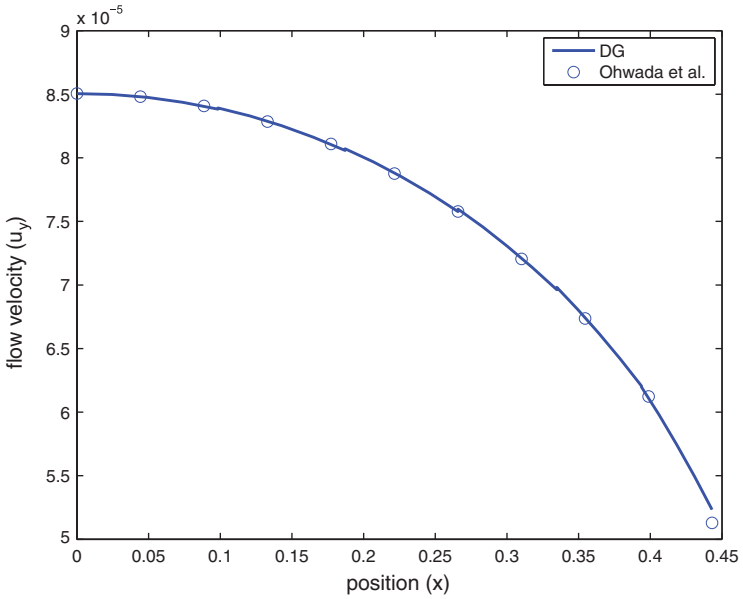

Figure 11. Pressure-driven flow at $Kn = 2/\sqrt{\pi} \approx 1.1$.

DOI: 10.1002/fld

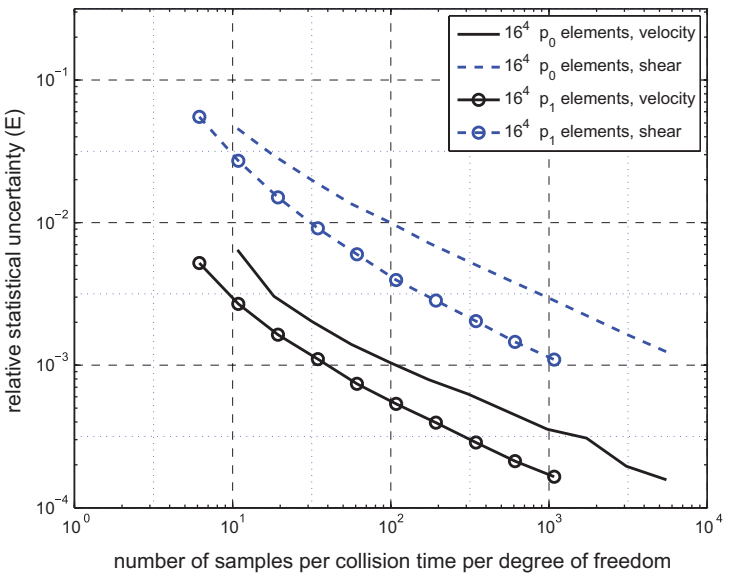

Figure 12. Scaling of relative statistical uncertainty in the flow velocity and shear stress with number of Monte Carlo samples per collision time per degree of freedom. Results obtained using $16^4 p_0$ and p_1 elements.

More specifically, we perform a steady-state shear flow calculation (using a Chapman–Enskog distribution for boundary conditions to minimize edge effects) and measure the mean standard deviation of the flow velocity and shear stress over a set of nodes in physical space.**** These values are normalized by the boundary velocity and the theoretical value for the shear stress, respectively, and will be referred to as the relative statistical uncertainties. All tests in this section are performed for $Kn = 0.1$; for simplicity, these tests use a uniform mesh.

Figure 12 shows the dependence of the velocity and shear stress relative statistical uncertainties on the number of Monte Carlo samples per collision time†††† per degree of freedom for p_0 and p_1 elements using the *same*‡‡‡‡ mesh. The asymptotic convergence rate appears to be approaching $N^{-1/2}$, which is expected of Monte Carlo integration. In other words, the statistical uncertainty in the collision integral evaluation seems to have an (asymptotically) proportional effect on the statistical uncertainty of the hydrodynamic quantities of interest. When the number of samples is small, this trend no longer holds; in fact the method becomes unstable if too few samples are used.

