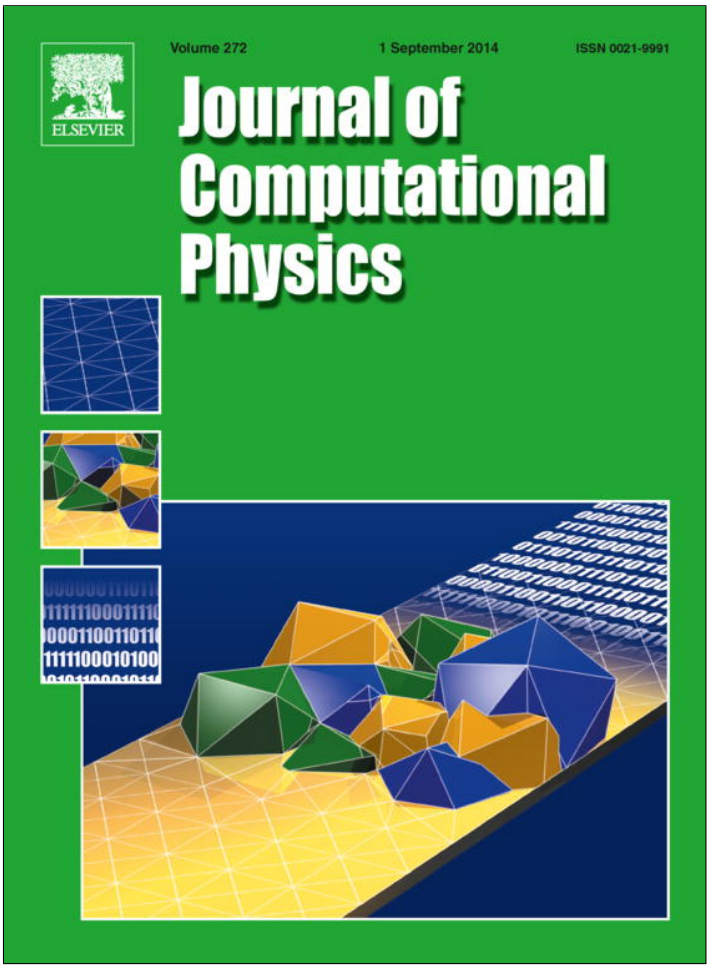

Journal of Computational Physics 272 (2014) 170–188

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Deterministic solution of the spatially homogeneous Boltzmann equation using discontinuous Galerkin discretizations in the velocity space

A. Alekseenko [a,b,*], E. Josyula [b]

[a] Department of Mathematics, California State University Northridge, Northridge, CA, USA
[b] Air Force Research Laboratory, Wright-Patterson AFB, OH, USA

ARTICLE INFO

Article history:
Received 1 January 2013
Received in revised form 11 March 2014
Accepted 14 March 2014
Available online 31 March 2014

Keywords:
Boltzmann collision operator
Deterministic solution
Discontinuous Galerkin methods

ABSTRACT

We present a new deterministic approach for the solution of the spatially homogeneous Boltzmann kinetic equation based on nodal discontinuous Galerkin (DG) discretizations in the velocity space. In the new approach the collision operator has the form of a bilinear operator with a pre-computed kernel; its evaluation requires $O(n^5)$ operations at every point of the phase space where n is the number of degrees of freedom in one velocity dimension. The method is generalized to any molecular potential. Results of numerical simulations are presented for the problem of spatially homogeneous relaxation for the hard spheres potential. Comparison with the method of Direct Simulation Monte Carlo showed excellent agreement.

1. Introduction

Being central to gas dynamics the Boltzmann equation has the capacity to describe gas flows in regimes from continuum to rarefied. Its descriptive power is derived from the microscopic probabilistic representation of gas by a space and time dependent velocity distribution function of a large collection of particles. The particles interact according to known potentials producing a change in the distribution function that is modeled by the five-fold (three velocity and two spatial integrals) Boltzmann collision integral. The Boltzmann equation is one of the most intensely studied subjects over the last fifty years. Analytic solutions to this equation have been constructed for simple geometries and in special cases of molecular potential. However, the complexity of the equation suggests that solutions to applications arising in engineering and physics with complex boundaries and complex gas-to-gas and gas-to-surface interactions can only be computed approximately, using numerical techniques. The costs associated with the direct evaluation of the collision integral, however, are very high even with the most advanced discretization methods. As a result, the Boltzmann equation is rarely solved directly in multidimensional applications. Instead, alternative techniques such as direct simulation Monte Carlo (DSMC) methods, see e.g., [1], are used for simulating engineering applications. However statistical noise that is inherent to DSMC methods makes it cumbersome to couple these methods to deterministic models, for example, to the Navier–Stokes equations. Time-dependent problems represent an additional challenge since the stochastic noise can propagate and perturb continuum solutions. Also, DSMC methods may become prohibitively expensive for the simulation of the low speed flows where the flow velocity is less than the mean molecular thermal velocity. It requires a very large statistical sample in this case to keep the statistical noise

* Corresponding author at: Department of Mathematics, California State University Northridge, Northridge, CA, USA.
E-mail addresses: Alexander.Alekseenko@csun.edu (A. Alekseenko), Eswar.Josyula@us.af.mil (E. Josyula).

http://dx.doi.org/10.1016/j.jcp.2014.03.031
0021-9991/

from overpowering the signal. To overcome these difficulties several simplified deterministic approaches were developed. In particular, Lattice–Boltzmann methods [2], the discrete velocity methods [3], the method of model kinetic equations [4,5], the method of moments and the extended hydrodynamics approach [6] are used to obtain approximate deterministic solutions to the Boltzmann equation. However, analytical error evaluation of these models is not straightforward. It is therefore important to develop methods capable of solving the full Boltzmann equation directly. Such methods can be used both for validation and for obtaining solutions where approximate techniques fail to be sufficiently accurate.

Historically, however, one can count only a handful of successful attempts at the solution of the full Boltzmann equation. In [7] Tcheremissine presented an approach for fast evaluation of the collision integral based on a uniform discrete ordinate velocity discretization. In this approach the collision integral is written in the form of an eight-fold integral (six velocity and two spatial integrals) using Dirac delta-function formalism. Conservation of mass, momentum and energy is achieved by using a special interpolation of the velocity distribution function at the off-grid values of post-collision velocities. Efficient evaluation of the multidimensional integral is accomplished using quasi-stochastic Korobov integration. Tcheremissine's approach was successfully applied to simulations of the Boltzmann equation in more than one spatial dimension [8–10] (however, see also [11]). On the other hand, being a not fully deterministic method, its accuracy is not easy to evaluate. In particular, it is anticipated that accuracy of Korobov schemes deteriorates on non-smooth solutions, such as ones arising in problems with gas-to-surface interactions.

A number of numerical methods for solving the Boltzmann equation have been proposed based on the Fourier transform and Fourier–Galerkin discretizations in the velocity variable. To the best knowledge of the authors, the first application of Fourier–Galerkin discretizations to the deterministic evaluation of the Boltzmann collision operator appeared in Pareschi and Perthame [12]. A particularly simple form of the collision operator was obtained in their work using pre-computed kernels that depend only on the collision model. Their discretization requires $O(n^6)$ operations, where n is the number of Fourier frequencies and collocation points in the velocity space. This is smaller than $O(n^8)$ operations required for a straightforward application of the discrete ordinate approach. Stability and accuracy of Fourier–Galerkin approximations are discussed in detail in [13,14]. In [15] Bobylev and Rjasanow proposed a related approach based on a Carleman-type representation of the collision operator and the fast Fourier transform. In [16–18] the Carleman representation was adapted to be used together with the Fourier–Galerkin discretization resulting in a method of computational complexity $O(Mn^3 \log n)$, where M is associated with the number of the quadrature points on the impact parameter sphere. The method was applied to the simulation of one and two dimensional flows in [19,20]. Recently, the Carleman representation was generalized to a discrete ordinate approach in [21]. A closely related method results from a direct application of the Fourier transform in the velocity variable to the Boltzmann collision operator. In Kirsch and Rjasanow [22] an approach based on the Fourier transform is formulated for a deterministic evaluation of the Boltzmann collision operator with $O(n^6)$ operations. The approach was later extended by Gamba and Tharkabhushanam [23] and applied to the simulation of spatially inhomogeneous flows in [24,25].

