# Implicit Discontinuous Galerkin Method for the Boltzmann Equation

Wei Su[1] · Peng Wang[1] · Yonghao Zhang[1] · Lei Wu[2]

## Abstract

An implicit high-order discontinuous Galerkin (DG) method is developed to find the steady-state solution of rarefied gas flow described by the Boltzmann equation with full collision operator. In the physical space, the velocity distribution function is approximated by the piecewise polynomial of degree up to 4, while in the velocity space the fast spectral method is incorporated into the DG discretization to evaluate the collision operator. A specific polynomial approximation for the collision operator is proposed to reduce the computational cost of the fast spectral method by $K$ times, where for two-dimensional problem $K$ is 15 when the DG with 4th-order polynomial is used on triangular mesh. Based on the first-order upwind scheme, a sweeping technique is employed to solve the local linear equations resulting from the DG discretization sequentially over spatial elements. This technique can preserve stability of the scheme and requires no nonlinear limiter in solving hypersonic rarefied gas flows when the shock wave structure is fully resolved by fine spatial grid. Moreover, without assembling large sparse linear system, the computational cost in terms of memory consumption and CPU time is significantly reduced. Five different one/two-dimensional tests including low-speed microscale flows and hypersonic rarefied gas flows are used to assess the accuracy and efficiency of proposed approach. Our results show that, DG scheme of different order of approximating polynomial requires the same number of iterative steps to obtain the steady-state solution with the same order of accuracy; and the higher order the scheme is, the fewer spatial elements are needed, thus leading to less CPU time. Besides, our method can be faster than the finite difference solver by about one order of magnitude. The produced solutions can be used as benchmark data for assessing the accuracy of other gas kinetic solvers for the Boltzmann equation and gas kinetic models that simplify the Boltzmann collision operator.

✉ Lei Wu
wul@sustech.edu.cn

[1] James Weir Fluids Laboratory, Department of Mechanical and Aerospace Engineering, University of Strathclyde, Glasgow G1 1XJ, UK

[2] Department of Mechanics and Aerospace Engineering, Southern University of Science and Technology, Shenzhen 518055, China

## 1 Introduction

In gas kinetic theory, the dynamics of dilute gas is described by the one-particle velocity distribution function and the macroscopic flow properties are derived from corresponding velocity moments [13]. This theory has been used to describe transport phenomena in a wide range of scientific disciplines and applications such as the aerothermal dynamics in aerospace engineering, fusion processes in nuclear science, natural gas extraction in unconventional reservoirs, freeze drying techniques in pharmaceutical and food manufacture, electron transport in semiconductor devices, and physics of diffuse matter in interstellar medium, just to name a few. In Boltzmann's description, the variation of velocity distribution function comes from the linear streaming in the phase space, and the nonlinear interaction due to binary collisions. Thus, the velocity distribution function is a seven dimensional variable, with three in the physical space, three in the molecular velocity space, and one in time. Meanwhile, the nonlinear collision operator is a fivefold operator with three dimensions in the velocity space and two dimensions in a unit sphere (i.e. solid angle).

The multi-dimensional structure of the Boltzmann equation poses a real challenge to its numerical solution [24]. Historically, two major categories of approaches have been developed. One is the stochastic approach, which uses simulation particles to represent a large number of real molecules and mimic the molecular behaviors. The prevailing one is the direct simulation Monte Carlo (DSMC) method [6]. During the simulation, particles move through the spatial space in a realistic manner with respect to the time, while intermolecular collisions and molecule-surface interactions are calculated in probabilistic manner. The other category is the deterministic approach, which adopts a numerical quadrature to approximate the integration with respect to the molecular velocity on a set of fixed discrete points [3,33]. As a result of discretization in the velocity space, the original kinetic equation is represented as a set of linear hyperbolic equations with nonlinear source terms that couple all the equations. To solve the resulting system, schemes from the traditional computational fluid dynamics (CFD) for hyperbolic conservation law can be applied straightforwardly for the streaming term. Some hybrid stochastic-deterministic approaches have also been proposed to solve the Boltzmann equation [4,11,67].

In this paper, we focus on the deterministic method to solve the Boltzmann equation, which requires proper treatment of the linear streaming and nonlinear collision operator. The finite difference method (FDM), finite volume method and finite element method have been successfully employed to approximate spatial derivatives [27,37,38,43,65,76]. The advantage of these methods is that they have been well developed to achieve high order spatial and temporal accuracy. However, they might lose robustness and produce nonphysical solution, when the velocity distribution function has large variation and/or the kinetic equation becomes stiff [24]. Another category of schemes is the semi-Lagrangian [19,20,31] and Lagrangian methods [21,22], which are designed to ensure positivity of solutions. The basic idea of these schemes is to solve the streaming for successive time steps by following the characteristics, i.e. molecular trajectories. The semi-Lagrangian methods still utilize fixed computational grid, but evaluate solution at the points that can be transported by the molecular velocity onto the computational grid within a time step. The Lagrangian methods update solution according to streaming without using a spatial mesh. Instead, the calculation reduces to a single manipulation for each discrete velocity.

To simulate the collision, the most simple and widely used way is to replace the complicated collision operator by a relatively simple kinetic model, e.g. the Bhatnagar–Gross–Krook (BGK) model [5], ellipsoidal statistical BGK model [32], and Shakhov model [55], which describes the relaxation of velocity distribution function to the local equilibrium distributions

determined by macroscopic flow properties. Otherwise, the full Boltzmann collision operator should be calculated. Attempts to directly solve the full Boltzmann collision operator started from the late 1980s. Goldstein et al. constructed a discrete collision mechanics on the velocity nodes, which can preserve the main physical properties of the collision operator [28]. However, a large amount of discrete velocities are required, since post-collision velocities must fall on the grid points. The computational cost is $O\left(\bar{N}^7\right)$ ($\bar{N}$ is the number of points in each velocity direction), and the nominal accuracy is less than first order of the step size in the velocity space [8]. Improvement by using an interpolation to map the post-collision velocities onto the velocity grid makes the performance of the scheme comparable to or even faster than DSMC in simulating normal shock wave structure [44]. The projection method has also been proposed to evaluate the Boltzmann collision operator over a set of "on-lattice" collision pairs with different velocities, aim distances and reflect angles [63,64]. The kinetic theory group in Kyoto introduced another family of method evaluating collision in the velocity space [39,48,49,57], in which the velocity distribution function is expanded in terms of basis functions. The numerical kernels are pre-computed by numerical integration, which are restricted to the hard-sphere model and the distribution function with cylindrical symmetry. Note that there are other schemes such as the method based on nodal-discontinuous Galerkin discretization of the collision operator and a bi-linear convolution of the Galerkin projection [1].

Instead of directly calculating the collision integral on discrete velocities, there is another route to approximate collision in the frequency domain using Fourier transform technique. These methods can not only possess spectral accuracy, but also reduce computational cost through fast spectral algorithm. The pioneering work was introduced by Bobylev for Maxwell molecules [7]. Then, several spectral methods were developed, which have computational cost at the order of $\bar{N}^6$ [9,50]; the computational cost can be reduced to $O\left(\bar{N}^3\log\bar{N}\right)$ for distribution functions possessing cylindrical symmetry, when the fast Fourier transform (FFT) and Hankel transform are employed [68]. However, the accuracy is only $O\left(\bar{N}^{-1/2}\right)$. Based on the Carleman representation, an algorithm was developed for hard-sphere molecules to achieve accuracy of $O\left(\bar{N}^{-2}\right)$, where the integration over the unit sphere is separated from the one over the velocity space [10]. By employing the generalized Radon and X-ray transform, its computational cost is of $O\left(\bar{N}^6\log\bar{N}\right)$. The algorithm for variable hard-sphere molecules of accuracy $O\left(\bar{N}^{-2}\right)$ was also proposed with complexity of $O\left(\bar{N}^6\right)$ [34]. The fast spectral method (FSM) that is spectrally accurate has been developed since the new century [26,51]. By means of the Carleman representation, the method is improved with the computational cost reduced to $O\left(\bar{M}^2\bar{N}^3\log\bar{N}\right)$, where $\bar{M}$ is the number of polar and azimuthal angles [25,45], which is the fastest algorithm been reported to date. To extend the applicability of FSM, novel anisotropic collision kernels were designed and incorporated, which can deal with all inverse power-law potentials (except the Coulomb potential) as well as the Lennard-Jones potential [70,72]. Later, the collision kernel for Lennard-Jones potential was fully resolved, however, the computational cost is increased to $O\left(\bar{M}^2\bar{N}^4\log\bar{N}\right)$ [69]; together with the fast iterative scheme for the streaming [73], the FSM for the Lennard-Jones potential is able to produce very close results to the experimental measurement of Poiseuille and thermal transpiration flows [71]. The FSM has also been extended to the Boltzmann collision operators for gas mixtures [74] with general intermolecular interactions like the Lennard-Jones potential [73]. The developed FSM has been successfully applied to solve many canonical rarefied gas flows, where the computational efficiency is much higher than the low-variance DSMC method for low-speed flow [53,70].

In this paper we aim to further increase the computational efficiency for the Boltzmann equation with full collision operator. The numerical challenge is that the computational cost becomes prohibitive for realistic problems, since: (1) the number of governing equations is large due to the discretization in the velocity space; (2) for each equation, the collision operator needs to be evaluated at every spatial grid point or element (even the Lagrangian methods need spatial mesh for approximation of collision). Therefore, high-order CFD approach is critical to improve efficiency of discretization in the spatial space, thus reduce the computational cost. One of the promising methods for this purpose is the discontinuous Galerkin (DG) method, which was first introduced for the neutron transport equation [54]. The DG method provides advantages including: achieving high-order of accuracy with relatively small effort, easy formulation for arbitrary geometry, straightforward implementation of boundary condition with the same high-order accuracy as in the interior of computational domain, as well as efficient implementation for parallelism and adaptive refinement. After combining an explicit high-order Runge-Kutta time marching scheme, the method has achieved great success in solving convection-dominated problems [15,16]. The explicit DG method has been applied to solve the kinetic model equations [60]. Very recently, it has also been applied to the full Boltzmann equation with variable soft-sphere collision kernel, in which the collision operator is calculated based on a FSM having a cost at the order of $\bar{M}^2 \bar{N}^4 \log \bar{N}$ [35]. It has been shown that the second-order DG method is 15 times faster than the second-order finite volume scheme [60]. However, higher-order explicit DG scheme is not superior to the lower-order one, mainly due to the fact that the time step restricted by the Counrant-Friedrichs-Lewy (CFL) condition becomes extremely small [40]; thus the number of iterations becomes very large in finding steady-state solution with high-order discretization.

Note that the FSM has also been incorporated in the Boltzmann solver based on Lagrangian method for streaming [23]. The solution from this method is currently limited to first-order accuracy in space and time. Besides, the time marching is an explicit scheme, thus the total number of time steps is still enormous to obtain steady solution. It is also interesting to mention that there is a class of methods, named (discrete) unified gas-kinetic scheme, sharing some properties with the semi-Lagrangian scheme [29,30,42,80], in which the flux transport across spatial cell interface contains the evolution of distribution function along the molecular trajectories within a time step due to both streaming and collision. By coupling the evaluations of both streaming and collision, the scale of spatial discretization can be reduced. These methods are first developed based on kinetic model equations. Recently, approximation of the Boltzmann collision operator using FSM is incorporated to correct the relaxation of velocity distribution function to the local equilibrium state beyond the continuum flow regime [42].

In this paper, we propose a high order DG method to solve the full Boltzmann equation, which is devoted to improving the scheme in the following ways:

– An novel scheme is proposed to reduce the computational cost when using FSM to calculate the collision operator, say, by 15 times when using 4th order approximating polynomials on two-dimensional triangular mesh.
– Implicit iterative scheme for the convection term is employed to remove the limitation on time step from the CFL condition. As a result, the superiority of high-order discretization in the DG method can be reflected, which is in sharp contrast to the explicit DG where the CFL number is rather small.
– A strategy based on the sweeping technique for solutions of the local linear systems resulting from DG discretization is introduced, which can avoid solving large sparse

linear system, and stabilize the scheme without using any nonlinear limiter when the high-speed rarefied gas flow is resolved.

The remainder of this paper is organized as follows. In Sect. 2, the Boltzmann equation and FSM are introduced. In Sect. 3, the implicit DG method is described with details in the formulation of collision operator. A scheme to reduce the complexity of DG discretization for the collision operator is proposed in Sect. 4, while the sweeping strategy to solve the linear systems is described in Sect. 5. In Sect. 6, five different problems including one-dimensional shock wave, two-dimensional hypersonic flow past a square cylinder, lid driven cavity flow and two thermal low-speed microscale flows are simulated to assess the accuracy and efficiency of the proposed scheme. Conclusions and outlooks are presented in Sect. 7.

## 2 The Boltzmann Equation and Fast Spectral Method

In kinetic theory, the state of a gas system is described by the one-particle velocity distribution function $f(t, \boldsymbol{x}, \boldsymbol{v})$, which is a function of the time $t$, spatial position $\boldsymbol{x} = (x_1, x_2, x_3)$, and molecular velocity $\boldsymbol{v} = (v_1, v_2, v_3)$. Neglecting the external force, evolution of the velocity distribution function for a single-species monatomic gas is governed by the following Boltzmann equation:

$$\frac{\partial f}{\partial t} + \boldsymbol{v} \cdot \frac{\partial f}{\partial \boldsymbol{x}} = \mathcal{C}(f, f_*), \tag{1}$$

where $\mathcal{C}(f, f_*)$ is the Boltzmann collision operator that is usually split into the gain term $\mathcal{C}_+$ and loss term $\mathcal{C}_-$:

$$\mathcal{C}(f, f_*) = \underbrace{\int \int B(\theta, |\boldsymbol{v} - \boldsymbol{v}_*|) f(\boldsymbol{v}'_*) f(\boldsymbol{v}') \, \mathrm{d}\Omega \mathrm{d}\boldsymbol{v}_*}_{\mathcal{C}_+} - \underbrace{\nu(\boldsymbol{v}) f(\boldsymbol{v})}_{\mathcal{C}_-}. \tag{2}$$

with the collision frequency

$$\nu(\boldsymbol{v}) = \int \int B(\theta, |\boldsymbol{v} - \boldsymbol{v}_*|) f(\boldsymbol{v}_*) \, \mathrm{d}\Omega \mathrm{d}\boldsymbol{v}_*. \tag{3}$$

Note that here $B(\theta, |\boldsymbol{v} - \boldsymbol{v}_*|)$ is the collision kernel; $\boldsymbol{v}$, $\boldsymbol{v}_*$ are the pre-collision molecular velocities of a collision pair, and $\boldsymbol{v}'$, $\boldsymbol{v}'_*$ are the corresponding post-collision molecular velocities; $\Omega$ is the unit vector along the relative post-collision velocity $\boldsymbol{v}' - \boldsymbol{v}'_*$, while $\theta$ is the deflection angle between the pre- and post-collision relative velocities. For simplicity, the time and spatial position is omitted in writing the velocity distribution function, collision operator, and collision frequency.

The velocity distribution function is defined such that $f(t, \boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{v}$ is the number of gas molecules in the phase-space volume $\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{v}$. All macroscopic quantities, such as mass density $\rho$, bulk velocity $\boldsymbol{u} = (u_1, u_2, u_3)$, temperature $T$, stress tension $\boldsymbol{P}$ and heat flux $\boldsymbol{Q} = (Q_1, Q_2, Q_3)$ can then be calculated via velocity moments of this distribution function. For simplicity, we use non-dimensional variables hereafter: $\boldsymbol{x}$ is normalized by a characteristic flow length $H$, $T$ is normalized by a reference temperature $T_0$, $\rho$ is normalized by the average density $\rho_0$ at $T_0$, $\boldsymbol{v}$ and $\boldsymbol{u}$ are normalized by the most probable speed $v_{\mathrm{m}} = \sqrt{2k_{\mathrm{B}}T_0/m}$ with $k_{\mathrm{B}}$ and $m$ being the Boltzmann constant and molecular mass, respectively, $t$ is normalized by $H/v_{\mathrm{m}}$, $f$ is normalized by $\rho_0/mv_{\mathrm{m}}^3$, $\boldsymbol{P}$ is normalized by $\rho_0 k_{\mathrm{B}} T_0/m$, and $Q_i$ is normalized by $\rho_0 k_{\mathrm{B}} T_0 v_{\mathrm{m}}/m$. Therefore, we have

$$\rho = \int f \, \mathrm{d}\boldsymbol{v}, \quad \boldsymbol{u} = \frac{1}{\rho} \int \boldsymbol{v} f \, \mathrm{d}\boldsymbol{v}, \quad T = \frac{2}{3\rho} \int |\boldsymbol{v} - \boldsymbol{u}|^2 f \, \mathrm{d}\boldsymbol{v},$$

$$\boldsymbol{P} = 2 \int (\boldsymbol{v} - \boldsymbol{u}) \otimes (\boldsymbol{v} - \boldsymbol{u}) \, f \, \mathrm{d}\boldsymbol{v}, \quad \boldsymbol{Q} = \int (\boldsymbol{v} - \boldsymbol{u}) |\boldsymbol{v} - \boldsymbol{u}|^2 f \, \mathrm{d}\boldsymbol{v}. \tag{4}$$

The collision kernel $B\left(\theta, |\boldsymbol{v} - \boldsymbol{v}_*|\right)$, depending on the modules of the pre-collision relative velocity and the deflection angle, is determined when the intermolecular potential is given [13]. The detailed structure of the collision kernel is complicated, except that of the ideal hard-sphere molecule. In history, both for analytical and numerical convenience, specific simplification is adopted with the aim to recover correct transport coefficients, which results in various molecular models that are widely used in the DSMC method. The key to these models is that the transport coefficients such as the shear viscosity, thermal conductivity, and diffusion coefficient are recovered over the temperature range considered. In this paper, the collision kernel for the inverse power-law potential is modeled as [45]:

$$B\left(\theta, |\boldsymbol{v} - \boldsymbol{v}_*|\right) = \frac{5|\boldsymbol{v} - \boldsymbol{v}_*|^{2(1-\omega)}}{2^{7-\omega} \Gamma\left(\frac{5-2\omega}{2}\right) Kn} \sin^{1-2\omega}\left(\frac{\theta}{2}\right), \tag{5}$$

where $\Gamma$ is the Gamma function, $\omega$ is the viscosity index (i.e. the shear viscosity $\mu$ of gas is proportional to $T^\omega$) and $Kn$ is the unconfined Knudsen number given at the reference condition:

$$Kn = \frac{\mu(T_0)}{\rho_0 H} \sqrt{\frac{m\pi}{2k_{\mathrm{B}} T_0}}. \tag{6}$$

It is noted that the specific form (5) introduced by Mouhot and Pareschi enables the development of Carleman-representation-based FSM to deterministically compute the collision operator. It has the ability to mimic the growth trend of collision kernel when decreasing the deflection angle and recover correct value of the shear viscosity, however it cannot deal with general forms of soft potentials. By introduce another free-parameter into the collision kernel, the authors have extended the applicability of FSM to all inverse power law potentials (except the Coulomb potential), thus to recover the correct value of diffusion coefficient [70,72]. We also mention that more general collision models including the Lennard-Jones potential has been incorporated into FSM [69], where the computational cost is one order of magnitude higher than that of Eq. (5). Therefore, in this paper Eq. (5) is adopted to demonstrate the efficiency and accuracy of DG method on the spatial discretization. As a matter of fact, if the viscosity index $\omega$ is chosen appropriately, the collision kernel can yield accurate result when compared to that of the realistic Lennard-Jones potential [69,70]. On the other hand, it will be shown that the DG method can reduce the computational cost of the Boltzmann equation when compared to the finite difference method, as the higher computational cost of the Boltzmann collision operator is, the more reduction in the computational time will be.