Also, note that in both cases, higher-order methods require fewer samples per degree of freedom to obtain a given uncertainty level (although higher-order methods require more total samples

****Note that the nodes in the middle of the domain will typically have a smaller variance in these quantities; this is because f^d is smaller there as the mean flow velocity is closer to zero; hence, our choice for f^{MB} is a better approximation to the distribution function. This could be alleviated, although at the cost of a slightly more involved implementation than that discussed in Section 3.4.

††††The statistical uncertainty is not primarily affected by the number of samples per time step, but rather than the number of samples per collision time.

‡‡‡‡As opposed to, for example, meshes that would lead to the same degree of discretization error.

DOI: 10.1002/fld

for a given degree of uncertainty). In cases where extremely low levels of statistical uncertainty are required, this consideration might affect the choice of polynomial order and discretization used.

We can also see that the relative statistical uncertainty in the shear stress is typically an order of magnitude larger than the uncertainty in the velocity. Both, however, are very small; typical values are of $O(10^{-3})$ and $O(10^{-2})$, respectively, which as shown in the previous comparisons of Figures 3–11, are essentially imperceptible.

Figure 13 shows how the number of elements used affects that level of relative statistical uncertainty for p_1 elements. We see that, when using more elements, fewer samples are needed per degree of freedom to obtain a fixed degree of relative statistical uncertainty (although the total number of samples needed is larger when more elements are used).

We have not yet touched on perhaps the most important aspect of our method for evaluating the collision integral: its performance for low-speed flows. Figure 14 illustrates how the degree of relative statistical uncertainty is affected by the characteristic flow velocity (i.e. deviation from equilibrium). Two wall velocities (± 0.1 and ± 0.01) are shown; the figure shows that the relative statistical uncertainty does not change significantly between these two cases. This is in sharp contrast to DSMC, for which lower-flow velocities are associated with much higher relative statistical uncertainties [6]—for the same number of samples the level of relative statistical uncertainty in a DSMC calculation would have increased by a factor of 10, while the number of samples required to bring the relative statistical uncertainty to the same level would have increased by a factor of 100.

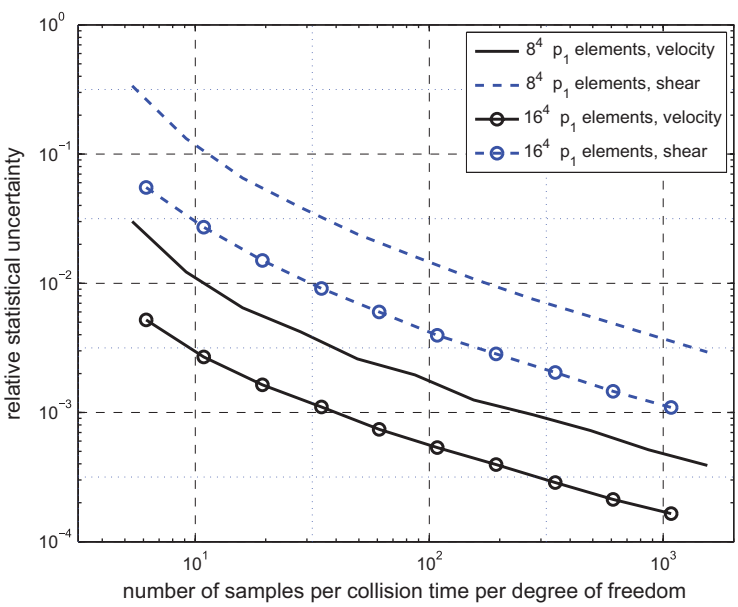

Figure 13. Scaling of relative statistical uncertainty in flow velocity and shear stress fluctuations with number of Monte Carlo samples per collision time per degree of freedom. Results use 8^4 and $16^4 p_1$ elements.