Although extremely efficient, methods using the Fourier basis functions (or any other globally supported functions) are expected to require large numbers of degrees of freedom when approximating solutions that are discontinuous or have sharp gradients. Such situations arise frequently in modeling gas–surface interactions in rarefied gas and also in strong shock waves. It is expected that better approximation properties can be achieved in these problems using locally supported approximations of the velocity distribution function. An approach based on the uniform piece-wise constant approximation of the distribution function was proposed by Aristov, see e.g., [26]. By exploring the simple form of the discrete distribution function and re-ordering quadrature rules the author transforms the discrete collision integral to the form of a bilinear operator. Moreover, the author establishes analytical expressions for components of the kernel of the collision integral in the case of hard spheres potential, e.g., [11,26]. Other examples of solutions of the Boltzmann equations include [27,28] and most recently [21,29–33].

Discontinuous Galerkin approximations can provide high order approximations in regions where the distribution function is smooth and can also accommodate for the functions' discontinuities. In [34,35] it was shown that high order DG approximations in the velocity space can be very accurate in preserving mass, momentum and energy in the discrete solution even if the solution is rough. In addition, DG methods are well suited for adaptive techniques and parallel implementation. This motivated the authors to develop a nodal DG discretization in the velocity variable for the full Boltzmann equation. Our nodal DG basis in the velocity space is constructed from the Lagrange basis functions on Gauss quadrature nodes [36]. The resulting discretized form of the kinetic transport equation is equivalent to the discrete ordinate method. Similar to [12,37] we rewrite the Galerkin projection of the Boltzmann collision operator in the form of a bilinear operator with a pre-computed kernel. The storage for the components of the collision kernel grows only as $O(n^5)$ where n is the number of nodal points in one velocity dimension. This is rather small compared to pre-computed kernels in other deterministic techniques. The savings are due to a number of symmetries that hold for the kernels on regular grids and also due to the locality of support of the basis functions. The number of operations to compute the collision operator at one point in physical space is $O(n^8)$. This is significantly higher than the $O(n^6)$ operations for methods based on the Fourier basis functions. However, savings may be found for DG methods in the future by using adaptive grids in the velocity space and using sparse three dimensional DG bases, e.g., the hyperbolic cross. Another option to improve performance of DG velocity methods is to use stochastic integration similar to that used in [8]. Attractive features of the method are that it is easy to program and is suitable for scalable parallelization. The method is implemented on large parallel computers and was used to simulate spatially homogeneous relaxation with 61^3 velocity points. Preliminary results from this method appeared in [33]. In this work we establish some useful symmetry properties of the collision kernel, perform additional study of the algorithm

computational complexity and accuracy and present a comparison of the new deterministic method with the DSMC method [38] for the problem of spatially homogeneous relaxation.

DG velocity approximations were first applied to deterministic evaluation of the Boltzmann collision operator by Majorana [39]. For a DG basis using the five collisional invariants, the author showed exact conservation of mass momentum and energy under the assumption that the collision kernels are computed precisely. The present paper extends the approach to arbitrary high order nodal-DG bases and addresses problems that are specific to high fidelity implementation of the method. Specifically, we minimized the storage of the pre-computed kernels and limited the accumulation of roundoff errors in calculations. To the best knowledge of the authors we present the first implementation of the DG velocity discretization of the Boltzmann collision operator.

Finally we note that the paper is primarily concerned with the spatially homogeneous case of the Boltzmann equation. That is, we assume early on that the velocity distribution function is constant in physical space at all times. This simplification allows us to eliminate the transport part of the Boltzmann equation and restrict the numerical solution to evaluations of the collision operator (which is the main challenge in any deterministic approach to the Boltzmann equation). In Section 3, however, we apply a nodal DG discretization in the velocity variable to the full transport part. The result is a first order symmetric hyperbolic system of equations virtually equivalent to a discrete ordinate formulation. Spatial and temporal discretizations of discrete ordinate formulations have been developed by many authors using various techniques, see, e.g., [4,5,8,11,34,40]. Thus, rather than adopting a particular technique for the spatial discretization of the transport part, we point out that our DG velocity discretization and the method for evaluating the collision operator can be merged with existing discrete ordinate approaches relatively easily.

This paper is organized as follows. Section 2 presents the kinetic framework and the Boltzmann equation. Section 3 describes the nodal DG discretization in the velocity space. The symmetric form of the Galerkin projection of the Boltzmann collision operator is introduced in Section 4. The fully discrete form of the collision operator is discussed in Section 5. The results of numerical simulations for the problem of spatially homogeneous relaxation are given in Sections 6 and 7.

2. The Boltzmann equation

In the kinetic approach the gas is described using the molecular velocity distribution function $f(t,\vec{x},\vec{v})$ which is defined by the following property: $f(t,\vec{x},\vec{v})\,d\vec{x}\,d\vec{v}$ gives the number of molecules that are contained in the box with the volume $d\vec{x}$ around point $\vec{x}$ whose velocities are contained in a box of volume $d\vec{v}$ around point $\vec{v}$. Here by $d\vec{x}$ and $d\vec{v}$ we denote the volume elements $dx\,dy\,dz$ and $du\,dv\,dw$, correspondingly. Evolution of the molecular distribution function is governed by the Boltzmann equation, which, in the case of one component atomic gas, has the form (see, for example [41,42])

$$\frac{\partial}{\partial t} f(t,\vec{x},\vec{v}) + \vec{v}\cdot\nabla_x f(t,\vec{x},\vec{v}) = I[f](t,\vec{x},\vec{v}). \tag{1}$$

Here $I[f](t,\vec{x},\vec{v})$ is the molecular collision operator. In most instances, it is sufficient to only consider binary collisions between molecules. In this case the collision operator takes the form

$$I[f](t,\vec{x},\vec{v}) = \int_{\mathbb{R}^3}\int_0^{2\pi}\int_0^{b_*} \left(f\left(t,\vec{x},\vec{v}'\right)f\left(t,\vec{x},\vec{v}_1'\right) - f(t,\vec{x},\vec{v})\,f(t,\vec{x},\vec{v}_1)\right)|\vec{g}|b\,db\,d\varepsilon\,d\vec{v}_1, \tag{2}$$

where $\vec{v}$ and $\vec{v}_1$ are the pre-collision velocities of a pair of molecules, $\vec{g}=\vec{v}-\vec{v}_1$, b is the distance of closest approach (the separation of the unperturbed trajectories), b^* is the maximum distance of approach at which interaction between molecules is non-negligible (in fact, b^* is infinite for some interaction potentials) and ε is the angle between the collision plane and some reference plane. The vectors $\vec{v}'$ and $\vec{v}_1'$ are the post-collision velocities of the pair of particles. The post-collision velocities depend on the pre-collision velocities $\vec{v}$ and $\vec{v}_1$, the impact parameters b and ε and the molecular interaction potential used to model molecular collisions. The exact dependencies for $\vec{v}'$ and $\vec{v}_1'$ are not important for the formulation of the method. Therefore, for the sake of brevity, we omit them here and refer the reader to Sections 1.2 and 2.2 of [41] and Chapter I, Sections 6 and 7 of [42] where additional references can be found. In the simulations presented in this paper the hard spheres molecular potential was used, although the method is formulated with an arbitrary potential in mind.

3. Discontinuous Galerkin velocity discretization

Let us describe the DG velocity discretization that will be employed. We select a rectangular parallelepiped in the velocity space that is sufficiently large so that contributions of the molecular distribution function to first few moments outside of this parallelepiped are negligible. In most cases, some a-priori knowledge about the problem is available and such parallelepiped can be selected. We partition this region into rectangular parallelepipeds K_j. In this paper, only uniform partitions are considered; the advantages of using uniform partitions are explained in the next section. However, most of the approach carry over to non-uniform partitions and extensions to hierarchical and overlapping meshes are straightforward. On each element K_j, $j=1,\ldots,M$, we introduce a finite dimensional functional basis $\phi(\vec{v})_i^j$, $i=1,\ldots,s$. Notice that in general different approximation spaces can be used on different K_j. Thus the number of basis functions s may be different for different

velocity cells. However, the implementation of the method presented in this paper uses the same basis functions on all elements to save on computational storage.