### 2.1 The Fast Spectral Method

In this paper, the FSM is applied to compute the full Boltzmann collision operator (2), details of which can be found in Refs. [45,69,72]. Firstly, the velocity distribution function is periodized on a truncated domain $\mathcal{D} = [-L, L]^3$ and expanded into Fourier series with $N_1 \times N_2 \times N_3$ components:

$$f(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_{j=-N/2}^{N/2-1} \bar{f}^j(t, \boldsymbol{x}) \exp\left(\iota \boldsymbol{\xi}^j \cdot \boldsymbol{v}\right), \tag{7}$$

$$\bar{f}^j(t, \boldsymbol{x}) = \frac{1}{(2L)^3} \int_{\mathcal{D}} f(t, \boldsymbol{x}, \boldsymbol{v}) \exp\left(-\iota \boldsymbol{\xi}^j \cdot \boldsymbol{v}\right) d\boldsymbol{v}, \tag{8}$$

where $L$ is the maximum truncated velocity, $\iota$ is the imaginary unit, $\bar{f}^j$ is the spectrum of the velocity distribution function, $\boldsymbol{\xi}^j = j\pi/L$ is the discrete frequency with $j = (j_1, j_2, j_3)$ and $N = (N_1, N_2, N_3)$ denoting the index and total number of frequencies. In order to take the advantage of FFT, the discretized frequency components are equally spaced, but the discretized velocity grid points can be non-uniformly distributed to capture the discontinuities and/or large variations in the velocity distribution function.

Then, the gain term in the collision operator and the collision frequency are evaluated through expanding in Fourier series:

$$\mathcal{C}_+ = \sum_{j=-N/2}^{N/2-1} \bar{\mathcal{C}}_+^j \exp\left(\iota \boldsymbol{\xi}^j \cdot \boldsymbol{v}\right), \quad \nu = \sum_{j=-N/2}^{N/2-1} \bar{\nu}^j \exp\left(\iota \boldsymbol{\xi}^j \cdot \boldsymbol{v}\right), \tag{9}$$

where the $j$-th Fourier modes of the gain term in Eq. (2) and collision frequency (3) are calculated from the spectrum $\bar{f}$ as follows [70,72]:

$$\bar{\mathcal{C}}_+^j = \sum_{\substack{l+m=j \\ l,m=-N/2}}^{N/2-1} \bar{f}^l \bar{f}^m \beta(l, m), \quad \bar{\nu}^j = \bar{f}^j \beta(j, j). \tag{10}$$

Here, $\beta$ is the collision kernel mode, whose $(l, m)$-th component is approximated through $M_{\text{qua}}$-point numerical quadrature in spherical coordinates as [72]:

$$\beta(l, m) \simeq \frac{20 \sum_{p,q=1}^{M_{\text{qua}}} \sin(\theta_p) \Psi\left(\sqrt{|\boldsymbol{\xi}^m|^2 - \left(\boldsymbol{\xi}^m \cdot \boldsymbol{e}_{p,q}\right)^2}\right) \Phi\left(\boldsymbol{\xi}^l \cdot \boldsymbol{e}_{p,q}\right) \varpi_p \varpi_q}{2^{7-\omega} \Gamma\left(\frac{5-2\omega}{2}\right) Kn}, \tag{11}$$

with

$$\begin{aligned}
\Psi(a) &= 2\pi \int_0^R \rho^{1-\gamma} J_0(\rho a)\, d\rho, \\
\Phi(a) &= 2 \int_0^R \rho^{2(1-\omega)+\gamma} \cos(\rho a)\, d\rho,
\end{aligned} \tag{12}$$

where $\theta_p$ ($\phi_q$) and $\varpi_p$ ($\varpi_q$) are the $p$ ($q$)-th point and weight of the quadrature rule, respectively, for $\theta, \phi \in [0, \pi]$, $\boldsymbol{e}_{p,q} = (\sin\theta_p \cos\phi_q, \sin\theta_p \sin\phi_q, \cos\theta_p)$, $J_0$ is the zeroth-order Bessel function, and $R$ is the radius of the sphere to support the distribution function, which is chosen approximately as $R = 2\sqrt{2}L/(2 + \sqrt{2})$ to avoid aliasing error [72]. Note that by estimating through numerical quadrature, frequencies $\boldsymbol{\xi}^m$ and $\boldsymbol{\xi}^l$ appear in two different functions in the final form of $\beta(l, m)$, thus Eq. (10) can be calculated by FFT-based convolution [72].

## 3 Implicit Discontinuous Galerkin Method

To obtain stationary solution of the Boltzmann equation, the following implicit iterative scheme can be applied:

$$\bar{v} f^{(t+1)} + \boldsymbol{v} \cdot \frac{\partial f^{(t+1)}}{\partial \boldsymbol{x}} = \bar{v} f^{(t)} + \mathcal{C}\left(f^{(t)}, f_*^{(t)}\right), \tag{13}$$

where the superscripts $(t)$ and $(t+1)$ represent two consecutive iteration steps. The iteration is terminated when convergence to the steady solution is achieved. The parameter $\bar{v}$ is a positive constant which is the reciprocal time step in the backward-Euler method and highly influences convergence property of the iterative scheme: too large (small) $\bar{v}$ results in slow convergence (numerical instability). Usually, to strike a balance between efficiency and stability of the iteration, $\bar{v}$ is chosen to be the order of mean collision frequency $\int v(\boldsymbol{v}) f(\boldsymbol{v}) d\boldsymbol{v}$. Therefore, a safe choice of $\bar{v}$ is the minimum mean collision frequency in the whole computational domain. However, one needs a good estimation for the minimum $\bar{v}$ before calculation.

Another way to find the steady-state solution is to neglect the derivative of distribution function with respect to the time, yielding $\boldsymbol{v} \cdot \partial f/\partial \boldsymbol{x} = \mathcal{C}$. Then, the collision frequency and gain term of the Boltzmann collision operator are evaluated based on the approximation of distribution at the iteration step $(t)$, while other terms are solved at the next iteration step by:

$$v^{(t)}(\boldsymbol{x}, \boldsymbol{v}) f^{(t+1)}(\boldsymbol{x}, \boldsymbol{v}) + \boldsymbol{v} \cdot \frac{\partial f^{(t+1)}(\boldsymbol{x}, \boldsymbol{v})}{\partial \boldsymbol{x}} = \mathcal{C}_+^{(t)}(\boldsymbol{x}, \boldsymbol{v}). \tag{14}$$

In the following sections, we will denote the iterative scheme (13) with mean collision frequency as 'ITR-MEAN' and the iterative scheme (14) with local collision frequency as 'ITR-LOC'. The two iteration schemes can lead to different computational complexity and convergence history, which will be discussed in Sect. 5. Note that the implicit treatment is only applied to the convection term at the $(t+1)$-th iteration step, while the collision operator is explicitly evaluated at the $(t)$-th step. For conciseness, we will omit the index of iteration step in the following unless necessary.

### 3.1 DG Formulation for the Boltzmann Equation

Now we present the DG method to find steady-state solution of rarefied gas flow described by Eqs. (13) and (14). Let $\Delta \in \mathbb{R}^d$ be a computational domain in the $d$-dimensional spatial space with boundary $\partial \Delta$. Then, the domain is partitioned into $M_{el}$ disjoint regular elements $\Delta_i$. The DG method provides an approximate solution to the velocity distribution function $f$ on each element $\Delta_i$ in some piecewise finite element spaces $\mathcal{V}$ of the following form:

$$\mathcal{V} = \{\varphi_r(\boldsymbol{x}) : \varphi_r|_{\Delta_i} \in \mathcal{P}^k(\Delta_i), \ r = 1, \ldots, K, \forall \Delta_i \in \Delta\}, \tag{15}$$

where $\mathcal{P}^k$ denotes the space of $k$-th order polynomials, thus we have

$$f(\boldsymbol{x}, \boldsymbol{v}) = \sum_{r=1}^{K} \varphi_r(\boldsymbol{x}) F_r(\boldsymbol{v}), \tag{16}$$

with $F_r$ being the degree of freedom for the velocity distribution function. In general, the degrees of freedom are unknowns for which the equations are being solved. Together with the basis functions $\varphi_r$, they give the final polynomial estimation of $f$ within a spatial element $\Delta_i$. The number of degree of freedom, $K$, depends on the shape of element employed.

For example, $K = k + 1$ for line segment in one-dimensional (1D) problem, and $K = (k + 1)(k + 2)/2$ for triangular element in two-dimensional (2D) problem.

In order to determine $F_r$, standard techniques of finite element formulations are applied to obtain the weak formulation of the governing system. Introducing $(\cdot)$ and $\langle \cdot \rangle$ as $(a, b)_{\Delta_i} = \int_{\Delta_i} (a \cdot b) \, d\mathbf{x}$ and $\langle a, b \rangle_{\partial \Delta_i} = \int_{\partial \Delta_i} (a \cdot b) \, d\Upsilon$ to denote operators on the element $\Delta_i$ and its boundary $\partial \Delta_i$, respectively, we find the approximation of distribution function satisfies the following equation (take the ITR-MEAN scheme (13) as an example):

$$- (\nabla \varphi_s, \mathbf{v} f)_{\Delta_i} + \langle \varphi_s, \hat{\mathbf{H}} \cdot \mathbf{n} \rangle_{\partial \Delta_i} + (\varphi_s, \bar{\nu} f)_{\Delta_i} = (\varphi_s, \mathcal{C})_{\Delta_i} + (\varphi_s, \bar{\nu} f)_{\Delta_i}, \quad (17)$$

where $s = 1, \ldots, K$, $\mathbf{n}$ is the outward unit normal vector, and $\hat{\mathbf{H}}$ is the numerical flux that depends on the solutions from both sides of $\partial \Delta_i$, since the solution of $f$ is discontinuous there. We define the numerical flux from the first-order upwind principle as:

$$\hat{\mathbf{H}} \cdot \mathbf{n} = \frac{1}{2} \mathbf{v} \cdot \mathbf{n} (f + f_{\text{ext}}) + \frac{1}{2} |\mathbf{v} \cdot \mathbf{n}| (f - f_{\text{ext}}), \quad (18)$$

with $f_{\text{ext}}$ being the distribution from a neighboring element that shares the boundary $\partial \Delta_i$ with $\Delta_i$. If $\partial \Delta_i$ is at the boundary of computational domain, i.e. $\partial \Delta_i \cap \partial \Delta \neq 0$, $f_{\text{ext}}$ is evaluated using the given boundary condition.

Now, we focus on the formulation of $(\varphi_s, \mathcal{C})_{\Delta_i} = (\varphi_s, \mathcal{C}_+)_{\Delta_i} - (\varphi_s, \nu f)_{\Delta_i}$ in Eq. (17). Inserting the polynomial expansion of distribution function (16) into Eq. (8), the $j$-th spectrum component of the distribution function can be rewritten in the following polynomial form:

$$\bar{f}^j(\mathbf{x}) = \sum_{r=1}^{K} \varphi_r(\mathbf{x}) \bar{F}_r^j, \quad (19)$$

where $\bar{F}_r^j = \frac{1}{(2L)^3} \int_{\mathcal{D}} F_r(\mathbf{v}) \exp\left(-\iota \boldsymbol{\xi}^j \cdot \mathbf{v}\right) d\mathbf{v}$ is the spectrum of the degree of freedom.

With some algebraic calculations from Eqs. (9) and (10), the DG discretization of the gain term of the Boltzmann collision operator and the collision frequency is expressed as

$$\mathcal{C}_+ = \sum_{p=1}^{K} \sum_{r=1}^{K} \varphi_p \varphi_r \Xi_{p,r}, \quad \nu = \sum_{p=1}^{K} \varphi_p \Lambda_p, \quad (20)$$

where

$$\Xi_{p,r} = \sum_{j=-N/2}^{N/2-1} \sum_{\substack{l+m=j \\ l,m=-N/2}}^{N/2-1} \bar{F}_p^l \bar{F}_r^m \beta(l, m) \exp\left(\iota \boldsymbol{\xi}^j \cdot \mathbf{v}\right),$$

$$\Lambda_p = \sum_{j=-N/2}^{N/2-1} \bar{F}_p^j \beta(j, j) \exp\left(\iota \boldsymbol{\xi}^j \cdot \mathbf{v}\right). \quad (21)$$

Finally, we obtain that

$$(\varphi_s, \mathcal{C}_+)_{\Delta_i} = \sum_{p=1}^{K} \sum_{r=1}^{K} (\varphi_s, \varphi_p \varphi_r)_{\Delta_i} \Xi_{p,r}, \quad (22)$$

$$(\varphi_s, \nu f)_{\Delta_i} = \sum_{p=1}^{K} \sum_{r=1}^{K} (\varphi_s, \varphi_p \varphi_r)_{\Delta_i} \Lambda_p F_r. \quad (23)$$

## 3.2 Discretization in the Molecular Velocity Space

In order to obtain the macroscopic flow properties (4) and the spectrum (8), integrals with respect to the velocity space should be calculated. Numerically, the truncated but continuous velocity domain $\mathcal{D}$ needs to be represented by $M := (M_1, M_2, M_3)$ discrete points $\boldsymbol{v}^{j'}$ and integrals are approximated by certain quadrature rules, e.g. $\rho = \sum_{j'=1}^{M} f\left(\boldsymbol{x}, \boldsymbol{v}^{j'}\right) w^{j'}$ with $w^{j'}$ being the quadrature weight for the corresponding discretized velocity point $\boldsymbol{v}^{j'}$. The discrete velocities are not necessarily equidistant, especially for low-speed microflow with large Knudsen number, where the distribution function varies rapidly around $\boldsymbol{v} = 0$ due to gas-wall interaction and nonuniform velocity points with refinement in this area is more efficient to capture the variation of $f$ [61]. However, it should be emphasized that the FFT-based convolution can be efficiently employed only when the frequency space is uniformly discretized; also, since the FSM has spectral accuracy, the number of frequency components can be smaller than that of velocity grid points [70].

As a consequence, we need to approximate the distribution function at each discrete velocity point by solving $M_1 \times M_2 \times M_3 \times K$ equations on each element $\Delta_i$ (take the ITR-MEAN scheme (13) as an example):

$$-\left(\nabla\varphi_s, \boldsymbol{v}^{j'} f^{j'}\right)_{\Delta_i} + \langle\varphi_s, \hat{\boldsymbol{H}}^{j'} \cdot \boldsymbol{n}\rangle_{\partial\Delta_i} + \left(\varphi_s, \bar{\nu} f^{j'}\right)_{\Delta_i} = \left(\varphi_s, \mathcal{C}^{j'}\right)_{\Delta_i} + \left(\varphi_s, \bar{\nu} f^{j'}\right)_{\Delta_i}, \tag{24}$$

where $f^{j'} = f\left(\boldsymbol{x}, \boldsymbol{v}^{j'}\right)$, $\hat{\boldsymbol{H}}^{j'} = \hat{\boldsymbol{H}}\left(f^{j'}, f_{\text{ext}}^{j'}\right)$, and $\mathcal{C}^{j'} = \mathcal{C}\left(f^{j'}, f_*^{j'}\right)$ denote the corresponding variables at each discrete velocity. The resulting governing equations can be re-written into matrix form as:

$$\mathbf{A}^{i,j'}\mathbf{F}_i^{j'} + \mathbf{B}^{\text{ext},j'} = \mathbf{S}^{i,j'}, \tag{25}$$

where $\mathbf{F}_i^{j'} = [F_1(\boldsymbol{v}^{j'}), \dots, F_r(\boldsymbol{v}^{j'}), \dots]^{\mathrm{T}}$ are the unknowns, i.e. the vector of degrees of freedom of $f^{j'}$ on $\Delta_i$. Other coefficient matrices are given in the Appendix.