DOI: 10.1002/fld

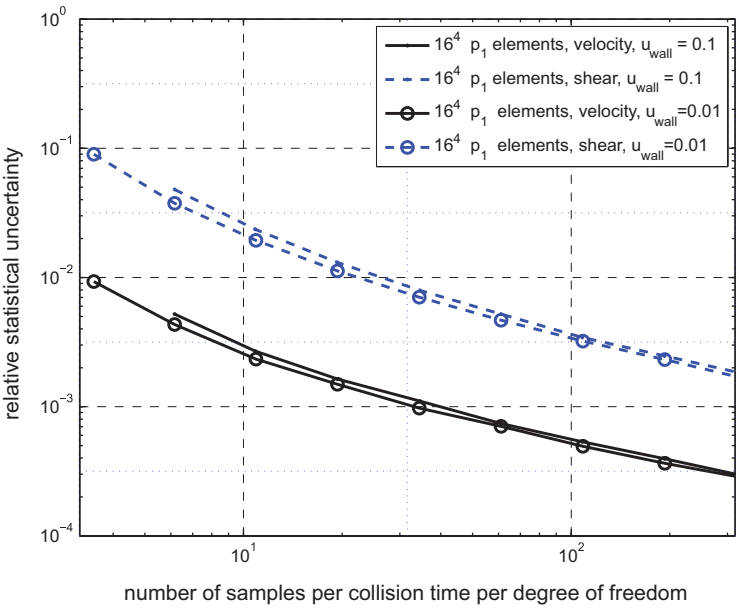

Figure 14. Effect of varying wall velocities on relative statistical uncertainty for the present method.

The superior performance scaling (with decreasing velocity) of the present method means that, below a certain characteristic velocity, the present method will be more efficient than DSMC. Precisely quantifying this crossover point is difficult, and will be implementation and problem dependent. However, work done to date suggests that this crossover point occurs at a wall velocity on the order of 0.1. More quantitatively, in our present implementation,[§§§§] time integrating a tensor product p_1 calculation with 40×20^3 elements at $Kn = 0.1$ for 7.5 collision times (i.e. the calculations shown in Figures 3 and 4) takes approximately 300 MB of storage and about 9 h of CPU time on a (single) 2000 MHz AMD Athlon 64 to obtain essentially 'noise-free' results; performing a similar calculation (same statistical uncertainty) using DSMC would take weeks. We further note that calculations at larger Knudsen numbers typically take less computational time due to the diminished effect of collisions.

Our preliminary results show that the effect of the number of Monte Carlo samples on the *average* value of the local flow velocity and shear stress is small—on the order of 1% for the shear stress over the range shown in Figure 12 and essentially negligible for more than 50 samples per collision time per degree of freedom. The effect on the flow velocity is even smaller.

Finally, we note that in the present method, only a relatively small number of samples are necessary per collision time per degree of freedom to obtain small relative statistical uncertainties.

[§§§§]The present code was expressed in C++, making use of the Lapack++ and Boost libraries. A more efficient implementation, perhaps utilizing existing finite element libraries, is certainly possible; the performance numbers given here are intended to give the reader a general sense of the computational cost associated with this method.

DOI: 10.1002/fld

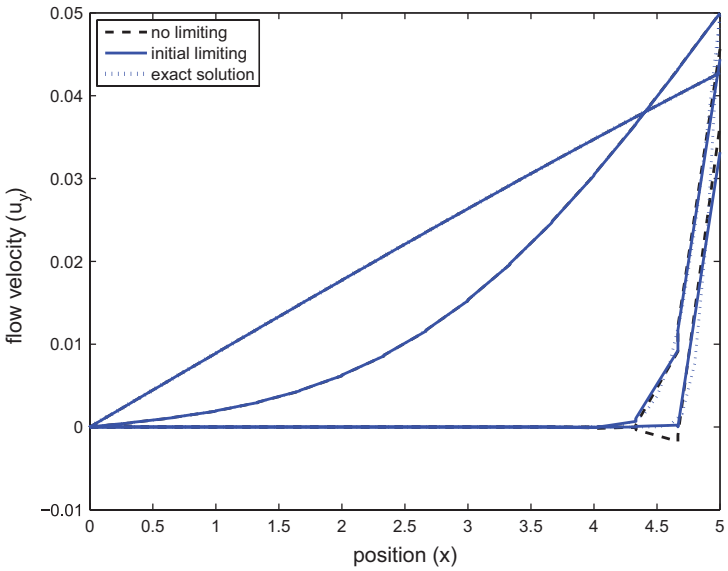

Figure 15. Effect of limiting on velocity profile for collisionless shear flow.