Let $\vec{v} = (u, v, w)$ and let the numbers s_u, s_v and s_w determine the degrees of the polynomial basis functions in the velocity components u, v and w, respectively. Let $K_j = [u_L^j, u_R^j] \times [v_L^j, v_R^j] \times [w_L^j, w_R^j]$. The basis functions are constructed as follows. We introduce nodes of the Gauss quadratures of orders s_u, s_v, and s_w on each of the intervals $[u_L^j, u_R^j]$, $[v_L^j, v_R^j]$, and $[w_L^j, w_R^j]$, respectively. Let these nodes be denoted $\kappa_p^{j,u}$, $p = 1, s_u$, $\kappa_q^{j,v}$, $q = 1, s_v$, and $\kappa_r^{j,w}$, $r = 1, s_w$. We define one-dimensional Lagrange basis functions as follows, see e.g., [36],

$$\phi_l^{j;u}(u) = \prod_{\substack{p=1,s^u \\ p\neq l}} \frac{\kappa_p^{j,u} - u}{\kappa_p^{j,u} - \kappa_l^{j,u}}, \qquad \phi_m^{j;v}(v) = \prod_{\substack{q=1,s^v \\ q\neq m}} \frac{\kappa_q^{j,v} - v}{\kappa_q^{j,v} - \kappa_m^{j,v}}, \qquad \phi_n^{j;w}(w) = \prod_{\substack{r=1,s^w \\ r\neq n}} \frac{\kappa_r^{j,w} - w}{\kappa_r^{j,w} - \kappa_n^{j,w}}. \tag{3}$$

The three-dimensional basis functions are defined as $\phi_i^j(\vec{v}) = \phi_l^{j;u}(u)\phi_m^{j;v}(v)\phi_n^{j;w}(w)$, where $i = 1, \ldots, s = s_u s_v s_w$ is the index running through all combinations of l, n, and m. (In the implementation discussed in this paper, i is computed using the following formula $i = (l-1) * s_v * s_w + (m-1) * s_w + n$.)

Lemma 3.1. *The following identities hold for basis functions $\phi_i^j(\vec{v})$:*

$$\int_{K_j} \phi_p^j(\vec{v})\phi_q^j(\vec{v})\, d\vec{v} = \frac{\Delta\vec{v}^{\,j}}{8}\omega_p\delta_{pq}, \qquad \int_{K_j} \vec{v}\phi_p^j(\vec{v})\phi_q^j(\vec{v})\, d\vec{v} = \frac{\Delta\vec{v}^{\,j}}{8}\vec{v}_p^{\,j}\omega_p\delta_{pq}, \tag{4}$$

where $\Delta\vec{v}^{\,j} = (u_R^j - u_L^j)(v_R^j - v_L^j)(w_R^j - w_L^j)$, $\omega_p := \omega_l^{s_u}\omega_m^{s_v}\omega_n^{s_w}$, and $\omega_l^{s_u}$, $\omega_m^{s_v}$, and $\omega_n^{s_w}$ are the weights of the Gauss quadratures of orders s_u, s_v, and s_w, respectively and indices l, n, and m of one-dimensional basis functions correspond to the three-dimensional basis function $\phi_p^j(\vec{v}) = \phi_l^{j;u}(u)\phi_m^{j;v}(v)\phi_n^{j;w}(w)$, and the vector $\vec{v}_p^{\,j} = (\kappa_l^{j,u}, \kappa_m^{j,v}, \kappa_n^{j,w})$.

The lemma follows by re-writing the three-dimensional integral as an iterative integral and by reviewing the integrals of products of one-dimensional basis functions. The orthogonality of one-dimensional basis functions in each variable follows by replacing the one-dimensional integrals with Gauss quadratures on s^u nodes (similarly, s^v ans s^w nodes) and recalling that these quadratures are precise on polynomials of degrees at most $2s^u - 1$ (similarly, $2s^v - 1$ and $2s^w - 1$) and by recalling that the constructed basis functions vanish on all nodes but one at which they are equal to one.

We assume that on each K_j the solution to the Boltzmann equation is sought in the form

$$f(t, \vec{x}, \vec{v})|_{K_j} = \sum_{i=1,s} f_{i;j}(t, \vec{x})\phi_i^j(\vec{v}). \tag{5}$$

The discontinuous Galerkin (DG) velocity discretization that we shall study results by substituting the representation (5) into (1) and multiplying the result by a test basis function and integrating over K_j. Repeating this for all K_j and using the identities (4) we arrive at

$$\partial_t f_{i;j}(t, \vec{x}) + \vec{v}_i^{\,j} \cdot \nabla_x f_{i;j}(t, \vec{x}) = I_{\phi_i^j}, \tag{6}$$

where $I_{\phi_i^j}$ is the projection of the collision operator on the basis function $\phi_i^j(\vec{v})$:

$$I_{\phi_i^j} = \frac{8}{\omega_i \Delta\vec{v}^{\,j}} \int_{K_j} \phi_i^j(\vec{v}) I[f](t, \vec{x}, \vec{v})\, d\vec{v}. \tag{7}$$

Various equivalent formulations may be proposed for $I_{\phi_i^j}$ for the purpose of numerical implementation. In Section 4, we introduce the bilinear form with a symmetric kernel that will be used. However, integrals in the velocity variable and in the impact parameters that are included in $I_{\phi_i^j}$ in all formulations are not expected to be easily admissible to analytic evaluations, even after the substitution of (5). Instead, additional approximations are made to define the fully discretized DG model. These approximations are discussed in Section 5. However, results of Sections 4 and 5 are concerned only with the collision integral. In particular, they do not change the transport part of (6) which has the complexity of a discrete ordinate formulation. A formulation with similar properties has been presented in Gobbert and Cale [40]. Their Galerkin formulation, however, uses global basis functions of high order Hermite's polynomials. Formulation (5) therefore extends their approach.

4. Reformulation of the Galerkin projection of the collision operator

We will now introduce the formalism that will be used for evaluating the DG projection of the collision operator. For a moment, we will assume that the Galerkin representation (5) has not been substituted into (7) yet. Also, we notice that $\phi_i^j(\vec{v})$ can be extended by zero to the entire $\mathbb{R}^3$. With these assumptions, (7) becomes

$$I_{\phi_i^j} = \frac{8}{\omega_i \Delta \vec{v}^j} \int_{R^3} \phi_i^j(\vec{v}) \int_{R^3} \int_0^{2\pi} \int_0^{b_*} \left(f\left(t, \vec{x}, \vec{v}'\right) f\left(t, \vec{x}, \vec{v}_1'\right) - f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) \right) |\vec{g}| b \, db \, d\varepsilon \, d\vec{v}_1 \, d\vec{v}. \tag{8}$$

Using symmetry properties of the collision operator (see, e.g., [41], Section 2.4), we can replace the last expression with

$$I_{\phi_i^j} = \frac{8}{\omega_i \Delta \vec{v}^j} \int_{R^3} \int_{R^3} \frac{1}{2} \int_0^{2\pi} \int_0^{b_*} \left(\phi_i^j\left(\vec{v}'\right) + \phi_i^j\left(\vec{v}_1'\right) - \phi_i^j(\vec{v}) - \phi_i^j(\vec{v}_1) \right) f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) |\vec{g}| b \, db \, d\varepsilon \, d\vec{v}_1 \, d\vec{v}. \tag{9}$$

The first principles of the kinetic theory imply that changes in $f(t, \vec{x}, \vec{v})$ and $f(t, \vec{x}, \vec{v}_1)$ with respect to $\vec{x}$ are extremely small at distances of a few b^*, see e.g., [6], Section 3.1.4. We will neglect these changes and therefore will assume that values of $\vec{x}$ in $f(t, \vec{x}, \vec{v})$ and $f(t, \vec{x}, \vec{v}_1)$ are independent of the impact parameters b and ε. With this assumption, $f(t, \vec{x}, \vec{v})$ and $f(t, \vec{x}, \vec{v}_1)$ can be removed from under the integrals in b and ε in (9) to obtain

$$\begin{aligned} I_{\phi_i^j} &= \frac{8}{\omega_i \Delta \vec{v}^j} \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) \frac{|\vec{g}|}{2} \int_0^{2\pi} \int_0^{b_*} \left(\phi_i^j\left(\vec{v}'\right) + \phi_i^j\left(\vec{v}_1'\right) - \phi_i^j(\vec{v}) - \phi_i^j(\vec{v}_1) \right) b \, db \, d\varepsilon \, d\vec{v}_1 \, d\vec{v} \\ &= \frac{8}{\omega_i \Delta \vec{v}^j} \int_{R^3} \int_{R^3} f(t, \vec{x}, \vec{v}) f(t, \vec{x}, \vec{v}_1) A\left(\vec{v}, \vec{v}_1; \phi_i^j\right) d\vec{v}_1 \, d\vec{v}, \end{aligned} \tag{10}$$

where

$$A\left(\vec{v}, \vec{v}_1; \phi_i^j\right) = \frac{|\vec{g}|}{2} \int_0^{2\pi} \int_0^{b_*} \left(\phi_i^j\left(\vec{v}'\right) + \phi_i^j\left(\vec{v}_1'\right) - \phi_i^j(\vec{v}) - \phi_i^j(\vec{v}_1) \right) b \, db \, d\varepsilon. \tag{11}$$

We notice that because $A(\vec{v}, \vec{v}_1; \phi_i^j)$ is independent of time, it can be pre-computed and stored to be used in many individual simulations as long as the velocity discretization is the same. Integrals in (11) can be computed with good accuracy for an arbitrary potential.