The strategy to solve the linear systems that are coupled through numerical fluxes over all the spatial elements will be described in Sect. 5.

## 3.3 Boundary Condition

At the boundary of computational domain, to determine the flux for an element $\Delta_i$, the distribution function obtained from the exterior of the element, $f_{\text{ext}}$, is described by a given boundary condition $b^{j'}$. In this paper, the diffuse boundary condition for the gas-surface interaction is used. Suppose the solid wall moves with a constant speed $\boldsymbol{u}_{\text{w}}$, and has a temperature $T_{\text{w}}$ that can either be a constant or vary along the wall, the distribution function for reflected molecules [i.e. when $\left(\boldsymbol{v}^{j'} - \boldsymbol{u}_{\text{w}}\right) \cdot \boldsymbol{n}_{\text{w}} \leq 0$, $\boldsymbol{n}_{\text{w}}$ is the outward unit normal vector of the solid surface] is given by the equilibrium distribution:

$$b^{j'} = \frac{\rho_{\text{w}}}{(\pi T_{\text{w}})^{3/2}} \exp\left(-\frac{|\boldsymbol{v}^{j'} - \boldsymbol{u}_{\text{w}}|^2}{T_{\text{w}}}\right), \tag{26}$$

where, $\rho_{\mathrm{w}}$ is defined by:

$$\sum_{\left(\boldsymbol{v}^{j'}-\boldsymbol{u}_{\mathrm{w}}\right)\cdot\boldsymbol{n}_{\mathrm{w}}<0} \left(\boldsymbol{v}^{j'}-\boldsymbol{u}_{\mathrm{w}}\right)\cdot\boldsymbol{n}_{\mathrm{w}}f^{j'} + \sum_{\left(\boldsymbol{v}^{j'}-\boldsymbol{u}_{\mathrm{w}}\right)\cdot\boldsymbol{n}_{\mathrm{w}}\leq0} \left(\boldsymbol{v}^{j'}-\boldsymbol{u}_{\mathrm{w}}\right)\cdot\boldsymbol{n}_{\mathrm{w}}b^{j'} = 0 \quad (27)$$

such that the mass flux across wall is equal to zero. Implementation of other types of boundary conditions such as symmetry boundary, far-pressure inlet/outlet boundaries, and supersonic inlet/outlet boundaries can be found in Ref. [60].

## 4 Reduction of Computational Complexity in DG Formalism

The major computational cost to solve the system (25) arises from two parts: (1) evaluating collision operator and (2) solving linear equations. In this section, we analyze the computational complexity for the evaluation of collision operator, while the solution of linear systems will be discussed in Sect. 5. For simplification, we assume that equidistant discrete molecular velocities and frequencies are employed with $M = N$ and $N_1 = N_2 = N_3 = \bar{N}$. Then, at each iterative step, equipped with the FFT-based convolution, the computational cost is $O\left(K^2 M_{\mathrm{el}} M_{\mathrm{qua}}^2 \bar{N}^3 \log \bar{N} + K^3 M_{\mathrm{el}} \bar{N}^3\right)$, in which the first term arises in the calculation of $\varXi_{p,r}$ and $\varLambda_r$ in Eq. (21), while the second term is for conducting the loops in Eqs. (22) and (23).

We propose an approach to reduce the cost in evaluating the Boltzmann collision operator. In the following discussion, we will omit the index of discrete molecular velocities $j'$. The approach may be described heuristically in the following manner. If we choose the supporting polynomials as nodal shape functions:

$$\varphi_r\left(\boldsymbol{x}_p\right) = \begin{cases} 0, & \text{if } r \neq p, \\ 1, & \text{if } r = p, \end{cases} \quad (28)$$

where $\boldsymbol{x}_p$ is the nodal points for interpolation, the degree of freedom $F_r$ is actually the nodal value of distribution function, say $f(\boldsymbol{x}_r)$. We assume that the distribution of $\mathcal{C}$ within an element might as well be estimated by the nodal approximation such that the loss and gain terms in the collision operator are approximated by the corresponding nodal values:

$$\begin{aligned} \mathcal{C}_+ &\simeq \sum_{r=1}^{K} \varphi_r \tilde{\varXi}_r, \\ \mathcal{C}_- &\simeq \sum_{r=1}^{K} \varphi_r \varLambda_r F_r, \end{aligned} \quad (29)$$

where $\tilde{\varXi}_r = \varXi_{r,r}$ is evaluated from the nodal value of the distribution function $F_r$. As a result, the computational cost of $(\varphi_s, \mathcal{C})_{\Delta_i}$ in Eqs. (22) and (23) is reduced to $O\left(K M_{\mathrm{el}} M_{\mathrm{qua}}^2 \bar{N}^3 \log \bar{N} + K^2 M_{\mathrm{el}} \bar{N}^3\right)$, that is, by $K$ times; this is considerable especially when high-order approximation polynomials are employed. For instance, nominally, 14 times less cost is expected when $k = 4$ in 2D problem on triangular mesh.

It is interesting to note that, in the recent paper where an explicit DG Boltzmann solver has been developed, the singular value decomposition is proposed to reduce the computational cost [35]. The singular value decomposition is pre-computed to the $K \times K$ matrix for $(\varphi_s, \varphi_p \varphi_r)$. Thus, the computational cost for the loops in Eqs. (22) and (23) can be reduced to

$O\left(K^2 M_{\mathrm{el}} \bar{N}^3\right)$. However, the computational complexity for the collision operators by FSM [Eq. (21)] remains unchanged and always consumes the majority of CPU time; thus this saving may not be in the order of magnitude.

Comparing Eq. (29) to Eqs. (22) and (23), it is found that the error induced by the reduced calculation is:

$$
\left| \sum_{p=1}^{K} \sum_{r=1}^{K} \varphi_p \varphi_r \left( \varXi_{p,r} - \varLambda_p F_r \right) - \sum_{r=1}^{K} \varphi_r \left( \varXi_{r,r} - \varLambda_r F_r \right) \right|
$$

$$
\leq \left| \sum_{j=-N/2}^{N/2-1} \sum_{\substack{l+m=j \\ l,m=-N/2}}^{N/2-1} \left( \bar{f}^l \bar{f}^m - \sum_{r=1}^{K} \varphi_r \bar{F}_r^l \bar{F}_r^m \right) \beta\,(l,m) \exp\left(\iota \boldsymbol{\xi}^j \cdot \boldsymbol{v}\right) \right| \qquad (30)
$$

$$
+ \left| \sum_{l=-N/2}^{N/2-1} \sum_{m=-N/2}^{N/2-1} \left( \bar{f}^l \bar{f}^m - \sum_{r=1}^{K} \varphi_r \bar{F}_r^l \bar{F}_r^m \right) \beta\,(l,l) \exp\left(\iota \boldsymbol{\xi}^l \cdot \boldsymbol{v}\right) \exp\left(\iota \boldsymbol{\xi}^m \cdot \boldsymbol{v}\right) \right|
$$

which is small when the variation of distribution function within a spatial element is not significant. Note that Eq. (30) is obtained by simple algebraic calculus using the relations given by Eqs. (7)–(10). In Sect. 6, we are going to valid this approximation numerically. The scheme with full calculation of collision terms (22) and (23) is labeled as 'DG-FULL', while the one using reduced calculation (29) is labeled as 'DG-RED'.

## 5 Sweeping Technique to Solve Linear Systems

Now we present the strategy to solve the linear systems resulting from the DG discretization. In the linear equations (25) on each spatial element $\Delta_i$, the unknown distribution function on neighboring element appears in $\mathbf{B}^{\mathrm{ext},j'}$, where the usual treatment in implicit DG is to assemble the linear systems over all the spatial elements and solve a large sparse linear equation to determine the unknowns simultaneously. However, this requires huge memory and is prohibitively expensive in solving the Boltzmann equation, since we have to solve not one but a large number (e.g. several ten thousands) of large sparse linear systems. The matrix-free technique might be useful to improve the scheme [18]. In this paper, a more intuitive and simpler strategy is adopted. Note that similar algorithms have formed the basis of many modern radiation transport codes [41].

Due to the fact that the upwind flux is applied, it is important to notice that only the distribution function on neighboring elements in the upwind side appears in $\mathbf{B}^{\mathrm{ext},j'}$. Thus, the solution of $f^{j'}$ on $\Delta_i$ can be obtained by solving the small linear system (25), once $f_{\mathrm{ext}}$ on the upwind side is known, or it is equipped with prescribed boundary conditions. Hence, starting from the element at the inflow boundary of computational domain, we can obtain the solution of $f^{j'}$ sequentially for all the elements.

This sweeping technique, which does not require to assemble large sparse linear system, relies on finding an ordering of the spatial elements, which is determined by the characteristic wind direction (i.e. the direction of molecular velocity). The key to ensuring feasible implementation of the sweeping technique in solving the Boltzmann equation is that, the discrete molecular velocity is fixed in the governing equations. Hence, we can find and store the spatial element ordering for each discrete velocity immediately after discretization and before the first iteration.
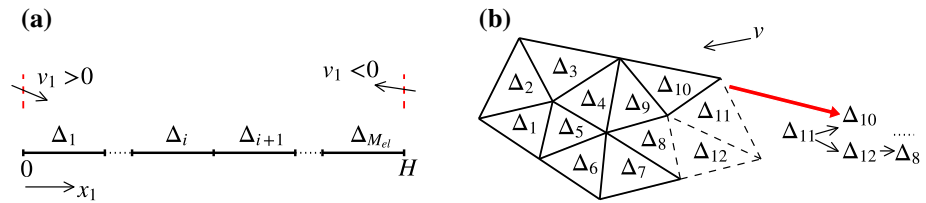
**(a)**                                                  **(b)**



**Fig. 1** Schematic demonstration for determination of the spatial element ordering with respect to a given molecular velocity. **a** Line mesh for 1D problem. **b** Triangular mesh for 2D problem

For a given discrete molecular velocity $\boldsymbol{v}^{j'}$, the topological ordering is easily found in 1D cases. As shown in Fig. 1a, a 1D computational domain $[0, H]$ is parallel to the $x_1$ axis. When $v_1^{j'} > 0$, starting from the boundary at $x_1 = 0$, the spatial ordering is of ascending order in index $i$, while when $v_1^{j'} < 0$ the spatial ordering has a descending order in $i$ starting from the boundary at $x_1 = H$. For higher dimensional problem, we assume that the spatial grid is paved with convex elements and the element ordering is acyclic. A simple topological sorting algorithm [46] is applied: gradually removing elements that have no incoming flux from the elements left in the computational domain, placing them in the ordering, until no element remains. Figure 1b illustrates the schematic demonstration for the sorting procedure, where the ordering starts from the element $\Delta_{11}$ since it has only one inflow boundary located at the boundary of the computational domain. After removing $\Delta_{11}$, either element $\Delta_{10}$ or $\Delta_{12}$ will be put into the ordering, because there is no flux flowing from the elements left in the computational domain to these two elements. Note that the sequence of $\Delta_{11}$ and $\Delta_{12}$ in the ordering is interchangeable, since they do not share any common interface. The pseudo-code of the algorithm can be found in Ref. [46] (Algorithm 3.2.2).

In Sect. 4, we have mentioned that solving the linear systems is one of the major costs in computational resources. On the basis of sweeping technique, if we use $LU$-fabrication-based direct solver to solve the linear equations, the computational cost is $O\left(2/3K^3 M_{\text{el}} \bar{N}^3 + 2K^2 M_{\text{el}} \bar{N}^3\right)$, since we have $M_{\text{el}} \bar{N}^3$ systems and each has a coefficient matrix of rank $K$. Note that we have assumed that the number of discrete velocities in each direction is $\bar{N}$. In the ITR-MEAN scheme (13), the cost to solve linear equations can be reduced to $O\left(2K^2 M_{\text{el}} \bar{N}^3\right)$ due to the fact that the coefficient matrix $\mathbf{A}^{i,j'}$ remains unchanged during the iterations and $LU$-decomposition can be calculated and stored before the first iteration. The computational cost for $LU$-decomposition is roughly $K/3$ times of that for substitution in solving the linear equations, which becomes large as the grid density and/or the order of approximating polynomial increases. For example, when $k = 4$ on triangular mesh, the computational complexity of $LU$-decomposition is 4 times larger than that of substitution. Therefore, computing $LU$-decomposition before iteration and only executing substitution during iteration can further save CPU time.

## 6 Numerical Results and Discussions

The DG method with $k$ up to 4 is applied to solve the Boltzmann equation with full collision operator. The convergence criterion for the iterative schemes described above is that the global relative residual in the flow property $\mathcal{Q}$ between two successive iteration steps:

**Table 1** Flow properties across normal shock waves

| | Ma = 2.05 | | Ma = 9.0 | |
| | Upstream | Downstream | Upstream | Downstream |
|---|---|---|---|---|
| $T$ | 1.0 | 2.144 | 1.0 | 26.185 |
| $\rho$ | 1.0 | 2.334 | 1.0 | 3.857 |
| $u_1$ | 1.871 | 0.802 | 8.216 | 2.130 |

$$R_{\mathcal{Q}} = \frac{\left| \int_{\Delta} \left( \mathcal{Q}^{(t+1)} - \mathcal{Q}^{(t)} \right) \mathrm{d}\boldsymbol{x} \right|}{\left| \int_{\Delta} \mathcal{Q}^{(t)} \mathrm{d}\boldsymbol{x} \right|}, \tag{31}$$

is less than a threshold value $\epsilon$.

The following tests are performed in double precision on a workstation with Intel Xeon-E5-2680 processors and 128 GB RAM. During iteration, we call the routines in Intel Math Kernel Library (MKL) to conduct $LU$-fabrication and solve linear equations. For the calculation of collision kernel $\beta\,(l, m)$, the trapezoidal rule is applied and we set $M_{\mathrm{qua}} = 5$ in Eq. (11) that is adequate to maintain the spectral accuracy of FSM [72]. Due to the fact that we only consider 1D and 2D flows, symmetry of the distribution function in the third ($v_3$) direction allows us to reduce the computational cost of Eq. (11) by half, that is, $\theta$ can be limited to the range of $[0, \pi/2]$; more details can be found in Ref. [70].

## 6.1 One-Dimensional Normal Shock Wave

We first simulate the normal shock wave problem to assess the accuracy and efficiency of the proposed method for steady-state solution of the Boltzmann equation. Due to the absence of boundary effects, the flow is ideal to test the accuracy of DG discretization for streaming and the FSM approximation for the Boltzmann collision operator in capturing highly non-equilibrium phenomena, especially to validate the scheme with reduced DG calculation as described in Sect. 4. The argon gas is considered with Mach numbers $Ma = 2.05$ and $Ma = 9.0$. We use the same parameters as those in Alsmeyer's experiments [2]: the upstream density $\rho_0 = 1.067 \times 10^{-4}$ kg/m$^3$ and temperature $T_0 = 300$ K, corresponding to the mean free path and collision frequency of hard sphere molecules as $\bar{\lambda} = 1.098 \times 10^{-3}$ m and $\bar{\nu} = 3.633 \times 10^5$ s$^{-1}$, respectively. For all the DG results, the length scale is normalized with $H = \bar{\lambda}$ resulting in $Kn = 5\pi/16$. The 1D computational domains $\Delta$ in the $x_1$ direction are $[-20, 20]$ and $[-30, 30]$ for $Ma = 2.05$ and $Ma = 9.0$ cases, respectively, which are partitioned by line elements with uniform length. The dimensionless up/downstream conditions normalized by the upstream properties are listed in Table 1. Initially, the domains $x_1 \leq 0$ and $x_1 > 0$ are setup by the equilibrium distributions at upstream and downstream conditions, respectively. The implicit iteration scheme (14) with local collision frequency. i.e. ITR-LOC is applied. Iteration is terminated when $\max\{R_T, R_n, R_{|u_1|}\} < 10^{-5}$, see Eq. (31). When $Ma = 2.05$, the truncated velocity domain $[-8, 8]^3$ is divided into $32^3$ uniform points, while when $Ma = 9.0$, the velocity domain $[-30, 30]^3$ is divided into $96 \times 64 \times 64$ uniform points. The same number of uniform frequencies are used in the approximation of full Boltzmann collision operator.