As the time step is typically significantly less than a collision time, even fewer samples are needed per time step; this makes the present method for evaluating the collision integral extremely computationally efficient.

4.4. Limiting

In general, RKDG methods require the use of a numerical limiter to ensure stability in the presence of discontinuities [8], although we have not found this necessary for the (linear) flux function of the Boltzmann equation. However, the lack of a flux limiter leads to a non-physical overshoot in the flow velocity and other hydrodynamic quantities at very short timescales due to the propagating discontinuity created by the impulsive wall acceleration in our shear flow problem. An example is shown in Figure 15; in this case collisionless shear flow was simulated both without a flux limiter and with the flux limiter described in [8] (the flux limiter was only used for the initial 10 time steps, during which time the propagating discontinuity has the sharpest effect on the velocity profile).

Without the flux limiter, there is a non-physical overshoot apparent at short times that disappears at longer timescales. Introducing a flux limiter removes this overshoot. However, we observe that, except at very short timescales, we closely match the analytic results either with or without the flux limiter (to the point where the three different solutions plotted are essentially indistinguishable). Discerning the full effect of slope limiting on the present method of solving the Boltzmann equation will require further research, although our initial exploration suggests that, while the use of a slope limiter is not necessary, in certain cases it can give qualitatively better results.

DOI: 10.1002/fld

5. CONCLUSION

We have presented a high-order method for solving the nonlinear Boltzmann equation that has a computational efficiency advantage over traditional Monte Carlo methods (such as DSMC) for low-signal flows.

The most important ingredient of the present work is a variance-reduced Monte Carlo method for evaluating moments of the collision integral, which enables the solution of the Boltzmann equation using an RKDG formulation. The resulting scheme exhibits low relative statistical uncertainty that is independent of the magnitude of the deviation from equilibrium, enabling *essentially noise-free* calculations at arbitrarily low-flow speeds or, more generally, arbitrarily low signal.

The method was validated for spatially homogeneous and one-dimensional (in space) problems. Our results indicate that engineering accuracy can be obtained using relatively coarse grids.

This last feature partially alleviates what is, perhaps, the most important disadvantage of the current method, namely the fact that the number of elements required scales as M^{D+3}, where M is the number of elements in one dimension. This scaling means that the current formulation and implementation (without the benefit of decades of development) can achieve essentially noise-free solutions to two-dimensional ($D=2$), low-signal problems using current personal computers or small clusters [28]; three-dimensional problems will be more costly but feasible. This can be put in context by noting that *for essentially noise-free solutions of low-signal problems* DSMC requires massively parallel computing resources even for $D=2$ [29]. At the same time, we expect DSMC to hold an advantage over the current formulation in high-signal flows.

As noted in the Introduction section, the present formulation offers the possibility of steady-state formulations (rather than explicit integration to steady state), which have the potential to provide significant computational savings for problems that evolve at timescales that are much longer than the collision time, and make this method even more competitive compared with time-explicit particle methods.

Extensions to multispecies and chemically reacting flows will be considered in the future.

ACKNOWLEDGEMENTS

This work was supported in part by Sandia National Laboratory. The authors are grateful to Dr. M. A. Gallis for valuable comments. Additionally, the authors are indebted to Prof. J. Peraire and Prof. A. T. Patera for help with the discontinuous Galerkin formulation.

REFERENCES

1. Vincenti WG, Kruger CH. *Introduction to Physical Gas Dynamics*. Krieger: Florida, 1965.
2. Cercignani C. *The Boltzmann Equation and its Applications*. Springer: New York, 1988.
3. Hadjiconstantinou NG. The limits of Navier–Stokes theory and kinetic extensions for describing small-scale gaseous hydrodynamics. *Physics of Fluids* 2006; **18**:1–19.
4. Bird GA. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Clarendon Press: Oxford, 1994.
5. Cercignani C. *Slow Rarefied Flows*: *Theory and Application to Micro-electro-mechanical Systems*, Birkhauser Verlag: Basel, 2006.
6. Hadjiconstantinou NG, Garcia AL, Bazant MZ, He G. Statistical error in particle simulations of hydrodynamic phenomena. *Journal of Computational Physics* 2003; **187**:274–297.
7. Baker LL, Hadjiconstantinou NG. Variance reduction for Monte Carlo solutions of the Boltzmann equation. *Physics of Fluids* 2005; **17**:1–4.
8. Cockburn B, Shu C. Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing* 2001; **16**:173–261.