The form (10), (11) of the discrete collision operator was first used by Pareschi and Perthame in [12] to achieve efficiency in a spectral Fourier discretization of the Boltzmann equation. In [14,18] explicit formulas were developed for the components of the Fourier discretization of the collision kernel for hard spheres and Maxwell molecules. This form of the collision operator was also used in connection with the method of moments. In [37] the form (10), (11) was used to develop differential estimates on even moments of the solution. A similar formalism is presented in detail in [6] in connection with the development of macroscopic approximations to the Boltzmann equation. In particular, in [6] expressions for collision kernels corresponding to globally polynomial moments are obtained in closed form for Maxwell molecules. Expressions for hard spheres are presented in [43]. In [44] and [43] the latter formulas are used in the context of Lattice–Boltzmann method to construct a closure that is based on the full Boltzmann collision operator. Also, in [45] a general algorithm is proposed to systematically develop values for moments of the collision operator for any collision potential. Most recently, the symmetric form of the collision operator was used in simulations of full Boltzmann equation in [21,32]. However, a very similar form of the collision operator appears in [7,26]. In particular, in [7] a formalism of Dirac delta-functions is employed in the context of a discrete ordinate approximation of the collision integral. We argue that the Galerkin velocity approach presented here can be generalized to obtain the approach of [7] by selecting appropriate trial and test spaces and taking appropriate limits.

While applications of Fourier–Galerkin methods and spectral methods using globally supported polynomials to the approximation of the collision operator were known for some time, an application of discontinuous Galerkin approximations appeared only recently [39]. One of the advantages of the DG methods is that the conservation properties can be achieved by a clever construction of the Galerkin basis (e.g., see the discussion in [39]). Another advantage is that DG methods are adaptive. Adaptivity is important in applications involving gas–surface interactions where the velocity distribution function is discontinuous in the velocity space near the walls. One of the biggest challenges that comes with using locally supported Galerkin basis is the large memory requirement for storing the pre-computed kernel $A(\vec{v}, \vec{v}_1; \phi_i^j)$. A straightforward counting suggests that the storage of $O(n^9)$ is required, where n is the number of velocity points in one velocity dimension. The following statements will be used to reduce the necessary storage to $O(n^5)$ in the case of a regular grid in the velocity space.

Lemma 4.1. *Let operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ be defined by (11) with all gas particles having the same mass and the potential of the particles interaction being spherically symmetric. Then $A(\vec{v}, \vec{v}_1; \phi_i^j)$ is symmetric with respect to $\vec{v}$ and $\vec{v}_1$, that is*

$$A\left(\vec{v}, \vec{v}_1; \phi_i^j\right) = A\left(\vec{v}_1, \vec{v}; \phi_i^j\right), \quad \forall \vec{v}, \vec{v}_1 \in \mathbb{R}^3. \tag{12}$$

Also,

$$A\left(\vec{v}, \vec{v}; \phi_i^j\right) = 0, \quad \forall \vec{v} \in \mathbb{R}^3. \tag{13}$$

A sketch of the proof of the lemma is in Appendix A.

Next lemma states that $A(\vec{v}, \vec{v}_1; \phi_i^j)$ is invariant with respect to a shift in the velocity space.

Lemma 4.2. *Let operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ be defined by (11) and let the potential of molecular interaction be dependent only on the distance between the particles. Then $\forall \xi \in \mathbb{R}^3$*

$$A\left(\vec{v} + \vec{\xi}, \vec{v}_1 + \vec{\xi}; \phi_i^j(\vec{v} - \vec{\xi})\right) = A\left(\vec{v}, \vec{v}_1; \phi_i^j\right). \tag{14}$$

A sketch of the proof of the lemma is in Appendix A.

We notice that Lemma 4.2 allows to significantly reduce the required memory storage for operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ in the case of a uniform rectangular elements and the same basis functions on each element. In this case, information about $A(\vec{v}, \vec{v}_1; \phi_i^j)$ needs to be stored for a single cell only. Values of $A(\vec{v}, \vec{v}_1; \phi_i^j)$ for the rest of the cells may be restored using its invariance with respect to a constant shift. Of course, strictly speaking, this can only be done on an infinite partition of the entire velocity space. However, one can still successfully apply the invariance property on finite partitions provided that the support of the solution is well contained inside the velocity domain.

The next lemma is a generalization of Lemma 4.2 in the sense that it allows for more general transformations of the velocity space. To formulate the lemma, we need to recall the following definition. Consider the Euclidean space of vectors $\mathbb{R}^3$ with the norm defined the usual way $|\vec{v}| = \sqrt{\vec{v} \cdot \vec{v}} = \sqrt{u^2 + v^2 + w^2}$. A linear operator $S : \mathbb{R}^3 \to \mathbb{R}^3$ is called a linear isometry if for any vector $\vec{v} \in \mathbb{R}^3$,

$$|S\vec{v}| = |\vec{v}|.$$

Thus a linear isometry maps a vector into a vector of equal length. Note that this means that an isometry conserves distance between any two points. It follows, in particular, that an isometry over the entire $\mathbb{R}^3$ will transform lines into lines and spheres into spheres of the same radius. It will also preserve angles between lines. The most useful examples of isometries for us will be translations, rotations and reflections of $\mathbb{R}^3$. We will show next that operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ is invariant under the action of an isometry, as is expressed by the next theorem.

Lemma 4.3. *Let operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ be defined by (11) and the potential of molecular interaction be dependent only on the distance between the particles. Let $S : \mathbb{R}^3 \to \mathbb{R}^3$ be a linear isometry of $\mathbb{R}^3$. Then*

$$A\left(S\vec{v}, S\vec{v}_1; \phi_i^j\left(S^{-1}\vec{u}\right)\right) = A\left(\vec{v}, \vec{v}_1; \phi_i^j\right) \tag{15}$$

A sketch of the proof of the lemma is given in Appendix A.

5. Discrete velocity form of the collision integral

The numerical approximation of (10) follows by replacing the velocity distribution function with the DG approximation (5) and the integrals in (10) with Gauss quadratures using the nodes $\vec{v}_p^j$. The resulting approximation of the collision operator takes the form

$$I_{\phi_i^j} = \frac{8}{\omega_i \Delta \vec{v}^j} \sum_{j^*=1}^{M} \sum_{i^*=1}^{s} \sum_{j'=1}^{M} \sum_{i'=1}^{s} f_{i^*;j^*}(t, \vec{x}) f_{i';j'}(t, \vec{x}) A_{i^* i' i}^{j^* j' j}. \tag{16}$$

Here the quantities

$$A_{i^* i' i}^{j^* j' j} = \frac{\Delta \vec{v}^{j^*} \omega_{i^*}}{8} \frac{\Delta \vec{v}^{j'} \omega_{i'}}{8} A\left(\vec{v}_{i^*}^{j^*}, \vec{v}_{i'}^{j'}; \phi_i^j\right) \tag{17}$$

are independent of time and are computed only once for each DG velocity discretization using adaptive quadratures based on Simpson's rule. It should be noted, however, that $A(\vec{v}, \vec{v}_1; \phi_i^j)$ is not a locally supported function. In particular, if one of

Table 1
The growth of the number of non-zero components of $A_{i^*i'i}^{j^*j'j}$ for a single basis function ϕ_i^j with respect to the numbers of velocity nodes and the estimated orders of growth.