Numerical tests show that by using the sweeping technique, the implicit DG method is stable without any limiter in solving the 1D normal shock structure. Figure 2 illustrates the DG results of normalized flow velocity, density and temperature, compared with the DSMC results and experimental data [2]. The DSMC results presented here are computed using the
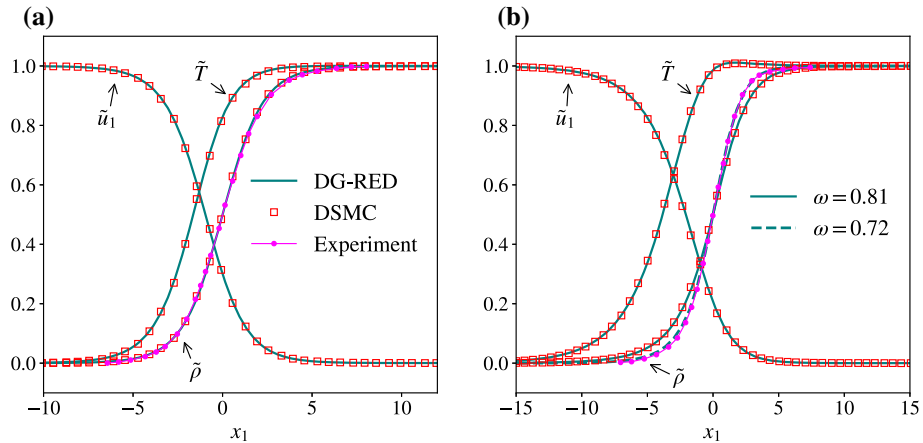
**Fig. 2** Profiles of normalized flow velocity $\tilde{u}_1 = \frac{u_1 - u_{1,R}}{u_{1,L} - u_{1,R}}$, density $\tilde{\rho} = \frac{\rho - \rho_L}{\rho_R - \rho_L}$ and temperature $\tilde{T} = \frac{T - T_L}{T_R - T_L}$ for normal shock wave of argon gas at **a** $Ma = 2.05$ and **b** $Ma = 9.0$. The subscripts 'L' and 'R' denote the properties in upstream and downstream, respectively. The DG-RED solutions are obtained with $k = 4$, $M_{el} = 16$. The ITR-LOC scheme (14) is applied in implicit iteration

code developed and verified in Ref. [62]. In order to ensure accuracy of the DSMC method, the cell sizes and time steps are set to be $\sim 0.13\bar{\lambda}$ and $\sim 0.12/\bar{v}$, respectively. The average number of molecules per spatial cell is about 50. About 30,000 iterations are needed to reach the steady-state solutions. To obtain smooth results, macroscopic flow properties are sampled over another 100,000 steps. For comparison, the viscosity index in both methods are set as $\omega = 0.81$. The DG results are obtained using 4th-order approximating polynomial on 16 elements, which agree well with those of DSMC simulation (the profiles from the DG-FULL scheme are not shown, since they overlap with the ones of DG-RED). We also compare the DG solutions for density with the experimental data. For $Ma = 2.05$, the agreement is good, although slight discrepancy can be observed in the downstream side of the shock wave. For $Ma = 9.0$ where the non-equilibrium effect is strong, the DG solution agrees well with the DSMC one. However, disagreement between the DG (DSMC) solutions and experimental data enlarges, where the density changes more rapidly in the experiment. Actually, the profiles in high Mach number flow are more sensitive with respect to the value of viscosity index $\omega$. The works in Refs. [66,75] suggest that setting $\omega$ to 0.7 the Boltzmann solver or DSMC can produce results close to the experimental measurements. Hence, we include the DG-RED solution with $\omega = 0.72$ obtain an improved agreement, see dash lines in Fig. 2b.

To further validate the DG-RED scheme, we compare the marginal distribution functions $\int f \mathrm{d}v_2 \mathrm{d}v_3$ at different locations of the shock wave with those obtained using the DG-FULL scheme. The profiles are plotted in Fig. 3. To ensure the accuracy of DG-FULL results, we have doubled the number of discrete velocity and frequency points in the longitudinal direction. It is demonstrated that in low Mach number flow, the velocity distribution functions are close to the corresponding equilibrium (Gaussian) distribution. As Mach number increases, the distribution functions within the shock wave structure greatly deviate from the Gaussian profile. The comparison shows that, even for highly non-equilibrium flow, the DG-RED scheme can produce correct solution, so that the numerical error brought by the reduced calculation of collision operator is negligible.
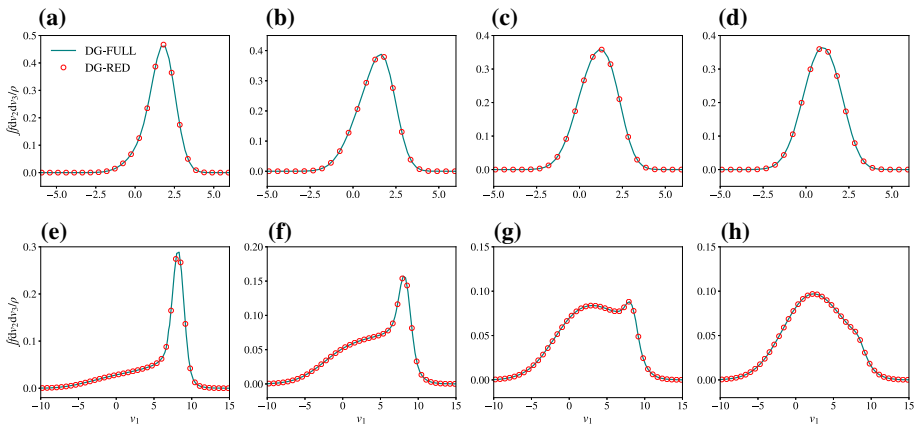
**Fig. 3** Comparison of the marginal distribution functions $\int f \mathrm{d}v_2 \mathrm{d}v_3/\rho$ from the DG-RED and DG-FULL schemes: first row is ones for $Ma = 2.05$ presented at **a** $\rho = 1.197$, **b** $\rho = 1.468$, **c** $\rho = 1.766$ and **d** $\rho = 1.991$; second row is ones for $Ma = 9.0$ presented at **e** $\rho = 1.423$, **f** $\rho = 1.943$, **g** $\rho = 2.543$ and **h** $\rho = 3.123$. For DG-FULL, the molecular velocity domains $[-8, 8]^3$ and $[-30, 30]^3$ are uniformly divided into $64 \times 32 \times 32$ and $192 \times 64 \times 64$ points for $Ma = 2.05$ and $9.0$, respectively. The ITR-LOC scheme (14) is applied in implicit iteration

Another important property of a shock wave with $Ma > \sqrt{9/5}$ in a monatomic gas is the overshoot of temperature associated with the longitudinal component of thermal velocity, $T_x$, which could be larger than the gas temperature behind the front of shock due to the non-equilibrium in translational energies of longitudinal and transversal directions. The analytical form of $T_x$ is related to density $\rho$ as [77]:

$$T_{x,\mathrm{an}} = \frac{1}{3}\left[\frac{(5Ma^2 + 3)}{\rho} - 5\left(\frac{Ma}{\rho}\right)^2\right]. \tag{32}$$

Based on $T_x$, we compare the convergence behavior of DG-RED and DG-FULL schemes with respect to various orders of approximating polynomials $k$ and numbers of spatial elements $M_{\mathrm{el}}$. The relative $L_2$ error of $T_x$ that is evaluated as

$$\mathcal{E} = \frac{\int_\Delta \left(T_x - T_{x,\mathrm{an}}\right)^2 \mathrm{d}x_1}{\int_\Delta T_{x,\mathrm{an}}^2 \mathrm{d}x_1}, \tag{33}$$

the number of iteration steps, and the total CPU time are listed in Table 2.

All the tests are conducted on single processor, and the internal parallelism for MKL functions is not activated. It is shown that for each $k$, as the number of elements increases, errors of $T_x$ gradually converges to 0.016% and 0.036% for Mach numbers of 2.05 and 9.0, respectively. Higher order approximating polynomial requires fewer elements to obtain converged results. The numbers of iterative steps to reach the steady-state solutions also converge to fixed values of around 201 and 225 for Mach numbers of 2.05 and 9.0, respectively. Therefore, compared to the lower-order scheme, the higher-order discretization consumes less CPU time to obtain solution with the same order of accuracy. For example, for $Ma = 2.05$, the DG-FULL scheme with $k = 4$ cost about 30% less CPU time to produce solution with $\mathcal{E} = 0.016\%$ on the mesh of 16 segments, compared to the one with $k = 3$ that obtains the same level of accuracy on 32 segments.

**Table 2** Comparisons between the DG-FULL (with FULL calculation of $(\varphi_s, C)_{\Delta_i}$) and DG-RED (with reduced calculation of $(\varphi_s, C)_{\Delta_i}$) in terms of the relative $L_2$ error ($\mathcal{E}$) of longitudinal temperature $T_x$ (compared with the analytical result), the number of iterations (Itr denotes the number of iteration steps to reach the convergence criterion $\max\{R_T, R_\rho, R_{|u_1|}\} < 10^{-5}$), and the CPU time $t_c$. Normal shock wave is considered. The ITR-LOC scheme (14) is applied in implicit iteration

| k | Ma = 2.05 DG-FULL $M_{el}$ | $\mathcal{E} \times 10^{-2}$ | Itr | $t_c$ (h) | DG-RED $\mathcal{E} \times 10^{-2}$ | Itr | $t_c$ (h) | Ma = 9.0 DG-FULL $M_{el}$ | $\mathcal{E} \times 10^{-2}$ | Itr | $t_c$ (h) | DG-RED $\mathcal{E} \times 10^{-2}$ | Itr | $t_c$ (h) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4 | 14.656 | 208* | 0.02 | 7.734 | 204** | 0.01 | 8 | 12.91 | 201 | 0.64 | 5.814 | 278 | 0.61 |
|   | 8 | 3.450 | 194 | 0.03 | 2.159 | 219 | 0.02 | 16 | 1.668 | 222 | 1.32 | 1.553 | 252 | 1.08 |
|   | 16 | 0.643 | 199 | 0.06 | 0.622 | 207 | 0.03 | 32 | 0.396 | 225 | 2.57 | 0.381 | 238 | 1.96 |
|   | 32 | 0.158 | 201 | 0.11 | 0.152 | 203 | 0.06 | 64 | 0.088 | 225 | 5.19 | 0.087 | 229 | 3.72 |
| 2 | 4 | 3.343 | 189 | 0.03 | 1.629 | 203 | 0.01 | 8 | 0.824 | 225 | 1.39 | 0.361 | 231 | 0.75 |
|   | 8 | 0.202 | 199 | 0.06 | 0.187 | 198 | 0.02 | 16 | 0.239 | 225 | 2.73 | 0.247 | 226 | 1.49 |
|   | 16 | 0.097 | 200 | 0.13 | 0.097 | 200 | 0.04 | 32 | 0.049 | 225 | 5.75 | 0.049 | 225 | 2.86 |
|   | 32 | 0.020 | 201 | 0.24 | 0.020 | 201 | 0.08 | 64 | 0.036 | 225 | 11.24 | 0.036 | 225 | 5.52 |
| 3 | 4 | 0.221 | 190 | 0.05 | 0.350 | 201 | 0.02 | 8 | 0.550 | 225 | 2.58 | 0.595 | 235 | 1.06 |
|   | 8 | 0.216 | 198 | 0.09 | 0.219 | 199 | 0.03 | 16 | 0.066 | 225 | 5.06 | 0.065 | 226 | 1.98 |
|   | 16 | 0.022 | 200 | 0.21 | 0.022 | 200 | 0.06 | 32 | 0.036 | 225 | 9.38 | 0.036 | 225 | 3.88 |
|   | 32 | 0.016 | 201 | 0.41 | 0.016 | 201 | 0.11 | 64 | 0.036 | 225 | 16.22 | 0.036 | 225 | 7.49 |
| 4 | 4 | 0.557 | 189 | 0.09 | 0.584 | 188 | 0.02 | 8 | 0.236 | 225 | 3.89 | 0.233 | 226 | 1.30 |
|   | 8 | 0.060 | 198 | 0.15 | 0.061 | 198 | 0.04 | 16 | 0.037 | 225 | 7.35 | 0.037 | 225 | 2.53 |
|   | 16 | 0.016 | 200 | 0.30 | 0.016 | 201 | 0.07 | 32 | 0.036 | 225 | 13.15 | 0.036 | 225 | 4.85 |
|   | 32 | 0.016 | 202 | 0.59 | 0.016 | 201 | 0.14 | 64 | 0.036 | 225 | 25.85 | 0.036 | 225 | 9.55 |

Due to round-off errors, *the solution is only converged to residual of $2 \times 10^{-5}$ and **the solution is only converged to residual of $3 \times 10^{-5}$

It is found that the DG-RED scheme can preserve these convergence properties. That is, by using the same order of approximating polynomial on the same mesh, DG-RED and DG-FULL require the same number of iterative steps to obtain solution of the same order of accuracy. However, DG-RED can significantly save the computational cost in terms of CPU time. Higher degree of approximating polynomial leads to more saving. For example, when $Ma = 9.0$, to obtained solution of $\mathcal{E} = 0.036\%$, both schemes need 64, 32 and 16 spatial elements for $k = 2$, 3 and 4, respectively, and the CPU time consumed by DG-RED is about 50%, 41%, and 36% of that by DG-FULL.

## 6.2 Hypersonic Flow Past a Square Cylinder

Now we consider a 2D high-speed flow. The DG-RED scheme of $k = 4$ is applied to compute the hypersonic flow past a square cylinder having a dimension of $1 \times 1$ and a constant wall temperature of $T_{\mathrm{w}} = 1.0$. The free stream has dimensionless temperature and density of $T_0 = 1.0$ and $\rho_0 = 1.0$. The Mach number and Knudsen number in the free stream are 5.0 and 0.13, respectively. As shown in Fig. 4a, the argon gas of viscosity index $\omega = 0.81$ moves from left to right along the $x_1$ direction. The computational domain is chosen with extension up to 1.95, 7.5 and 5.5 away from the cylinder in the upwind, downstream and $x_2$ directions, respectively. Due to symmetry, only half of the flow field is considered. The boundary conditions and triangular spatial meshes are also illustrated in Fig. 4a. In addition to the full-diffuse solid surface, the lower boundaries parallel to $x_1$ are symmetric boundaries, while other boundaries are set as hypersonic inlet/outlet boundaries where the velocity distribution function is the equilibrium distribution at the free-stream condition. 1490 unstructured triangles are employed to discretize the computational domain, with refinement near the solid surface. The truncated molecular velocity space $[-13, 13]^3$ is discretized by $48 \times 48 \times 48$ equidistant points, and the same number of uniform frequencies are used in the evaluation of full Boltzmann collision operator. The flow field is initialized by the free-stream condition, and the ITR-LOC scheme (14) is applied, which needs 346 steps to reach the convergence criterion of $\max\{R_T, R_\rho, R_{|\boldsymbol{u}|}\} < 5 \times 10^{-5}$. The test is run on 28 processors using OpenMP for parallelism and consumes about 24.6 h of wall time.

At the very beginning of iteration, strong discontinuity appears in the upwind side of the square cylinder due to intense stagnation effect of gas flow, and the DG scheme can generate spurious oscillation which may cause the approximated distribution functions to be negative. As a consequence, the loss term will become the gain term and the iteration will lead to unphysical blowup. To tackle this problem, instead of using any nonlinear limiter as one usually does, we take absolute value to the negative degrees of freedom after solving the linear systems at each iterative step. Numerical test shows that this simple treatment does not destroy the accuracy of DG discretization but does guarantee its stability. Note that the treating of taking absolute value to negative solution might lose mass conservation, however it is not important in open systems where the total mass in the computational domain is not a conserved quantity.

Contours of temperature, horizontal velocity and vertical velocity are illustrated in Fig. 4b–d. The white lines with background are the DG-RED solutions, while the red dashed contour lines are the DSMC results in Ref. [14]. Note that the Knudsen number in Ref. [14] is $2(7 - 2\omega)(5 - 2\omega)/15\pi$ times the unconfined Knudsen number in this paper. Comparison between the DG-RED solutions and DSMC ones on the distributions of density, temperature and horizontal velocity along the symmetric line in front of the stagnation point are shown in Fig. 5. It is found that due to the stagnation effect from the static cylinder to the gas
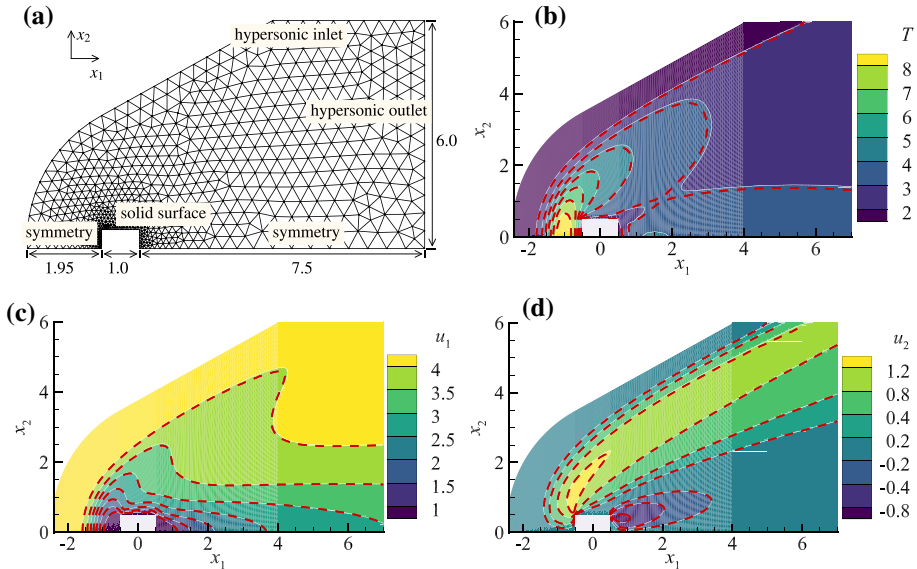
**Fig. 4** Hypersonic flow of $Ma = 5$ and $Kn = 0.13$ past a square cylinder. **a** Schematic for computational domain, boundary condition and unstructured triangular meshes. **b** Temperature contours. **c** Horizontal velocity contours. **d** Vertical velocity contours. The white solid contour lines with background illustrate the solutions from DG-RED of $k = 4$, where the molecular velocity domain $[-13, 13]^3$ is discretized by $48 \times 48 \times 48$ equidistant grid points. The red dashed contour lines are the DSMC results [14] (Color figure online)
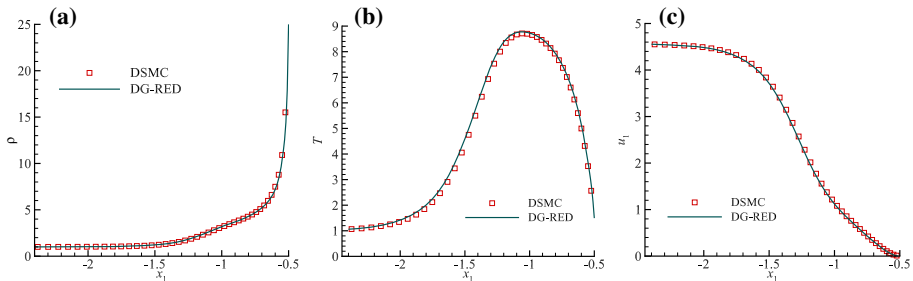


**Fig. 5** Hypersonic flow of $Ma = 5$ and $Kn = 0.13$ past a square cylinder. Profiles of **a** density **b** temperature and **c** horizontal velocity along the symmetric line in front of the stagnation point. Solid lines are the solutions from DG-RED of $k = 4$, where the molecular velocity domain $[-13, 13]^3$ is discretized by $48 \times 48 \times 48$ uniform grid points. Symbols are the DSMC results [14]

flow, the flow density increases about 25 times within 10 (free-streaming) mean free paths when approaching to the cylinder, and the bulk horizontal velocity drops to zero. Since the isothermal wall condition is applied, the flow temperature first increases to its maximum value of 8.7 at about 5 mean free paths away from the stagnation point and then decreases to 1.45 at the solid wall.