DOI: 10.1002/fld

9. Cockburn B. *An Introduction to the Discontinuous Galerkin Methods for Convection-dominated Problems*. Lecture Notes. Available at: http://www.math.umn.edu/∼cockburn/LectureNotes.html, accessed on 21 June 2007.
10. Karniadakis GE, Sherwin SJ. *Spectral/hp Element Methods for CFD*. Oxford University Press: Oxford, 1999.
11. Baker LL, Hadjiconstantinou NG. Variance reduction in particle methods for solving the Boltzmann equation. *Proceedings of the Fourth International Conference on Nanochannels*, *Microchannels and Minichannels*, Limerick, Ireland, Paper ICNMM2006-96089, 2006.
12. Baker LL, Hadjiconstantinou NG. Variance-reduced particle methods for solving the Boltzmann equation. *Journal of Computational and Theoretical Nanoscience* 2008. DOI: 10.1166/jctn.2008.004.
13. Chun J, Koch DL. A direct simulation Monte Carlo method for rarefied gas flows in the limit of small Mach number. *Physics of Fluids* 2005; **17**:1–14.
14. Homolle TMM, Hadjiconstantinou NG. LVDSMC: low-variance deviational simulation Monte Carlo. *Physics of Fluids* 2007; **19**:1–4.
15. Homolle TMM, Hadjiconstantinou NG. A low-variance deviational simulation Monte Carlo for the Boltzmann equation. *Journal of Computational Physics* 2007; **226**:2341–2358.
16. Carrilo JA, Gamba IM, Majorana A, Shu C-W. A WENO-solver for the transients of Boltzmann–Poisson system for semiconductor devices: performance and comparisons with Monte Carlo methods. *Journal of Computational Physics* 2003; **184**:498–525.
17. Dai Q, Yu X. RKDG finite element method combined with BGK scheme for solving fluid dynamics system. *SIAM Journal on Scientific Computing* 2006; **28**:805–831.
18. Gobbert MK, Webster SG, Cale TS. A Galerkin method for the simulation of the transient 2-D/2-D and 3-D/3-D linear Boltzmann equation. *Journal of Scientific Computing* 2006; DOI: 10.1007/s10915-005-9069-1.
19. Sone Y. *Kinetic Theory and Fluid Dynamics*. Birkhauser: Basel, 2002.
20. Press WH, Teukolsky SAS, Vetterling WT, Flannery BP. *Numerical Recipes in C* (2nd edn). Cambridge University Press: Cambridge, 1992.
21. Tan Z, Varghese PL. The $\Delta-\varepsilon$ method for the Boltzmann equation. *Journal of Computational Physics* 1994; **110**:327–340.
22. Ohwada T, Sone Y, Aoki K. Numerical analysis of the Poiseuille and thermal transpiration flows between two parallel plates on the basis of the Boltzmann equation for hard-sphere molecules. *Physics of Fluids A* 1989; **1**:2042–2049.
23. Bratley P, Fox BL, Schrage LE. *A Guide to Simulation* (2nd edn). Springer: New York, 1987.
24. Kronmal RA, Peterson AV. The alias and alias-rejection-mixture methods for generating random variables from probability distributions. *IEEE Winter Simulation Conference*, San Diego, CA, 1979.
25. Alexander FJ, Garcia AL. The direct simulation Monte Carlo method. *Computers in Physics* 1997; **11**:588–593.
26. Krook M, Wu TT. Exact solutions of the Boltzmann equation. *Physics of Fluids* 1977; **20**:1589–1595.
27. Cercignani C, Daneri A. Flow of a rarefied gas between two parallel plates. *Journal of Applied Physics* 1963; **34**:3509–3513.
28. Lian ZY. Discontinuous Galerkin solution of the Boltzmann equation in multiple spatial dimensions. *M.S. Thesis*, Computation for Design and Optimization, Massachusetts Institute of Technology, 2007.
29. Gallis MA, Torczynski JR. An improved Reynolds-equation model for gas damping of microbeam motion. *Journal of Microelectromechanical Systems* 2004; **13**:653–669.