Nodes per dimension, N	9	15	21	27	33
Components, $s_u = 1$	11 278	143 804	781 002	2 693 240	7 261 854
Order, $s_u = 1$	4.98	5.03	4.93	4.94	
Components, $s_u = 3$	39 022	459 455	2 355 130	8 142 006	21 915 065
Order, $s_u = 3$	4.83	4.86	4.94	4.93	

the vectors $\vec{v}$ or $\vec{v}_1$ falls in the support of ϕ_i^j then $A(\vec{v}, \vec{v}_1; \phi_i^j) = O(|\vec{g}|)$, as $|\vec{g}| \to \infty$. If neither $\vec{v}$ nor $\vec{v}_1$ is in the support of ϕ_i^j, then $A(\vec{v}, \vec{v}_1; \phi_i^j)$ is decreasing as $|\vec{g}| \to \infty$. However, it does not equal zero, no matter how large is $|\vec{g}|$. Because of this, additional considerations must be employed in order to limit the number of entries $A_{i^*i'i}^{j^*j'j}$ that will be stored. In our approach, two strategies are employed. The first strategy eliminates all values of $A_{i^*i'i}^{j^*j'j}$ whose magnitudes fall below the levels of expected errors in the adaptive quadratures. The second strategy eliminates values of $A_{i^*i'i}^{j^*j'j}$ that correspond to pairs $\vec{v}_{i^*}^{j^*}$ and $\vec{v}_{i'}^{j'}$ for which $|\vec{v}_{i^*}^{j^*} - \vec{v}_{i'}^{j'}|$ is greater than some specified diameter. While the rationale for the first strategy is self-explanatory, the second strategy can be justified by the fact that solutions to the Boltzmann equations decrease rapidly at infinity. They can generally be assumed to be zero outside of a ball of the diameter of several thermal velocities with the center at the stream velocity (see, e.g., [7] and also [13]). Because in most cases, thermal velocity can be estimated without knowing much about the final solution, one can limit $A_{i^*i'i}^{j^*j'j}$ to only those pairs of $\vec{v}_{i^*}^{j^*}$ and $\vec{v}_{i'}^{j'}$ that are contained in such a ball. Note, however, that an excessive use of these two strategies will adversely impact the accuracy of the method, especially, since Lemmas 4.1–4.3 are used in the implementation. We observed that discarding the components of $A_{i^*i'i}^{j^*j'j}$ that are smaller by the absolute value than 10^{-4} resulted in a considerable violation of conservation laws and in wrong relaxation times. Also, using a diameter smaller than four thermal velocities in the second strategy resulted in incorrect relaxation times (although the density, momentum and energy were conserved in the numerical solutions).

Table 1 illustrates the growth rate of the number of non-zero entries in $A_{i^*i'i}^{j^*j'j}$ for a single basis function ϕ_i^j. The dimensionless velocity domain is the cube with sides $[-3, 3]$ in each dimension. Two cases of DG basis are considered. The first one corresponds to DG approximations by constants given by $s_u = s_v = s_w = 1$. The second one corresponds to DG approximations by quadratic polynomials given by $s_u = s_v = s_w = 3$. Different number of velocity cells were used. Velocity pairs separated farther than 3 units were neglected. Threshold for adaptive quadrature error was set at 10^{-8}.

The top row of Table 1 corresponds to the numbers of degrees of freedom, N, in each velocity dimension. For example, if $s_u = s_v = s_w = 3$ and the velocity grid has $M = 5$ velocity cells in each dimension we obtain the number of degrees of freedom in each dimension to be $N = s_u M = 15$. The number of non-zero components of $A_{i^*i'i}^{j^*j'j}$ are listed in rows two and four. Estimated orders of growth are shown in rows three and five. It is observed that the number of non-zero components grows approximately as $O(N^5)$ in both piece-wise constant and piece-wise quadratic cases. This is considerably smaller than the expected growth rate of $O(N^6)$ obtained by counting of all possible pairs of pre-collision velocities. The reduction in size of $A_{i^*i'i}^{j^*j'j}$ can be attributed to the locality of the DG basis functions. Indeed when basis functions are locally supported, most of pairs $\vec{v}_{i^*}^{j^*}$ and $\vec{v}_{i'}^{j'}$ produce a collision sphere that does not overlap with the support of the basis function ϕ_i^j. As a result, the corresponding values of $A_{i^*i'i}^{j^*j'j}$ for all such combinations are zero.

It is however unlikely that $O(N^5)$ can be further reduced by, say, a more elaborate choice of the DG basis. Each evaluation of the Boltzmann collision equation in its original form requires the knowledge of $f(t, \vec{x}, \vec{v})$ at about N^5 pairs of velocity points. Indeed, to evaluate the collision operator one has to consider N^3 different selections of the second pre-collision velocity and to pair each selection with about N^2 combinations of impact parameters to produce on the order of N^5 various combinations of $\vec{v}$, $\vec{v}_1$, $\vec{v}'$ and $\vec{v}_1'$ that are averaged in (2). This suggests that (10) maintains the same information as the direct discretization of the full collision operator. Because of this, it is unlikely that the complexity $O(N^5)$ can be further reduced without restricting the collision operator itself.

It can be seen that the number of non-zero components is larger in the case of $s_u = 3$ than in the case of $s_u = 1$ for the same number of degrees of freedom, N. This is explained by the fact that supports of the basis functions are larger in the case of $s = 3$. Therefore, more pairs of velocities produce non-negligible integrals.

By multiplying the amount of storage required for one basis function, $O(N^5)$, by the total number of basis functions, N^3, one estimates the total amount of storage for $A_{i^*i'i}^{j^*j'j}$ to grow as $O(N^8)$. However, this number can be reduces back to $O(N^5)$ by using uniform partitions and Lemmas 4.2 and 4.3. Indeed, by identifying ways to map combinations of triples $\vec{v}_{i^*}^{j^*}$, $\vec{v}_{i'}^{j'}$ and ϕ_i^j into each other via linear isometries, a small subgroup of unique records in $A_{i^*i'i}^{j^*j'j}$ can be determined. Only this subgroup needs to be computed and stored. Records for the rest of the triples can be restored using (14) and (15).

In the simulations presented in this paper, the velocity domain was partitioned into uniform rectangular parallelepipeds and the same Lagrange basis functions were used on each element. One can notice that in this case, all cells can be obtained from a single cell by a constant shift. One also notices that basis functions and nodes can be obtained from the basis functions and the nodes of that selected cell using the substitution described in Lemma 4.2. It follows then that records can

Table 2
Time in seconds to advance one temporal step.

Number of nodes, N	9	15	21
Processor time in seconds, $s_u = 1$	0.14	8.63	134.07
Order, $s_u = 1$	8.05	8.15	
Processor time in seconds, $s_u = 3$	0.65	35.17	486.82
Order, $s_u = 3$	7.81	7.81	

be restored from the records of the canonical cell using (14) and that only records of the canonical cell need to be stored. Of course, in the case of a finite partition some shifts will produce values of the velocity that are not on the grid. However, if the support of the solution is well contained inside the domain, such values can be ignored in the summation of (16). Finally, if the nodes and basis functions have rotational symmetries within the element, more isometries can be considered and the storage for $A_{i^*i'i}^{j^*j'j}$ can be further reduced. However, it will still be proportional to N^5 even if more symmetries are found.

Because the components of $A_{i^*i'i}^{j^*j'j}$ are independent of each other, their evaluation can be parallelized using hundreds of processors. In the simulations presented in this paper up to 320 processors were used with MPI parallelization algorithms. The largest number of velocity nodes for which $A_{i^*i'i}^{j^*j'j}$ was computed was $N = 83$.