Figure 6 illustrates the distributions of normal stress $P_n$ and shear stress $P_t$ along the surfaces of the square cylinder, where $P_n = \boldsymbol{n}_w \cdot \boldsymbol{P} \cdot \boldsymbol{n}_w$ and $P_t = \boldsymbol{n}_w \cdot \boldsymbol{P} \cdot \boldsymbol{t}_w$ with $\boldsymbol{n}_w$ and $\boldsymbol{t}_w$ denoting the outward unit normal vector and tangential vector of the solid surface, respectively. The largest $P_n$ is at the surface in the upwind side where the normal momentum flux is large, while the shear stress gradually increases along that surface as the bulk vertical
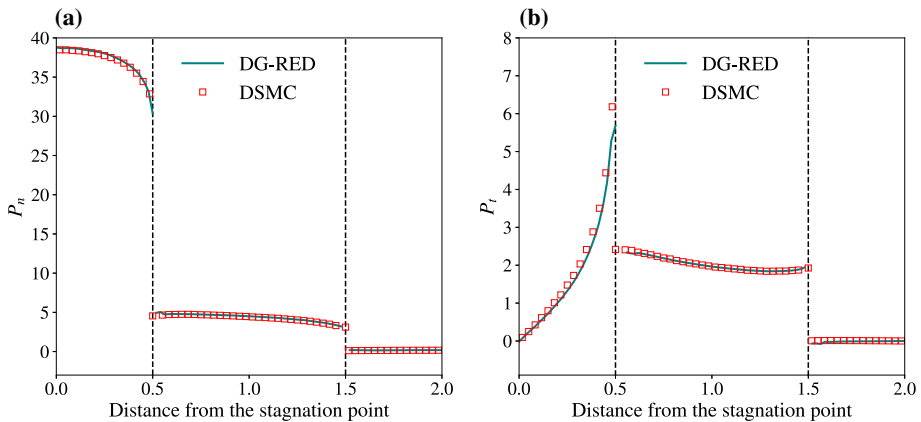
**Fig. 6** Hypersonic flow of $Ma = 5$ and $Kn = 0.13$ past a square cylinder. The distributions of **a** normal stress and **b** shear stress along the surface of cylinder. The horizontal axis represents the distance along the surface of the square, starting from the stagnation point in a counter-clockwise direction. Solid lines are the DG-RED solutions and symbols are the DSMC results [14]

velocity increases. Both $P_n$ and $P_t$ vary slightly along the top surface and the lateral surface in the weak region. Figures 4, 5 and 6 demonstrate the good agreement between the DG and DSMC results.

### 6.3 2D Lid-Driven Cavity Flow

By comparing with the DSMC results, a 2D low-speed flow in a square cavity driven by its top lid is used to compare the performance of DG Boltzmann solvers and a Boltzmann solver using the second-order FDM to approximate the spatial derivative [69]. The wall temperature is set as the reference temperature $T_0 = 273$ K. The speed of the driven lid is 50 m/s. The flow gas is argon with a viscosity index of 0.81. The gas flow is initially stationary at $T_0$ with $Kn = 1$, where the characteristic length $H$ is chosen to be the side length of the square cavity. The computational configuration for DSMC can be found in Ref. [36].

For deterministic solution, the truncated molecular domain is $[-6, 6]^3$. The DG and FDM solvers utilize the same FSM to evaluate collision terms in frequency domain, which is discretized with $32 \times 32 \times 24$ equidistant frequencies. For the discretization of molecular velocity, non-uniform points are used for $v_1$ and $v_2$, while uniform discrete velocities are used in the third direction. The non-uniform discretization with refinement around $v_{1(2)} = 0$ is efficient to calculate low-speed flow especially at large Knudsen number, where the distribution function changes rapidly within a narrow area around the origin in $v_1$ and $v_2$ directions [61]. For spatial discretization, uniform triangular mesh is used in the DG method, as shown in Fig. 7a, while FDM uses equidistant grid points in the $x_1$ and $x_2$ directions. Determination on the numbers of spatial elements and discrete velocities is a trivial task. General speaking, flow with small value of $Kn$ needs relatively large number of spatial elements to ensure that the numerical dissipation is much smaller than the physical viscosity that is small in the near-continuum flow regime, while highly rarefied flow requires a large number of discrete velocities to resolve significant variations and/or discontinuities in the velocity distribution function. Moreover, the spatial and velocity grids have 'contrary' effects, where finite discretization of the velocity space tends to capture discontinuities, whereas limited spatial
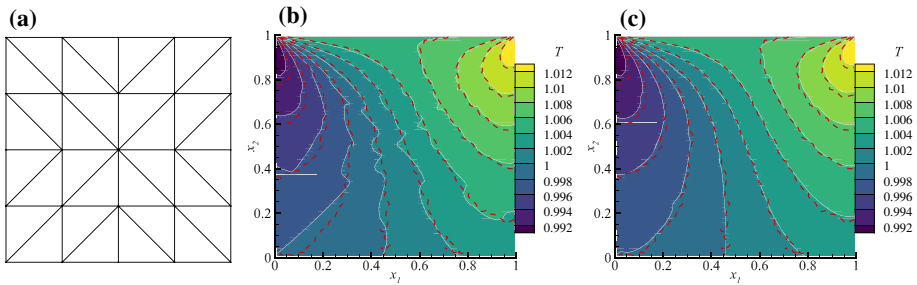
**Fig. 7** Comparison of DG-RED and DSMC on square cavity flow at $Kn = 1.0$ driven by a moving lid with the speed $U_0 = 0.148$. DG-RED solutions are obtain with $k = 4$ on 72 uniform triangles. **a** typical triangular meshes; **b, c** temperature contours when the molecular velocity domain $[-6, 6]^3$ is discretized by $36 \times 36 \times 24$ and $108 \times 108 \times 24$ grid points in DG-RED, respectively. White solid lines with background indicate the solutions of DG-RED, while red dashed lines are the DSMC results. The ITR-LOC scheme (14) is applied in implicit iteration (Color figure online)

discretization tends to smooth flow field due to artificial diffusion. Incompatible spatial and velocity grids can lead to the emergence of 'ray effect', which causes deterministic solution oscillating around its mean value [12,17], see Fig. 7b. To overcome this shortcoming, the velocity grid should be fine enough so that error induced by the ray effect is small, which can be restrained by the numerical dissipation [17].

Temperature contours from the DG-RED scheme of $k = 4$ and $M_{el} = 72$ (highly resolved in the spatial space) are compared with the DSMC results in Fig. 7. Results in Fig. 7b, c are obtained with $36 \times 36 \times 24$ and $108 \times 108 \times 24$ velocities, respectively. It is observed that relative coarser velocity grid produces temperature contour with violent fluctuations, and refinement in the velocity discretization can improve the accuracy significantly. Besides, the DG solver with higher order of approximating polynomial is more likely to suffer from the ray effect. This is mainly due to the fact that, compared to lower-order scheme, higher-order one can obtain more accurate result on same spatial grid so that the numerical dissipation is relatively small and the ray effect becomes more pronounced.

Further comparison on the results of DG-RED and DSMC is illustrated in Fig. 8 in terms of the horizontal (vertical) flow velocity along selected vertical (horizontal) lines. The DG-RED results are in good agreement with those of DSMC.

In Table 3, we list the relative $L_2$ error of velocity magnitude $|\boldsymbol{u}|$, the number of iterations to reach the convergence criterion $\max\{R_T, R_n, R_{\boldsymbol{u}}\} < 10^{-5}$, as well as the total CPU time cost for DG-FULL with ITR-LOC, DG-RED with ITR-LOC and DG-RED with ITR-MEAN. In the ITR-MEAN iterative scheme (13), the mean collision frequency is set as $\bar{\nu} = 1.4$ for this specific flow. For all the cases, the molecular velocity domain $[-6, 6]^3$ is discretized by $72 \times 72 \times 24$ grid points. The errors are calculated in reference to the DSMC results, which are obtained at $60 \times 60$ equidistant points $\boldsymbol{x}_p$ in the computational domain. The errors are evaluated as

$$\mathcal{E} = \sqrt{\sum \left(|\boldsymbol{u}(\boldsymbol{x}_p)|_{\mathrm{DG}} - |\boldsymbol{u}(\boldsymbol{x}_p)|_{\mathrm{DSMC}}\right)^2 / \sum |\boldsymbol{u}(\boldsymbol{x}_p)|_{\mathrm{DSMC}}^2}, \qquad (34)$$

where the DG solution at any point $\boldsymbol{x}_p$ can be easily obtained through polynomial approximation. All the tests are conducted on single processor. It is shown that, for each order of approximating polynomial, the three schemes can produce solution with the same accuracy on the same spatial mesh. The iterative scheme using local collision frequency can obtain the steady-state solution within 21 steps, no matter which DG calculation (DG-FULL or DG-
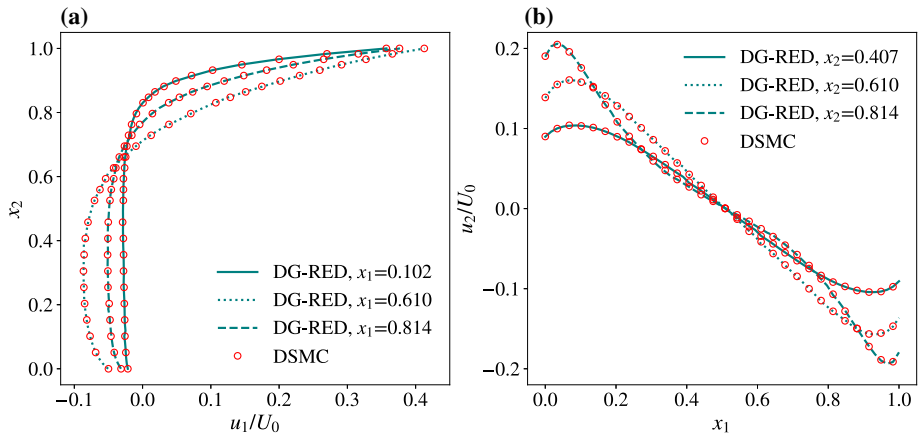
**Fig. 8** Comparison of DG-RED and DSMC on square cavity flow at $Kn = 1.0$ driven by a moving lid with the speed $U_0 = 0.148$. **a** normalized horizontal flow velocity $u_1/U_0$ along vertical lines at different locations; **b** normalized vertical flow velocity $u_2/U_0$ along horizontal lines at different locations. The DG-RED solutions are obtained with $k = 4$ and 72 triangles. The molecular velocity domain $[-6, 6]^3$ is discretized by $108 \times 108 \times 24$ grid points. The ITR-LOC scheme (14) is applied in implicit iteration

RED) is applied. Thus, due to the reduction of computational complexity in calculation of the Boltzmann collision operator, DG-RED cost less CPU time than DG-FULL. Equipped with the chosen mean collision frequency, the ITR-MEAN iterative scheme (13) uses 17 steps to reach the steady-state solution. Since it does not require $LU$-decomposition during iterations, the scheme combining DG-RED and ITR-MEAN can further reduce the computational cost. For example, with $k = 4$ and $M_{el} = 18$, to obtain a solution with the error in velocity magnitude less than 0.014, DG-RED plus ITR-MEAN costs about 50% and 92% less CPU time than that of DG-RED with ITR-LOC and DG-FULL with ITR-LOC, respectively.

We also list the error of velocity magnitude, number of iterations and CPU time for FDM in Table 4. Uniformly distributed points are employed to discretize the spatial space. Thus, the computational domain is partitioned by rectangular elements and flow properties are evaluated at the vertices of rectangles. Note that in the estimation of the error in velocity magnitude, $\boldsymbol{u}(\boldsymbol{x}_p)$ may not be fixed at discrete grid points; then it is obtained through linear interpolation using the four values at vertices of the grid cell where the point $\boldsymbol{x}_p$ is located. The FDM solver also uses 21 steps to obtain steady-state solutions, since the ITR-LOC iterative scheme (14) is employed. For comparison of DG and FDM, we find that the DG discritization is more efficient. For instance, the FDM solver predicts solution with error in $|\boldsymbol{u}|$ of 0.015 on the spatial grid with $71 \times 71$ grid points, while the DG scheme achieves the same level of accuracy on 50 and 18 triangles for $k = 3$ and 4, respectively. However, the DG method with $k = 3$ and full calculation in the collision terms [Eqs. (22), (23)] cost more CPU time than the FDM solver. This is because, although the computational cost for the Boltzmann collision operator in DG-FULL with $k = 3$ and $M_{el} = 50$ ($\propto M_{el}K^2$) and in the FDM with $M_p = 71 \times 71$ ($\propto M_p$) is similar, the DG scheme requires additional time to solve linear equations. As a consequence, only the DG-RED scheme can preserve the efficiency of DG in terms of CPU time. Equipped with the ITR-LOC iteration (14), to obtain solution with error in $|\boldsymbol{u}|$ of 0.015, the DG-RED solvers of $k = 3$ and 4 are about 4 and 7 times faster than the FDM. The ITR-MEAN iteration (13) can further boost its efficiency, where the DG-RED solvers of $k = 3$ and 4 can be 6 and 13 times faster than the FDM.

**Table 3** Comparisons between DG-FULL with the ITR-LOC iteration, and DG-RED with ITR-LOC as well as ITR-MEAN in the lid-driven square cavity flow with $Kn = 1.0$, in terms of the relative $L_2$ error $\mathcal{E}$ (34), the number of iterations (Itr denotes the number of iteration steps to reach the convergence criterion $\max\{R_T, R_\rho, R_{|\boldsymbol{u}|}\} < 10^{-5}$), and the CPU time $t_c$. The molecular velocity domain $[-6, 6]^3$ is discretized by $72 \times 72 \times 24$ grid points

| $k$ | $M_{el}$ | DG-FULL + ITR-LOC | | | DG-RED + ITR-LOC | | | DG-RED + ITR-MEAN | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $\mathcal{E}$ | Itr | $t_c$ (h) | $\mathcal{E}$ | Itr | $t_c$ (h) | $\mathcal{E}$ | Itr | $t_c$ (h) |
| 1 | 32 | 0.102 | 21 | 0.044 | 0.102 | 21 | 0.024 | 0.102 | 17 | 0.018 |
| | 50 | 0.080 | 21 | 0.068 | 0.080 | 21 | 0.038 | 0.080 | 17 | 0.029 |
| | 72 | 0.065 | 21 | 0.100 | 0.065 | 21 | 0.056 | 0.065 | 17 | 0.042 |
| | 98 | 0.054 | 21 | 0.258 | 0.054 | 21 | 0.077 | 0.054 | 17 | 0.059 |
| 2 | 32 | 0.039 | 21 | 0.154 | 0.039 | 21 | 0.058 | 0.039 | 17 | 0.043 |
| | 50 | 0.029 | 21 | 0.244 | 0.029 | 21 | 0.080 | 0.029 | 17 | 0.059 |
| | 72 | 0.023 | 21 | 0.457 | 0.023 | 21 | 0.125 | 0.023 | 17 | 0.088 |
| | 98 | 0.019 | 21 | 0.895 | 0.019 | 21 | 0.173 | 0.019 | 17 | 0.126 |
| 3 | 18 | 0.025 | 21 | 0.222 | 0.025 | 21 | 0.056 | 0.025 | 17 | 0.033 |
| | 32 | 0.019 | 21 | 0.551 | 0.019 | 21 | 0.120 | 0.019 | 17 | 0.082 |
| | 50 | 0.014 | 21 | 0.950 | 0.014 | 21 | 0.180 | 0.014 | 17 | 0.117 |
| | 72 | 0.012 | 21 | 1.078 | 0.012 | 21 | 0.272 | 0.012 | 17 | 0.190 |
| 4 | 8 | 0.024 | 21 | 0.219 | 0.024 | 21 | 0.044 | 0.024 | 17 | 0.023 |
| | 18 | 0.014 | 21 | 0.696 | 0.014 | 21 | 0.111 | 0.014 | 17 | 0.057 |
| | 32 | 0.011 | 21 | 1.321 | 0.011 | 21 | 0.234 | 0.011 | 17 | 0.143 |
| | 50 | 0.008 | 21 | 2.013 | 0.008 | 21 | 0.349 | 0.008 | 17 | 0.224 |

**Table 4** Performance of FDM combining the ITR-LOC iteration (14) for the solution of lid-driven square cavity flow at $Kn = 1.0$

| $M_p$ | $\mathcal{E}$ | Itr | $t_c$ (h) | $M_p$ | $\mathcal{E}$ | Itr | $t_c$ (h) |
|---|---|---|---|---|---|---|---|
| $31^2$ | 0.052 | 21* | 0.159 | $61^2$ | 0.018 | 21 | 0.611 |
| $41^2$ | 0.046 | 21 | 0.282 | $71^2$ | 0.015 | 21 | 0.845 |
| $51^2$ | 0.028 | 21 | 0.433 | $81^2$ | 0.016 | 21 | 1.065 |

$M_p$ is the number of discrete points in the spatial space. $\mathcal{E}$ is the relative $L_2$ error of velocity magnitude $|\boldsymbol{u}|$ compared with the DSMC results. Itr denotes the number of iteration steps to reach the convergence criterion $\max\{R_T, R_\rho, R_{|\boldsymbol{u}|}\} < 10^{-5}$. $t_c$ is the total CPU time. The molecular velocity domain $[-6, 6]^3$ is discretized by $72 \times 72 \times 24$ non-uniform grid points
*In this case, solution only converges to residual of $1.2 \times 10^{-5}$ due to round-off errors

Although higher-order FDM could achieve better efficiency, it needs more computational effort since stencils involving large numbers of points are required. Also, it has difficulty to handle complex geometries.

## 6.4 2D Flow Induced by a Hot Micro-beam in a Rectangular Chamber

We then consider the performance of DG method in the simulation of low-speed rarefied gas flows inside micro-channel. As depicted in Fig. 9a, we consider a 2D rarefied gas flow induced
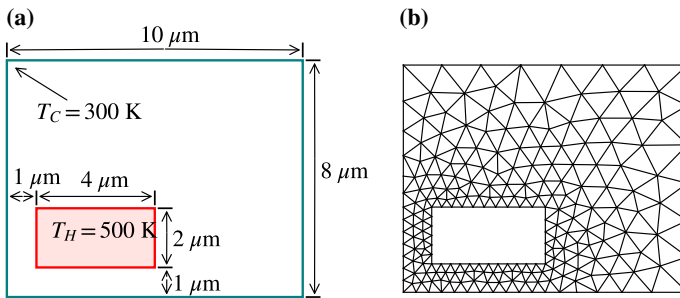
**Fig. 9** The micro gas flow around heated beam in a rectangular chamber: **a** geometry and **b** schematic of the triangular mesh

by a hot micro-beam with a thickness of 2 μm and a width of 4 μm, which is encompassed in a cold rectangular chamber with a dimension of 10 μm × 8 μm and a wall temperature of 500 K. The beam with a temperature of 300 K is placed 1 μm away from the left and bottom walls of the enclosure. Gas is filled between the beam and chamber. Unlike the continuum flow where the flow velocity is zero and the temperature is governed by Fourier's heat conduction law, at rarefied condition, the temperature inhomogeneity induces anisotropic momentum transfer that in turn produces pressure gradient and bulk gas flow. Due to the asymmetric geometry, momentum fluxes impinging on the beam surface are unbalanced, giving rise to a net Knudsen force [47,52], which can be exploited for microstructure actuation and gas sensing [59]. Previous researches have shown that the thermal edge flow occurring near the boundary with sharp corners plays a critical roles in formation of the Knudsen force [58,79].

The DG-RED scheme with $k = 4$ is applied to solve the rarefied gas flow using the ITR-LOC iteration (14). The truncated molecular velocity is set as $[-6, 6]^3$. 96 non-uniform velocity points are used to discretize $v_1$ and $v_2$, while 24 uniform points are used for $v_3$. For evaluation of collision terms, $32 \times 32 \times 24$ equidistant frequencies are employed. Figure 9b illustrates the schematic of the unstructured triangular meshes, where more triangles are placed near the micro-beam. We first consider flows at $Kn = 0.13$, 1.30 and 12.96. The Knudsen numbers are calculated using $T_0 = 400$ K and $H = 1$ μm. The total iterative steps and the CPU time to obtain the steady-state solutions vary with the Knudsen number. For the same spatial and velocity discretization, the smaller the Knudsen number, the more iterative steps thus more CPU time are required. To obtain the solution of $Kn = 12.96$ on 881 triangles, 84 steps are needed to reach the convergence criterion of $\max\{R_T, R_\rho, R_{|u|}\} < 10^{-5}$, which takes 4.3 h of wall time on 12 processors (OpenMP for parallelism).

Figure 10 shows the temperature contours and streamlines. It is observed that noticeable curls that originate at the corners of the beam emerge in the temperature contours at highly rarefied condition ($Kn = 12.96$). However, in small Knudsen number flow, sufficient intermolecular collisions gradually smooth these curls when they propagate to the chamber. When the Knudsen number is small, at each surface of the beam, thermal edge flows drive gas molecules from the corners to the surface centers and form a relatively high pressure region therein. Then, the high pressure promotes gas flowing to the chamber. Due to the confinement of chamber walls, gas molecules finally return to the corners of the beam. Hence, eight localized vortices are observed in the flow field. When $Kn$ increases to 1.30, 3 more vortices are developed with one in the lower-right corner of the chamber and two in the upper-left corner of the chamber. As the degree of gas rarefaction further increases, the vortex in the lower-right corner of the chamber gradually absorbs the localized vortices near the right and
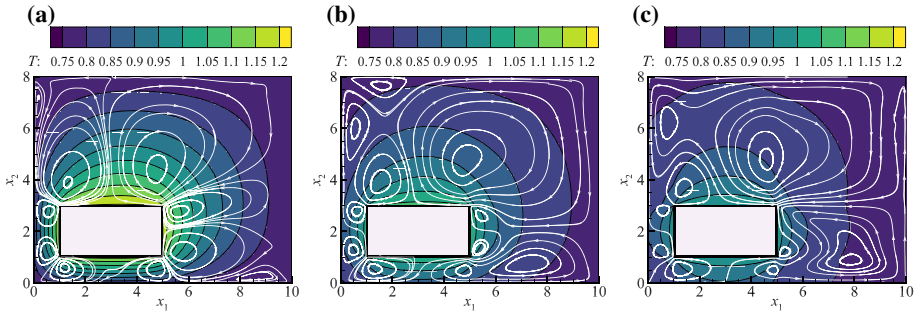
**Fig. 10** Temperature contours and streamlines in micro flow of argon gas with $\omega = 0.81$. **a** $Kn = 0.13$; **b** $Kn = 1.30$; **c** $Kn = 12.96$. The molecular velocity domain $[-6, 6]^3$ is discretized by $96 \times 96 \times 24$ non-uniform grid points. 1290 triangles are used for flows of $Kn = 0.13$ and $Kn = 1.30$, while 881 triangles for case of $Kn = 12.96$
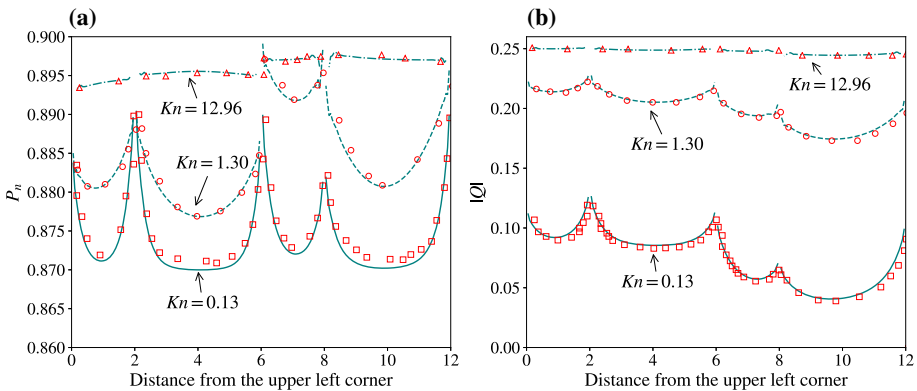


**Fig. 11** **a** Normal stress and **b** magnitude of heat flux along the surface of hot beam. Lines are the DG-RED solutions and symbols are the DSMC result in Ref. [78]. The horizontal axis represents the distance starting from the left-upper corner in a counter-clockwise direction. The gas is argon with $\omega = 0.81$

bottom sides of the beam, which forms a large counter-clockwise vortex. Besides, the vortices in the region above the beam also start to merge together.

Figure 11 illustrates the normal stress (pressure) $P_n$ and the magnitude of heat flux $|\boldsymbol{Q}|$ distributed on the surfaces of the hot beam, where $P_n$ is calculated as $P_n = \boldsymbol{n}_w \cdot \boldsymbol{P} \cdot \boldsymbol{n}_w$ with $\boldsymbol{n}_w$ denoting the outward unit normal vector of the beam surface. The DSMC solutions in Ref. [78] are included for comparison, where good agreement can be observed. It can be seen that the more rarefied flow the larger $P_n$. This is due to the fact that momentum fluxes are enhanced when fewer intermolecular collisions are involved. Moreover, heat transfer is also strengthened by the non-equilibrium effect. The unbalance of $P_n$ on the surfaces mainly contributes to arising of the Knudsen force (the shear stress component is smaller than the normal one by two orders of magnitude). It is observed that $P_n$ on the top (right) surface of the beam is greater than that on the bottom (left) surface, thus both the horizontal and vertical components of the resultant force point to the negative directions of axes.

The resultant force $\mathcal{F}$ acting on the hot beam and total heat $\mathcal{H}$ releasing from the hot beam at $Kn$ ranging from 0.2 to 10 are plotted in Fig. 12, where results for argon with $\omega = 0.81$, hard-sphere molecules with $\omega = 0.5$ and Maxwell molecules with $\omega = 1.0$ are compared.
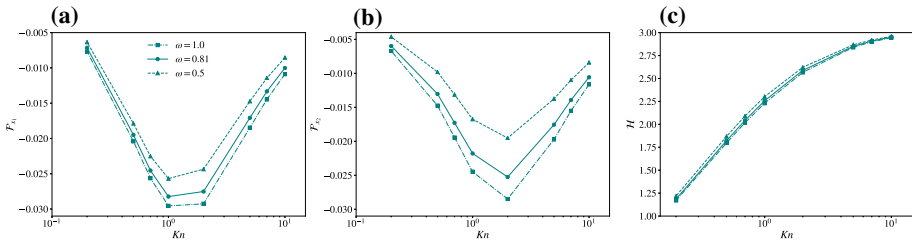
**Fig. 12 a**, **b** Resultant force $[\mathcal{F}_{x_1}, \mathcal{F}_{x_2}]^{\mathrm{T}} = -\oint_{\partial\Omega_b} \boldsymbol{P} \cdot \boldsymbol{n}_w \mathrm{d}\Upsilon$ acting on the hot beam; **c** total heat $\mathcal{H} = \oint_{\partial\Omega_b} \boldsymbol{Q}\mathrm{d}\Upsilon$ releasing by the hot beam. The solutions for argon of $\omega = 0.81$, hard-sphere molecules of $\omega = 0.5$ and Maxwell molecules of $\omega = 1.0$ at $Kn$ ranging from 0.2 to 10 are compared. The DG-RED scheme of $k = 4$ combing with the ITR-LOC iteration (14) is applied. The molecular velocity domian $[-6, 6]^3$ is discretized by $96 \times 96 \times 24$ non-uniform grid points, and 1290 triangles are used for all Knudsen numbers

The force and heat are calculated from as

$$[\mathcal{F}_{x_1}, \mathcal{F}_{x_2}]^{\mathrm{T}} = -\oint_{\partial\Delta_b} \boldsymbol{P} \cdot \boldsymbol{n}_w \mathrm{d}\Upsilon,$$

$$\mathcal{H} = \oint_{\partial\Delta_b} \boldsymbol{Q}\mathrm{d}\Upsilon,$$

(35)

where $\partial\Delta_b$ is the surface of the beam. It is observed that the magnitude of Knudsen force first rises and then falls. The maximum magnitude of Knudsen force occurs around $Kn = 2.0$. The total heat always increases with $Kn$. The variation of Knudsen force can be ascribed to the development and competition of the localized thermal flows described above. When $Kn$ is small, i.e. the non-equilibrium effect is insignificant, the variation of pressure on each beam surface is small with the same magnitude, hence the Knudsen force is weak. As the Knudsen number increases, the strength of local flows are enhanced, and the more spacious spaces on the top and right of the beam allow formations of bigger vortices, which drive more gas molecules from the upper- and lower-right corners of the chamber to the center of the right surface of the beam; thus the pressure there is larger than that near the left beam surface. On the other hand, the counter-clockwise vortex originating from the lower-right corner of the chamber penetrates into the bottom of the beam and efficiently takes gas molecules away from there. This causes the pressure near the bottom surface of the beam to be lower than that on its top surface. Therefore, the magnitudes of the horizontal and vertical components of Knudsen force both become larger. As the Knudsen number increases further, the thermal flows are strengthened even more. The large vortex on the top surface of the beam starts to swallow the small vortices near the upper-left corner of the chamber, while the large vortex at the lower-right corner of the chamber begins to absorb the small vortices on the right surface of the beam. The formations of two giant vortices release some pressure on the top and right surfaces of the beam, thus the magnitude of Knudsen force falls down. The profiles of $\mathcal{F}$ and $\mathcal{H}$ for $\omega = 0.81$ always lie between the ones for $\omega = 0.5$ and $\omega = 1$.

### 6.5 2D Thermal Cavity Flow

In this section, we test the thermal cavity flow induced by temperature gradients at wall and intend to provide accurate result for this flow that may serve as benchmark solution, when the Knudsen numbers are $Kn = 0.1, 0.5$ and $1$.
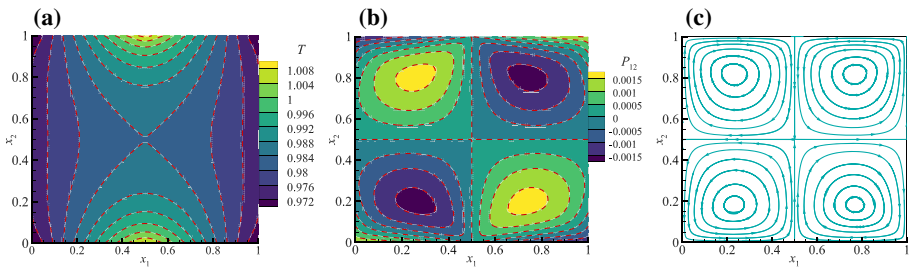
**Fig. 13** Comparison of DG-RED and FDM on the thermal cavity flow induced by the temperature gradients at wall when $Kn = 0.5$. Contours of temperature $T$ and shear stress $P_{12}$ are shown in **a**, **b**, respectively, where the solid lines with background are the solutions of DG-RED, and the red dashed lines are the FDM results. The stream lines are presented in **c**. The DG-RED solutions are obtained with $k = 4$ and 72 uniform triangles. The molecular velocity domain $[-6, 6]^3$ is discretized by $72 \times 72 \times 24$ non-uniform grid points (Color figure online)

The computational domain is a $1 \times 1$ square, partitioned by structured triangular mesh as shown in Fig. 7a. The left and right walls are maintained at constant temperature $T_C$, while the bottom and top walls have varied temperature given by:

$$T (x_1, x_2 = 0 \text{ or } 1) = \begin{cases} 2 (T_H - T_C) x_1 + T_C, & x_1 \leq 0.5, \\ -2 (T_H - T_C) x_1 + 2T_H - T_C, & x_1 > 0.5, \end{cases} \tag{36}$$

where $T_C$ and $T_H$ are set as 263 K and 283 K, respectively.

The argon gas with viscosity index $\omega = 0.81$ is initialized at the reference temperature of $T_0 = 273$ K. For all the cases, the molecular velocity domain is chosen as $[-6, 6]^3$, which is discretized by $72 \times 72$ non-uniform points in the $v_1$ and $v_2$ directions, and 24 uniform points in the $v_3$ direction. The corresponding frequency space, however, is discretized by $32 \times 32 \times 24$ equidistant frequencies for the evaluation of Boltzmann collision operator. For the verification of DG results, FDM results serve as reference solutions. In order to ensure the accuracy of FDM, $201 \times 201$ equidistant grid points are employed for the spatial discretization. Further refinement of both the velocity and spatial girds would only improve the solution by a magnitude no more than 0.5%. The DG-RED scheme with $k = 4$ is used to solve the rarefied gas flow on 72 triangles. The ITR-LOC iteration (14) is applied.