Times to compute (16) for a single time step on a single 2.3 MHz processor are presented in Table 2. One can notice that the computation time grows as $O(N^8)$. This is not at all surprising since the $O(N^5)$ operations for evaluation of collision operator need to be repeated at $O(N^3)$ velocity nodal points. The computational time grows at slightly higher rate for $s = 1$ than for $s = 3$. This can be explained by the fact that in the case of $s = 1$, $A_{i^*i'i}^{j^*j'j}$ is only stored at one node in the center of the grid $s = 1$ as compared to 27 nodes in the case of $s = 3$. The rest of the values are restored by Lemma 4.2 in both cases. However, in the case of $s = 1$ the application of Lemma 4.2 involves significantly more memory copying. Because memory copying is expensive, this causes computational time for $s = 1$ grow at a slightly faster rate. It is however believed that summation routines can be re-formulated so as to minimize the memory copying (see, e.g., [46]). However, even with the efficient summation the growth rate of $O(N^8)$ is believed to be intrinsic to the method and therefore is expected to quickly saturate the computational resources. However, it will be seen in the next section that the method yields reasonable calculation times on a single processor for values of N up to 33. Because of the fast growth of computational time it is expected that somewhere from 10^3 to 10^4 processors will be required to reach the value of $N = 100$. Because the method involves very little data exchange, we expect that parallelization of the algorithm will be very efficient. However, the main savings are expected from the construction of efficient Galerkin basis so as to minimize the total number of degrees of freedom while maintaining accuracy. Construction of such approximations will be the topic of the authors' future work.

6. Spatially homogeneous relaxation

To illustrate the work of the algorithm we present results of simulations of relaxation of monoatomic gas from perturbed states. Two problems are considered: relaxation of two equilibrium streams and relaxation of a discontinuous initial stage. The molecular collision was modeled using hard spheres potential in both problems. Two instances of DG discretizations were compared: approximations by piece-wise constants, corresponding to $s_u = s_v = s_w = 1$, and by piece-wise quadratic approximations, corresponding to $s_u = s_v = s_w = 3$. The time discretization in all simulations is by fifth order Adams–Bashforth method. The data for the Adams–Bashforth method is obtained using the fifth order Runge–Kutta method (see, e.g., [47]).

In the first problem the initial data is the sum of two Maxwellian distributions with mass densities, bulk velocities and temperatures of $\rho_1 = 6.634\text{E}{-6}$ kg/m^3, $\vec{v}_1 = (967.78, 0, 0)$ m/s, and $T_1 = 300$ K and $\rho_2 = 1.99\text{E}{-5}$ kg/m^3, $\vec{v}_2 = (322.59, 0, 0)$ m/s, and $T_2 = 1100$ K correspondingly. The spatially homogeneous relaxation is simulated for about 120 μs. The mean time between collisions for the steady state solution is estimated to be about 5.4 μs. The solution appeared to reach the steady state at about 45 μs.

In Figs. 1 and 2 solutions to the relaxation of the two Maxwellian streams are shown. Simulations corresponding to $s_u = s_v = s_w = 3$, $M = 7$ are shown in Fig. 1 and the simulation for $s_u = s_v = s_w = 1$, $M = 33$ in Fig. 2. One-dimensional sections of the solution by planes $v = 0$ and $w = 0$ are shown in Fig. 3. In Fig. 3(a) the DG approximations of the initial data are given and in Fig. 3(b) the approximations of the steady state are given. Both the piece-wise constant and the piece-wise quadratic case appear to capture the relaxation process successfully.

DG solutions were compared with the solutions obtained by established DSMC solvers [38]. In Fig. 4(a) the ratios of directional temperatures T_x and T_y to the average temperature $T_{\text{avg}} = T/3$ are shown. The solution reaches equilibrium state at about 45 μs. The DG approximation shows an excellent agreement with the DSMC solution. In Fig. 4(b) relative errors in the solution temperature are presented for $s_u = s_v = s_w = 3$, $M = 5, 7$ and $s_u = s_v = s_w = 1$, $M = 15, 27$. In our DG approach no special enforcement of conservation laws is used. Rather, it is expected that similar to [34,35] a satisfactory conservation can be achieved by using sufficiently refined DG approximations. It can be observed that for $s_u = s_v = s_w = 33$, $M = 7$ the temperature is computed correctly within three digits of accuracy. It was observed also that piece-wise constant approximations perform significantly better in this problem. This finding is consistent with [34] where it was found that

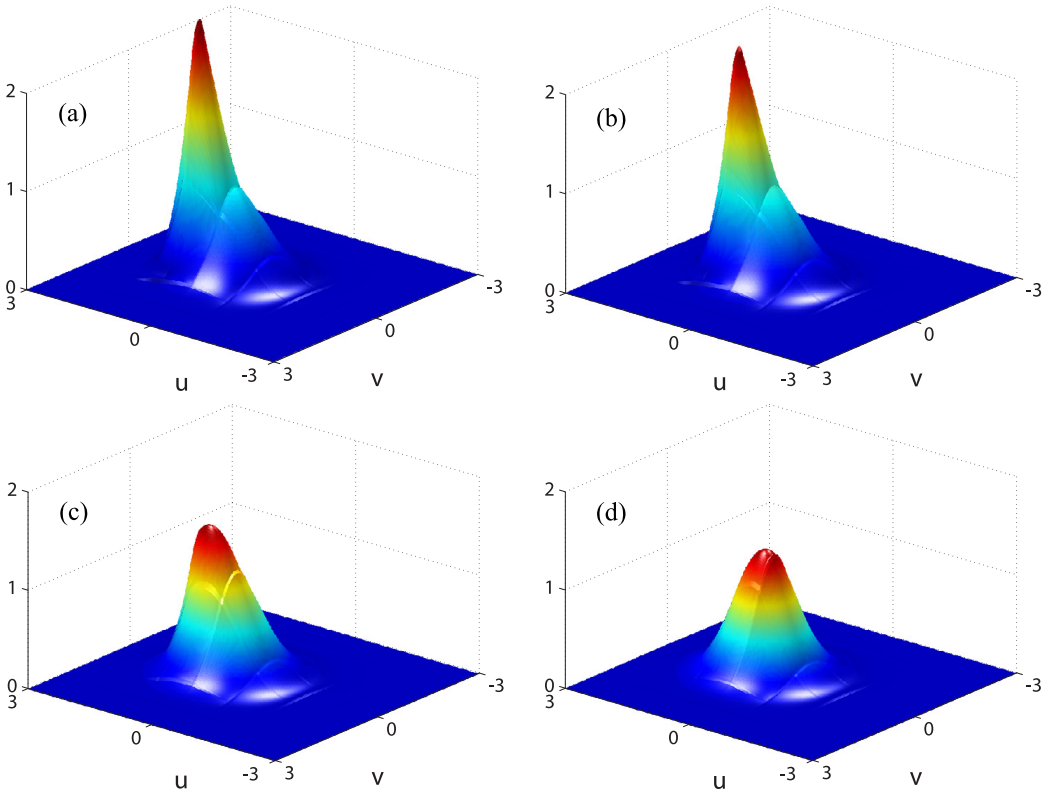

Fig. 1. Relaxation of two Maxwellian streams using piece-wise quadratic DG approximations. Sections of the solution by plane $w = 0$ are shown. The solutions correspond to $s_u = s_v = s_w = 3$ and 7 cells in each velocity dimension. (a) $t = 0$ μs, (b) $t = 1.3$ μs, (c) $t = 7.6$ μs, and (d) $t = 39.2$ μs.

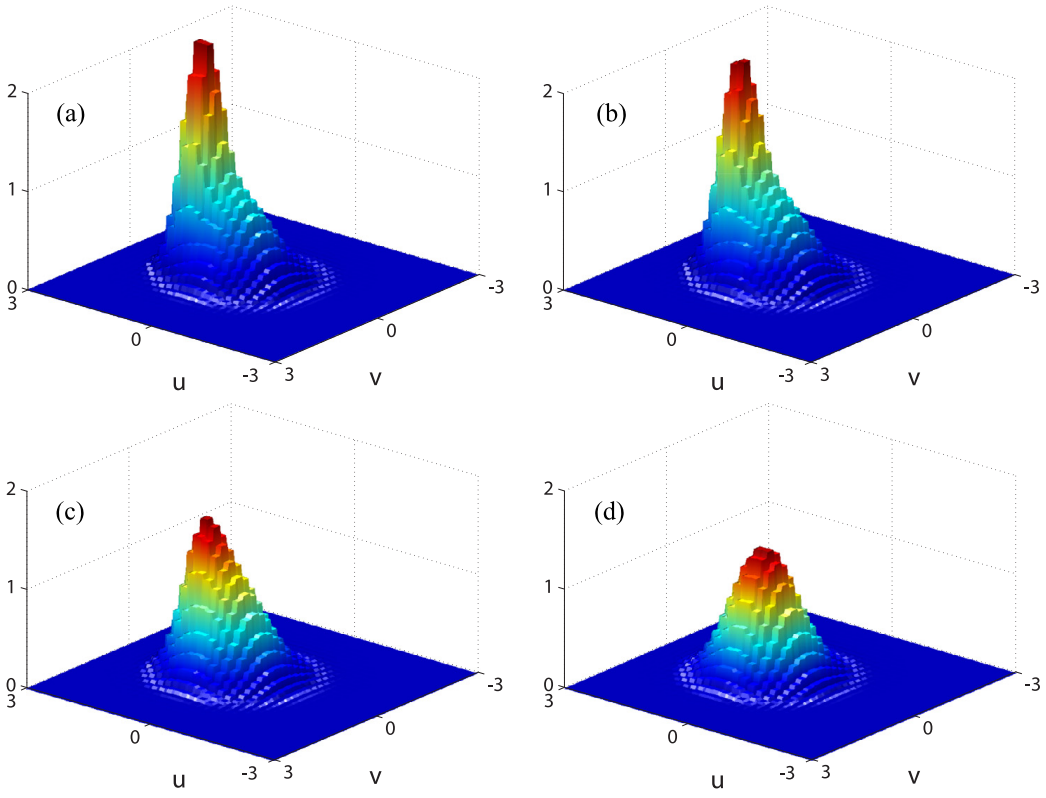

Fig. 2. Relaxation of two Maxwellian streams using piece-wise constant DG approximations. Sections of the solution by plane $w = 0$ are shown. The solutions correspond to $s_u = s_v = s_w = 1$ and 33 cells in each velocity dimension: (a) $t = 0$ μs; (b) $t = 1.3$ μs; (c) $t = 7.6$ μs; and (d) $t = 39.2$ μs.

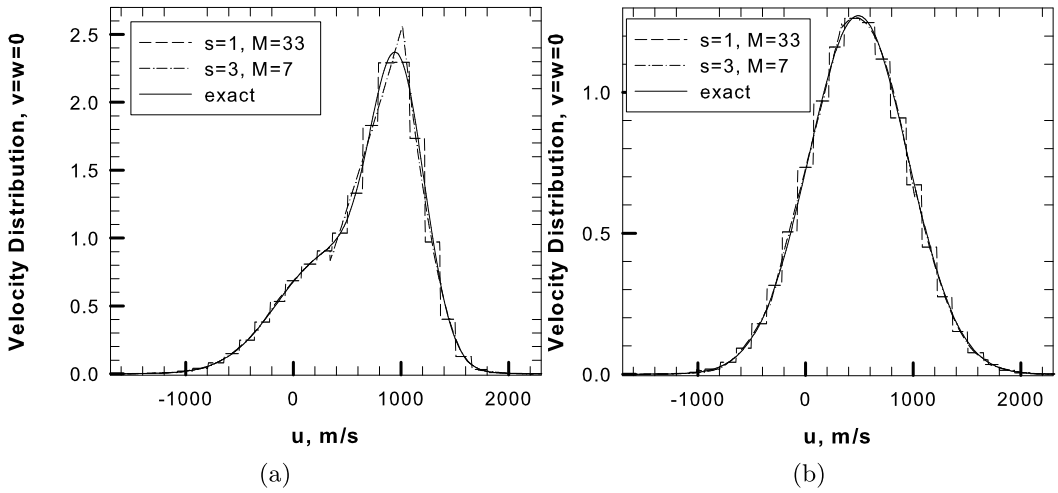

Fig. 3. Cross sections of the solution to the problem of relaxation of two Maxwellian streams by planes $v = 0$ and $w = 0$. DG approximations with $s_u = s_v = s_w = 1$, $M = 27$ and $s_u = s_v = s_w = 3$, $M = 7$ are compared. (a) Approximations of the initial data. (b) Approximations of the steady state.

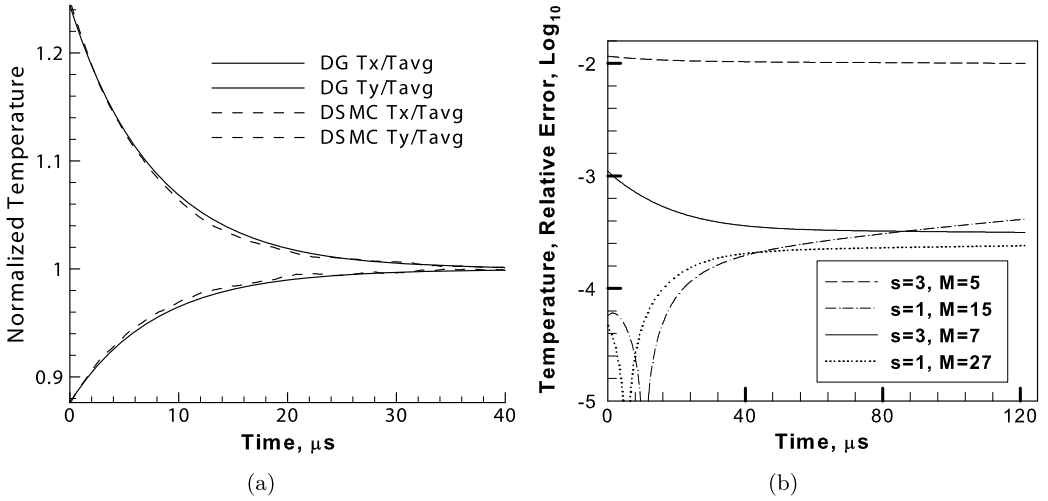

Fig. 4. Relaxation of two Maxwellian streams. (a) Comparison of the DG solution to the DSMC solution by I. Boyd [38]. Ratios T_x/T_{avg} and T_y/T_{avg} are plotted. Here T_x and T_y are directional temperatures and $T_{\text{avg}} = T/3$, where T is the temperature of the exact solution. (b) Relative errors in the solutions temperature. Piece-wise quadratic and piece-wise constant DG approximations are compared. The piece-wise constant approximations are more accurate in this problem.

piece-wise constant DG approximations are converging faster than high order DG approximations on smooth solutions. It is however expected that high order methods will be superior on the solutions that are not numerically smooth, e.g., due to truncation errors in spatially inhomogeneous problems.

Our second problem is concerned with the relaxation of two artificial streams with discontinuous initial data. Specifically, the initial distribution is a sum of two functions of the form

$$f_X(\vec{v}) = \begin{cases} \rho h, & \text{if } |\vec{v} - \vec{\bar{v}}| \leqslant r, \\ 0, & \text{if } |\vec{v} - \vec{\bar{v}}| > r, \end{cases} \qquad r = (T/5R)^{1/2}, \text{ and } h = (15\sqrt{5}/4\pi)(R/T)^{3/2}.$$

Here ρ, $\vec{\bar{v}}$ and T are given parameters. It is a straightforward exercise to verify that ρ, $\vec{\bar{v}}$ and T coincide with the mass density, bulk velocity and temperature of the distribution $f_X(\vec{v})$, correspondingly. The gas is argon. The values of the macroparameters for the two artificial streams are $\rho_1 = 0.332$ kg/m^3, $\vec{\bar{v}}_1 = (106.0, 0, 0)$ m/s, $T_1 = 300$ K and $\rho_2 = 0.332$ kg/m^3, $\vec{\bar{v}}_2 = (-106.0, 0, 0)$ m/s, $T_2 = 300$ K. Graphs of two-dimensional cross-sections of the initial distribution by plane $w = 0$ are given in Figs. 5(a) and 6(a). The mean time between molecular collisions is estimated to be about 0.39 ns. The simulations are performed for 190 ns. In Fig. 5 the simulations are presented using piece-wise quadratic approximations, $s_u = s_v = s_w = 3$, on $M = 7$ cells in each velocity dimension and in Fig. 6 using piece-wise constant approximations, $s_u = s_v = s_w = 1$, on $M = 27$ cells in each velocity dimension. One can see that the simulations are consistent in capturing the process of relaxation. The approximation artifacts visible in Fig. 5(a) are caused by the discontinuous initial data. These artifacts subside as the distribution functions relaxes and gets smoother.