Figure 13 illustrates the dimensionless temperature and shear stress contours, as well as the streamlines for flow at $Kn = 0.5$. The DG-RED steady-state solution presented here costs about 34 iterative steps and 0.96 h CPU time on a single processor. It is observed that high flow temperatures occur near the centers of bottom and top walls due to the heating from walls, while low temperatures appear in the four corners. The tangential temperature gradients near the walls lead to the thermal creep flows, where gas molecules along the bottom and top walls move from the colder regions to the hotter ones. Due to the confinement of vertical walls, 4 vortexes are generated: the two at the lower left and upper right quarters rotate counter-clockwise and the other two rotate clockwise. As a consequence, the maximum shear stresses appear at the centers of clockwise vortices, while the minimum ones occur at the centers of counter-clockwise vortices. The flow patterns at $Kn = 0.1$ and $Kn = 1.0$ are similar. The results for $Kn = 0.1$ and $Kn = 1.0$ costs about 140 and 22 iterative steps, respectively, and the CPU time on a single processor is about 4.89 h and 0.63 h. It is worth mentioning that the same case of $Kn = 1.0$ was calculated by a recently developed explicit DG fast spectral Boltzmann solver, where the solutions obtained on 16 spatial elements and $24 \times 24 \times 24$ uniform velocity gird with $k = 2$ cost more than 15 h on a single processor
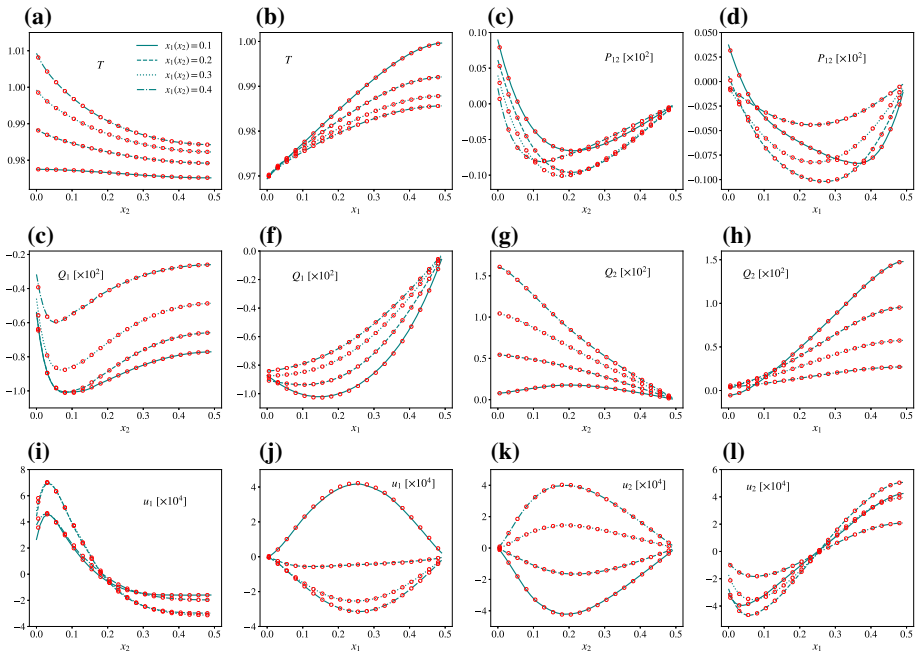
**Fig. 14** Comparison of DG-RED (solid lines) and FDM (circles) on thermal cavity flow induced by temperature gradients at wall when $Kn = 0.1$. The first and third columns are flow properties along vertical lines at $x_1 = 0.1, 0.2, 0.3$ and $0.4$, while the second and forth columns are flow properties along horizontal lines at $x_2 = 0.1, 0.2, 0.3$ and $0.4$. The DG-RED solutions are obtained with $k = 4$ and 72 uniform triangles

[35]. With the same numbers of spatial elements and discrete velocities, as well as the same order of approximating polynomial, our scheme only costs 14 seconds to obtained the steady-state solution; thus our scheme is faster by three orders of magnitude. The efficiency of the proposed scheme enables more complex calculations in real engineering problems.

Figure 14 illustrates the variations of temperature $T$, shear stress $P_{12}$, horizontal (vertical) heat fluxes $Q_1$ ($Q_2$) and horizontal (vertical) flow velocities $u_1$ ($u_2$) along selected horizontal and vertical lines for rarefied gas flow when $Kn = 0.1$; those for $Kn = 0.5$ and $Kn = 1$ are plotted in Figs. 15 and 16, respectively. Due to the symmetry of flow field, the results are only shown within the lower left quarter of the computational domain. It is found that from the regions near solid walls to the flow field center, the gas temperature increases along the horizontal lines, while decreases along the vertical lines. However, along both the horizontal and vertical directions, the shear stress first drops to the local minimum values then rises back to zero. The variations of horizontal heat flux are similar as those of shear stress, while the changes of the vertical component of heat flux are in accordance with the variations of gas temperature. The changes of bulk flow velocity are more complicated. Along the vertical lines, the horizontal velocity $u_1$ first increases to the local peaked values and then falls to the minimums. Along the horizontal lines near the bottom wall, $u_1$ is positive and has a local maximum at $x_1 = 0.25$, while in the regions away from the bottom wall, $u_1$ becomes negative and has a local minimum at $x_1 = 0.25$. Similarly, near the left lateral wall, the vertical velocity $u_2$ is negative and gradually changes its sign and reaches the local maximal values when approaching to the field center along the horizontal lines. For all the flow properties, agreement between the DG-RED and FDM results is good. It is also interesting to note that,
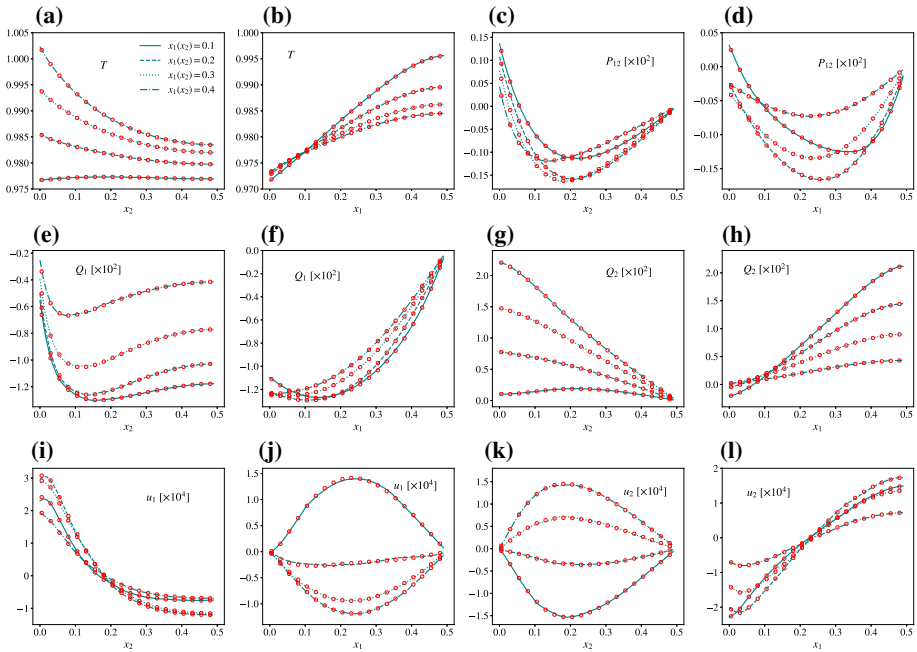
**Fig. 15** Comparison of DG-RED (solid lines) and FDM (circles) on thermal cavity flow induced by temperature gradients at wall when $Kn = 0.5$. The first and third columns are flow properties along vertical lines at $x_1 = 0.1, 0.2, 0.3$ and $0.4$, while the second and forth columns are flow properties along horizontal lines at $x_2 = 0.1, 0.2, 0.3$ and $0.4$. The DG-RED solutions are obtained with $k = 4$ and 72 uniform triangles

as the degree of rarefaction increases, the maximum values of temperature decrease since the intensity of gas-gas/gas-wall interactions becomes weaker. On the other hand, the maximum heat flux $|\boldsymbol{Q}|$ occurring near the centers of the bottom and top walls becomes larger, due to the larger temperature jump in highly rarefied gas.

# 7 Conclusions

In summary, we have developed a high-order discontinuous Galerkin discretization to solve the Boltzmann equation with full collision operator. The proposed numerical scheme is based on the classical discrete velocity method. At each discrete velocity grid points, the velocity distribution function is approximated in piecewise polynomial spaces of degree up to 4 in the spatial space. Concerning resolution of the Boltzmann collision operator, we rely on the Carleman-representation-based Fourier techniques, which can preserve mass, momentum and energy with spectral accuracy. Due to incorporation of the DG discretization and the fast spectral method, the computational cost to evaluate the Boltzmann collision operator is $O\left(K^2 M_{\text{el}} M_{\text{qua}}^2 \bar{N}^3 \log \bar{N} + 2K^3 M_{\text{el}} \bar{N}^3\right)$, which can be significantly large when high order approximating polynomial is used. Based on the nodal DG approximating, we have proposed a reduced DG discretization for the collision operator, which can reduce the computational cost by $K$ times of magnitude.
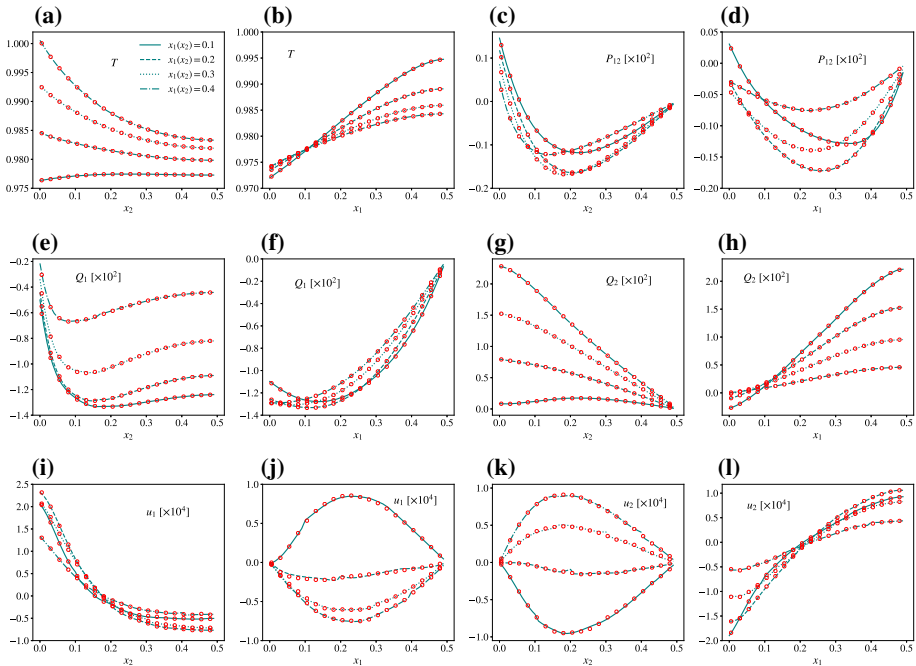
**Fig. 16** Comparison of DG-RED (solid lines) and FDM (circles) on thermal cavity flow induced by temperature gradients at wall when $Kn = 1.0$. The first and third columns are flow properties along vertical lines at $x_1 = 0.1, 0.2, 0.3$ and $0.4$, while the second and forth columns are flow properties along horizontal lines at $x_2 = 0.1, 0.2, 0.3$ and $0.4$. The DG-RED solutions are obtained with $k = 4$ and 72 uniform triangles

Implicit iterative scheme scheme for streaming operator is employed to find the steady-state solution. At each iterative step, the DG discretization results in a system of linear equations for the degrees of freedom of velocity distribution functions on each spatial element. Since the first-order upwind principle is applied to approximate fluxes on the faces of spatial elements, the local linear equations only couple the unknowns on the immediate neighboring elements in the upwind side. Due to the fact that the direction of molecular velocity is fixed after discretization in the molecular velocity space, we have successfully employed the sweeping technique to solve the local linear systems sequentially. This strategy avoids solving large sparse linear systems for all the elements that consumes large memory and CPU time when a large number of discrete velocities are required.

Five different test cases including hypersonic flows, as well as shear-driven and thermal-driven low-speed flows have been presented to show accuracy and capability of the proposed method. Several conclusions are summarized through the performance analysis:

– The implicit iterative scheme has no restriction on time step imposed by the CFL condition. The DG schemes with different order of approximating polynomials can obtain steady-state solution of the same order of accuracy within the same number of iterative steps. Therefore, higher-order discretization needs fewer spatial elements and less CPU time.
– Compared to the full DG discretization in the collision operator, the proposed reduced DG approximation preserves the accuracy of the numerical scheme even for highly non-equilibrium flow, but significantly reduces the computational cost. To obtain the results

with the same order of accuracy, the higher degree of approximation polynomial, the more the saving of CPU time in the reduced DG approximation.

– Based on the same fast spectral method for the approximation of the Boltzmann collision operator, comparison with the finite difference method shows that the DG discretization is more efficient. Equipped with the implicit iterative scheme involving global mean collision frequency, the DG scheme can be faster than the finite difference method by one order of magnitude.

– The implicit iterative scheme combining with the sweeping technique to sequentially solve the local linear systems on each spatial element preserves the stability of DG scheme. In simulating rarefied gas flow, the shock wave structure is resolved by well refined spatial mesh so that the variation of solution within a grid cell is mild. In addition, absolute value is taken to the negative solution of velocity distribution function to ensure the error does not accumulate. Due to the sweeping technique, the error induced by the DG discretization is localized and will be suppressed during iteration. As a consequence, the proposed method can solve hypersonic flow without any complex limiter.

The developed numerical method is to be extended for the simulation of rarefied gas mixtures straigtforwardly [74], where the velocity distribution function for each species is governed by its own Boltzmann equation. The Boltzmann equations for all constituents are coupled through pairwise collision operators. Thus, the computational complexity in resolving the collision terms via FSM significantly increases as the number of gas species increases. In such situation, the advantage of using implicit DG method as well as the reduced calculation in collision operators will become more pronounced. Moreover, by incorporating more realistic intermolecular potentials such as the Lennard-Jones potential or even the *ab initio* potential based on quantum scattering [56], the developed scheme is ready to simulate a wide range of rarefied gas problems with a high level of accuracy.

## Compliance with Ethical Standards

**Conflict of interest** The authors declare that they have no conflict of interest.

## Appendix

Here, we present details of the DG formulation for the Boltzmann equation. The linear systems (25) to determine the solution of $f^{j'}$ on spatial element $\Delta_i$ are recalled here:

$$\mathbf{A}^{i,j'}\mathbf{F}_i^{j'} + \mathbf{B}^{\text{ext},j'} = \mathbf{S}^{i,j'}, \tag{A.37}$$

for $i = 1, \ldots, M_{\text{el}}, \ j' = 1, \ldots, M$.

We denote $F_r^{j'} = F_r(\boldsymbol{v}^{j'})$, $\Lambda_p^{j'} = \Lambda_p(\boldsymbol{v}^{j'})$ and $\Xi_{p,r}^{j'} = \Xi_{p,r}(\boldsymbol{v}^{j'})$ as values of the corresponding variables at each discrete velocity point, and $\mathbf{F}_i^{j'} = [F_1^{j'}, \ldots, F_r^{j'}, \ldots]^{\text{T}}$ is the vector of degrees of freedom of $f^{j'}$ on $\Delta_i$. For the ITR-LOC scheme, the coefficient matrices are:

$$\mathbf{A}_{sr}^{i,j'} = \frac{1}{2}\left(\boldsymbol{v}^{j'}\cdot\boldsymbol{n} + |\boldsymbol{v}^{j'}\cdot\boldsymbol{n}|\right)\langle\varphi_s,\varphi_r\rangle_{\partial\Delta_i} - \left(\boldsymbol{v}^{j'}\cdot\nabla\varphi_s,\varphi_r\right)_{\Delta_i} + \sum_{p=1}^{K}\left(\varphi_s,\varphi_p\varphi_r\right)_{\Delta_i}\Lambda_p^{j'},$$

(A.38)

$$\mathbf{B}_s^{\text{ext},j'} = \begin{cases} \frac{1}{2}\left(\boldsymbol{v}^{j'}\cdot\boldsymbol{n} - |\boldsymbol{v}^{j'}\cdot\boldsymbol{n}|\right)\sum_{r=1}^{K}\langle\varphi_s,\varphi_r^{\text{ext}}\rangle_{\partial\Delta_i}F_{r,\text{ext}}^{j'}, & \partial\Delta_i \not\subset \partial\Delta \\ \frac{1}{2}\left(\boldsymbol{v}^{j'}\cdot\boldsymbol{n} - |\boldsymbol{v}^{j'}\cdot\boldsymbol{n}|\right)\langle\varphi_s,b^{j'}\rangle_{\partial\Delta_i}, & \partial\Delta_i \subset \partial\Delta \end{cases} , \quad (A.39)$$

$$\mathbf{S}_s = \sum_{p=1}^{K}\sum_{r=1}^{K}\left(\varphi_s,\varphi_p\varphi_r\right)_{\Delta_i}\varXi_{p,r}^{j'},$$

(A.40)

where $\varphi_r^{\text{ext}}$ denotes the supporting polynomials on the neighboring element, from which $f_{\text{ext}}$ is obtained. For the ITR-MEAN scheme, the coefficient matrices become:

$$\mathbf{A}_{sr}^{i,j'} = \frac{1}{2}\left(\boldsymbol{v}^{j'}\cdot\boldsymbol{n} + |\boldsymbol{v}^{j'}\cdot\boldsymbol{n}|\right)\langle\varphi_s,\varphi_r\rangle_{\partial\Delta_i} - \left(\boldsymbol{v}^{j'}\cdot\nabla\varphi_s,\varphi_r\right)_{\Delta_i} + \left(\varphi_s,\varphi_r\right)_{\Delta_i}\bar{v},$$

(A.41)

$$\mathbf{B}_s^{\text{ext},j'} = \begin{cases} \frac{1}{2}\left(\boldsymbol{v}^{j'}\cdot\boldsymbol{n} - |\boldsymbol{v}^{j'}\cdot\boldsymbol{n}|\right)\sum_{r=1}^{K}\langle\varphi_s,\varphi_r^{\text{ext}}\rangle_{\partial\Delta_i}F_{r,\text{ext}}^{j'}, & \partial\Delta_i \not\subset \partial\Delta \\ \frac{1}{2}\left(\boldsymbol{v}^{j'}\cdot\boldsymbol{n} - |\boldsymbol{v}^{j'}\cdot\boldsymbol{n}|\right)\langle\varphi_s,b^{j'}\rangle_{\partial\Delta_i}, & \partial\Delta_i \subset \partial\Delta \end{cases} , \quad (A.42)$$

$$\mathbf{S}_s = \sum_{p=1}^{K}\sum_{r=1}^{K}\left(\varphi_s,\varphi_p\varphi_r\right)_{\Delta_i}\left(\varXi_{p,r}^{j'} - \Lambda_p^{j'}F_r^{j'}\right) + \sum_{r=1}^{K}\left(\varphi_s,\varphi_r\right)_{\Delta_i}\bar{v}F_r^{j'}.$$

(A.43)

In this paper, nodal shape functions are chosen as the approximating polynomials. Integrals of the shape functions such as $(\varphi_s,\varphi_r)$, $(\nabla\varphi_s,\varphi_r)$, $(\varphi_s,\varphi_p\varphi_r)$ and $\langle\varphi_s,\varphi_r\rangle$ can be obtained analytically. To evaluate $\langle\varphi_s,b^{j'}\rangle$, the Gaussian rule is applied.

# References

1. Alekseenko, A., Nguyen, T., Wood, A.: A deterministic-stochastic method for computing the Boltzmann collision integral in $\mathcal{O}(mn)$ operations. Kinetic Relat. Models **11**(5), 1211–1234 (2018)
2. Alsmeyer, H.: Density profiles in argon and nitrogen shock waves measured by the absorption of an electron beam. J. Fluid Mech. **74**(3), 497–513 (1976)
3. Aristov, V.V.: Direct Methods for Solving the Boltzmann Equation and Study of Nonequilibrium Flows. Springer, Dordrecht (2001)
4. Baker, L.L., Hadjiconstantinou, N.G.: Variance-reduced Monte Carlo solutions of the Boltzmann equation for low-speed gas flows: a discontinuous Galerkin formulation. Int. J. Numer. Methods Fluids **58**(4), 381–402 (2008)
5. Bhatnagar, P.L., Gross, E.P., Krook, M.: A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. Phys. Rev. **94**, 511–525 (1954)
6. Bird, G.A.: Molecular Gas Dynamics and the Direct Simulation. Clarendon, Oxford (1994)
7. Bobylev, A.: The theory of the nonlinear spatially uniform Boltzmann equation for Maxwell molecules. Math. Phys. Rev. **7**, 111–233 (1988)
8. Bobylev, A., Palczewski, A., Schneider, J.: On approximation of the Boltzmann equation by discrete velocity models. C. R. Acad. Sci. **320**, 639–644 (1995)
9. Bobylev, A., Rjasanow, R.: Difference scheme for the Boltzmann equation based on fast Fourier transformation. Eur. J. Mech. B Fluids **16**(2), 293–306 (1997)
10. Bobylev, A., Rjasanow, S.: Fast deterministic method of solving the Boltzmann equation for hard spheres. Eur. J. Mech. B Fluids **18**(5), 869–887 (1999)
11. Buet, C.: A discrete-velocity scheme for the Boltzmann operator of rarefied gas dynamics. Transp. Theory Stat. Phys. **25**(1), 33–60 (1996)

12. Chai, J.C., Lee, H.S., Patankar, S.V.: Ray effect and false scattering in the discrete ordinates method. Numer. Heat Transf. Part B Fund. **24**(4), 373–389 (1993)
13. Chapman, S., Cowling, T.: The Mathematical Theory of Non-uniform Gases, 3rd edn. Cambridge University Press, New York (1970)
14. Chen, S., Zhang, C., Zhu, L., Guo, Z.: A unified implicit scheme for kinetic model equations. Part I. Memory reduction technique. Sci. Bull. **62**(2), 119–129 (2017)
15. Cockburn, B., Shu, C.W.: The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. J. Comput. Phys. **141**(2), 199–224 (1998)
16. Cockburn, B., Shu, C.W.: Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. J. Sci. Comput. **16**(3), 173–261 (2001)
17. Coelho, P.: The role of ray effects and false scattering on the accuracy of the standard and modified discrete ordinates methods. J. Quant. Spectrosc. Radiat. Transf. **73**(2), 231–238 (2002)
18. Crivellini, A., Bassi, F.: An implicit matrix-free discontinuous Galerkin solver for viscous and turbulent aerodynamic simulations. Comput. Fluids **50**(1), 81–93 (2011)
19. Crouseilles, N., Mehrenberger, M., Sonnendrücker, E.: Conservative semi-Lagrangian schemes for Vlasov equations. J. Comput. Phys. **229**(6), 1927–1953 (2010)
20. Crouseilles, N., Respaud, T., Sonnendrücker, E.: A forward semi-Lagrangian method for the numerical solution of the Vlasov equation. Comput. Phys. Commun. **180**(10), 1730–1745 (2009)
21. Dimarco, G., Loubere, R.: Towards an ultra efficient kinetic scheme. Part I: basics on the BGK equation. J. Comput. Phys. **255**, 680–698 (2013)
22. Dimarco, G., Loubere, R.: Towards an ultra efficient kinetic scheme. Part II: the high order case. J. Comput. Phys. **255**, 699–719 (2013)
23. Dimarco, G., Loubère, R., Narski, J., Rey, T.: An efficient numerical method for solving the Boltzmann equation in multidimensions. J. Comput. Phys. **353**, 46–81 (2018)
24. Dimarco, G., Pareschi, L.: Numerical methods for kinetic equations. Acta Numer. **23**, 369–520 (2014)
25. Filbet, F., Mouhot, C., Pareschi, L.: Solving the Boltzmann equation in $n \log^2 n$. SIAM J. Sci. Comput. **28**(3), 1029–1053 (2006)
26. Filbet, F., Russo, G.: High order numerical methods for the space non-homogeneous Boltzmann equation. J. Comput. Phys. **186**(2), 457–480 (2003)
27. Gobbert, M., Webster, S., Cale, T.: A Galerkin method for the simulation of the transient 2-D/2-D and 3-D/3-D linear Boltzmann equation. J. Sci. Comput. **30**(2), 237–273 (2007)
28. Goldstein, D., Sturtevant, B., Broadwell, J.E.: Investigations of the motion of discrete-velocity gases. Prog. Astron. Aeron. **117**, 100–117 (1989)
29. Guo, Z., Wang, R., Xu, K.: Discrete unified gas kinetic scheme for all Knudsen number flows. II. Thermal compressible case. Phys. Rev. E **91**(3), 033313 (2015)
30. Guo, Z., Xu, K., Wang, R.: Discrete unified gas kinetic scheme for all Knudsen number flows: low-speed isothermal case. Phys. Rev. E **88**, 033305 (2013)
31. Güçlü, Y., Hitchon, W.: A high order cell-centered semi-Lagrangian scheme for multi-dimensional kinetic simulations of neutral gas flows. J. Comput. Phys. **231**(8), 3289–3316 (2012)
32. Holway, L.H.: New statistical models for kinetic theory: methods of construction. Phys. Fluids **9**(9), 1658–1673 (1966)
33. Huang, A.B., Giddens, D.P.: The discrete ordinate method for the linearized boundary value problems in kinetic theory of gases. In: Brundin C.L. (ed.) Rarefied Gas Dynamics, vol. 1, p. 481 (1967)
34. Ibragimov, I., Rjasanow, S.: Numerical solution of the Boltzmann equation on the uniform grid. Computing **69**(2), 163–186 (2002)
35. Jaiswal, S., Alexeenko, A.A., Hu, J.: A discontinuous Galerkin fast spectral method for the full Boltzmann equation with general collision kernels. J. Comput. Phys. **378**, 178–208 (2019)
36. John, B., Gu, X.J., Emerson, D.R.: Investigation of heat and mass transfer in a lid-driven cavity under nonequilibrium flow conditions. Numer. Heat Transf. Part B Fund. **58**(5), 287–303 (2010)
37. Kitzler, G., Schöberl, J.: A high order space-momentum discontinuous Galerkin method for the Boltzmann equation. Comput. Math. Appl. **70**(7), 1539–1554 (2015)
38. Kolobov, V., Arslanbekov, R., Aristov, V., Frolova, A., Zabelok, S.: Unified solver for rarefied and continuum flows with adaptive mesh and algorithm refinement. J. Comput. Phys. **223**(2), 589–608 (2007)
39. Kosuge, S., Aoki, K., Takata, S.: Shock-wave structure for a binary gas mixture: finite-difference analysis of the Boltzmann equation for hard-sphere molecules. Eur. J. Mech. B Fluids **20**(1), 87–126 (2001)
40. Kubatko, E.J., Dawson, C., Westerink, J.J.: Time step restrictions for Runge–Kutta discontinuous Galerkin methods on triangular grids. J. Comput. Phys. **227**(23), 9697–9710 (2008)
41. Lewis, E., Miller, W.: Computational Methods of Neutron Transport. Wiley, New York (1984)
42. Liu, C., Xu, K., Sun, Q., Cai, Q.: A unified gas-kinetic scheme for continuum and rarefied flows IV: full Boltzmann and model equations. J. Comput. Phys. **314**, 305–340 (2016)

43. Mieussens, L.: Discrete-velocity models and numerical schemes for the Boltzmann-BGK equation in plane and axisymmetric geometries. J. Comput. Phys. **162**(2), 429–466 (2000)
44. Morris, A.B., Varghese, P.L., Goldstein, D.B.: Improvement of a discrete velocity Boltzmann equation solver with arbitrary post-collision velocities. AIP Conf. Proc. **1084**(1), 458–463 (2008)
45. Mouhot, C., Pareschi, L.: Fast algorithms for computing the Boltzmann collision operator. Math. Comput. **75**(256), 1833–1852 (2006)
46. Murphy, S.: Methods for solving discontinuous-Galerkin finite element equations with application to neutron transport. Ph.D. Thesis, Institute National Polytechnique de Toulouse (2015)
47. Nabeth, J., Chigullapalli, S., Alexeenko, A.A.: Quantifying the Knudsen force on heated microbeams: a compact model and direct comparison with measurements. Phys. Rev. E **83**, 066306 (2011)
48. Ohwada, T.: Structure of normal shock waves: direct numerical analysis of the Boltzmann equation for hard-sphere molecules. Phys. Fluids A Fluid Dyn. **5**(1), 217–234 (1993)
49. Ohwada, T.: Heat flow and temperature and density distributions in a rarefied gas between parallel plates with different temperatures. Finite-difference analysis of the nonlinear Boltzmann equation for hard-sphere molecules. Phys. Fluids **8**(8), 2153–2160 (1996)
50. Pareschi, L., Perthame, B.: A Fourier spectral method for homogeneous Boltzmann equations. Transp. Theory Stat. Phys. **25**(3–5), 369–382 (1996)
51. Pareschi, L., Russo, G.: Numerical solution of the Boltzmann equation I: spectrally accurate approximation of the collision operator. SIAM J. Numer. Anal. **37**(4), 1217–1245 (2000)
52. Passian, A., Warmack, R.J., Ferrell, T.L., Thundat, T.: Thermal transpiration at the microscale: a crookes cantilever. Phys. Rev. Lett. **90**, 124503 (2003)
53. Radtke, G.A., Hadjiconstantinou, N.G., Wagner, W.: Low-noise Monte Carlo simulation of the variable hard sphere gas. Phys. Fluids **23**(3), 030606 (2011)
54. Reed, W.H., Hill, T.R.: Triangular mesh methods for the neutron transport equation. Technical Report, vol. 836 (1973)
55. Shakhov, E.: Generalization of the Krook kinetic relaxation equation. Fluid Dyn. **3**(5), 95–96 (1968)
56. Sharipov, F.: Modeling of transport phenomena in gases based on quantum scattering. Phys. A Stat. Mech. Appl. **508**, 797–805 (2018)
57. Sone, Y., Ohwada, T., Aoki, K.: Temperature jump and Knudsen layer in a rarefied gas over a plane wall: numerical analysis of the linearized Boltzmann equation for hard-sphere molecules. Phys. Fluids A Fluid Dyn. **1**(2), 363–370 (1989)
58. Sone, Y., Yoshimoto, M.: Demonstration of a rarefied gas flow induced near the edge of a uniformly heated plate. Phys. Fluids **9**(11), 3530–3534 (1997)
59. Strongrich, A., Alexeenko, A.: Microstructure actuation and gas sensing by the Knudsen thermal force. Appl. Phys. Lett. **107**(19), 193508 (2015)
60. Su, W., Alexeenko, A.A., Cai, G.: A parallel Runge–Kutta discontinuous Galerkin solver for rarefied gas flows based on 2D Boltzmann kinetic equations. Comput. Fluids **109**, 123–136 (2015)
61. Su, W., Lindsay, S., Liu, H., Wu, L.: Comparative study of the discrete velocity and lattice Boltzmann methods for rarefied gas flows through irregular channels. Phys. Rev. E **96**, 023309 (2017)
62. Tang, Z., He, B., Cai, G.: Investigation on a coupled Navier–Stokes direct simulation Monte Carlo method for the simulation of plume flowfield of a conical nozzle. Int. J. Numer. Methods Fluids **76**(2), 95–108 (2014)
63. Tcheremissine, F.: Conservative evaluation of Boltzmann collision integral in discrete ordinates approximation. Comput. Math. Appl. **35**(1), 215–221 (1998)
64. Tcheremissine, F.G.: Solution to the Boltzmann kinetic equation for high-speed flows. Comput. Math. Math. Phys. **46**(2), 315–329 (2006)
65. Titarev, V.A.: Efficient deterministic modelling of three-dimensional rarefied gas flows. Commun. Comput. Phys. **12**(1), 162–192 (2012)
66. Valentini, P., Schwartzentruber, T.E.: Large-scale molecular dynamics simulations of normal shock waves in dilute argon. Phys. Fluids **21**(6), 066101 (2009)
67. Wagner, W.: Approximation of the Boltzmann equation by discrete velocity models. J. Stat. Phys. **78**(5), 1555–1570 (1995)
68. Watchararuangwita, C., Grigorievb, Y.N., Meleshkoa, S.V.: A deterministic spectral method for solving the Boltzmann equation for one-dimensional flows. Sci. Asia **35**(1), 70–79 (2009)
69. Wu, L., Liu, H., Zhang, Y., Reese, J.M.: Influence of intermolecular potentials on rarefied gas flows: fast spectral solutions of the Boltzmann equation. Phys. Fluids **27**(8), 082002 (2015)
70. Wu, L., Reese, J.M., Zhang, Y.: Solving the Boltzmann equation deterministically by the fast spectral method: application to gas microflows. J. Fluid Mech. **746**, 53–84 (2014)
71. Wu, L., Struchtrup, H.: Assessment and development of the gas kinetic boundary condition for the Boltzmann equation. J. Fluid Mech. **823**, 511–537 (2017)

72. Wu, L., White, C., Scanlon, T.J., Reese, J.M., Zhang, Y.: Deterministic numerical solutions of the Boltzmann equation using the fast spectral method. J. Comput. Phys. **250**, 27–52 (2013)
73. Wu, L., Zhang, J., Liu, H., Zhang, Y., Reese, J.M.: A fast iterative scheme for the linearized Boltzmann equation. J. Comput. Phys. **338**, 431–451 (2017)
74. Wu, L., Zhang, J., Reese, J.M., Zhang, Y.: A fast spectral method for the Boltzmann equation for monatomic gas mixtures. J. Comput. Phys. **298**, 602–621 (2015)
75. Xu, K., Huang, J.C.: A unified gas-kinetic scheme for continuum and rarefied flows. J. Comput. Phys. **229**(20), 7747–7764 (2010)
76. Yang, J., Huang, J.: Rarefied flow computations using nonlinear model Boltzmann equations. J. Comput. Phys. **120**(2), 323–339 (1995)
77. Yen, S.: Temperature overshoot in shock waves. Phys. Fluids **9**(7), 1417–1418 (1966)
78. Zhu, L., Guo, Z.: Application of discrete unified gas kinetic scheme to thermally induced nonequilibrium flows. Comput. Fluids. **193**, 103613 (2019)
79. Zhu, T., Ye, W.: Origin of Knudsen forces on heated microbeams. Phys. Rev. E **82**, 036308 (2010)
80. Zhu, Y., Zhong, C., Xu, K.: Implicit unified gas-kinetic scheme for steady state solutions in all flow regimes. J. Comput. Phys. **315**, 16–38 (2016)