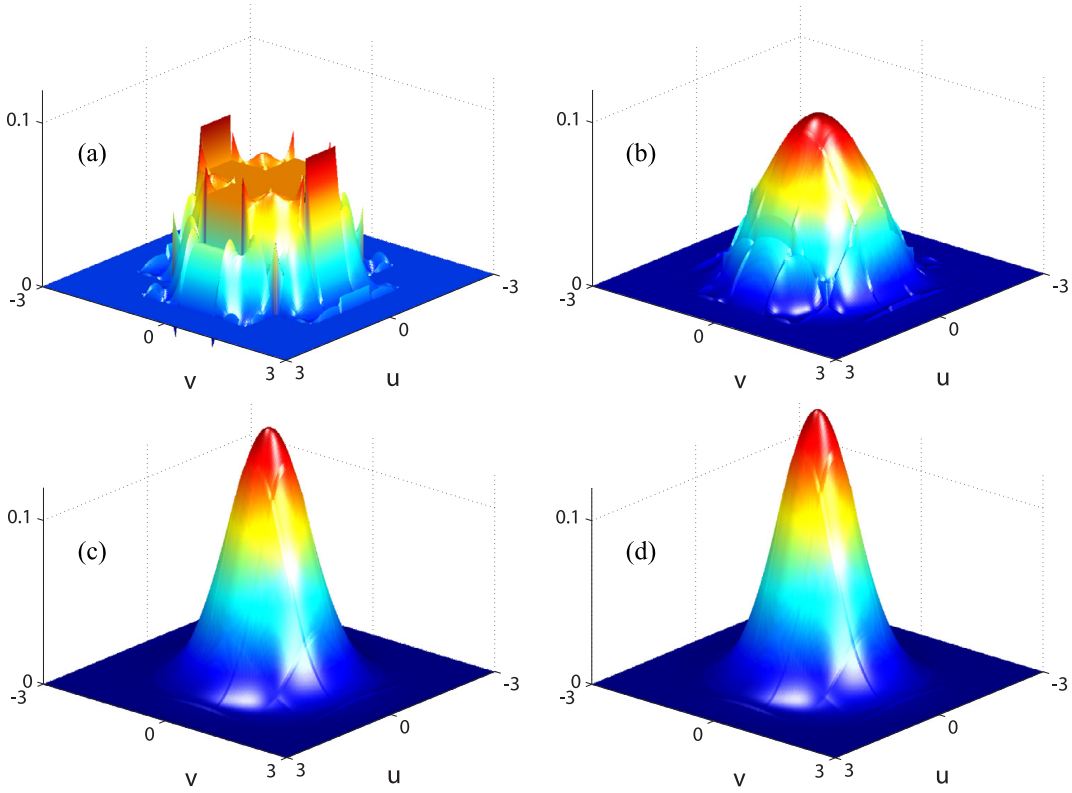

Fig. 5. Relaxation of two artificial streams using piece-wise quadratic DG approximations. Sections of the solution by plane $w = 0$ are shown. The solutions correspond to $s_u = s_v = s_w = 3$ and 7 cells in each velocity dimension. (a) $t = 0$ ns, (b) $t = 0.5$ ns, (c) $t = 2.1$ ns, and (d) $t = 4.5$ ns.

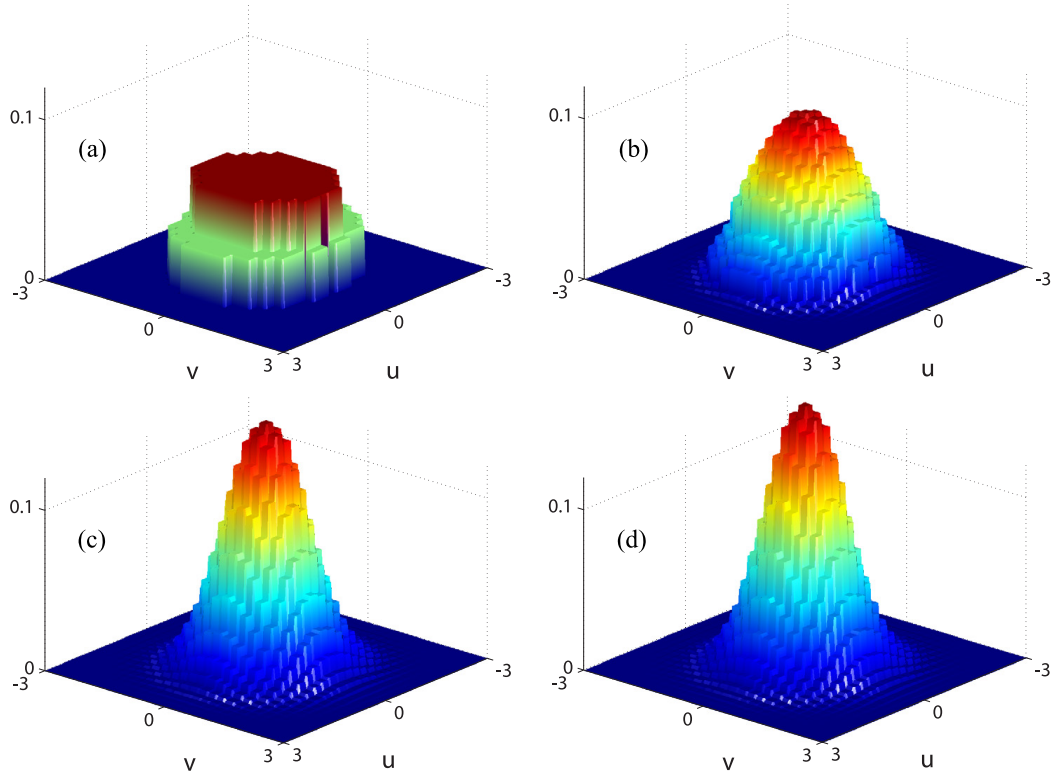

Fig. 6. Relaxation of two artificial streams using piece-wise constant DG approximations. Sections of the solution by plane $w = 0$ are shown. The solutions correspond to $s_u = s_v = s_w = 1$ and 27 cells in each velocity dimension. (a) $t = 0$ ns; (b) $t = 0.5$ ns; (c) $t = 2.1$ ns; and (d) $t = 4.5$ ns.

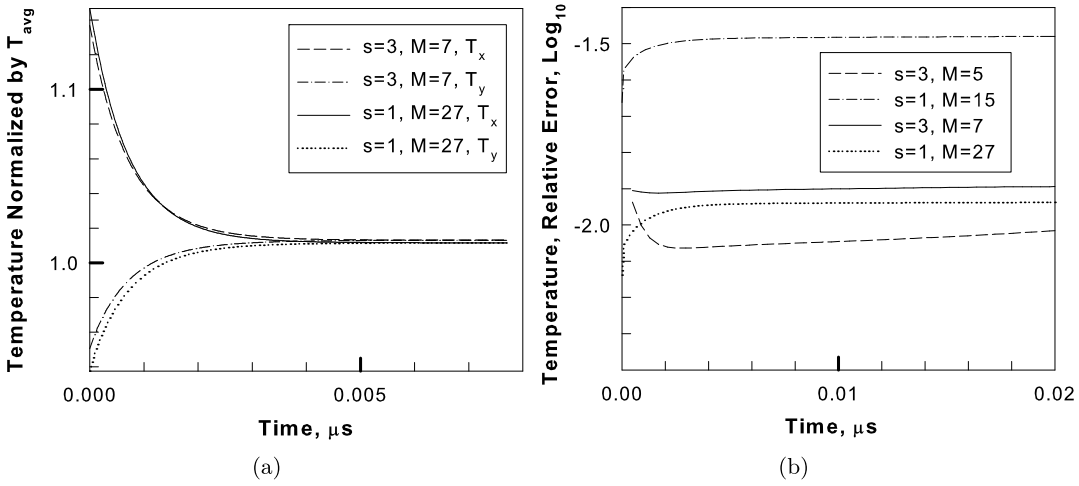

Fig. 7. Relaxation of two artificial streams. (a) Ratios T_x/T_{avg} and T_y/T_{avg} are shown. Here T_x and T_y are directional temperatures and $T_{\text{avg}} = T/3$. (b) Relative error in the solutions temperature. Piece-wise constant and piece-wise quadratic DG approximations are compared. Fast growth of error occurs in the piece-wise constant approximation at the initial stages of relaxation.

In Fig. 7(b) the conservation of temperature in the DG solutions is given in the cases of $s_u = s_v = s_w = 1$ and $s_u = s_v = s_w = 3$. One can notice that the approximations of discontinuous initial data are less accurate and also that the piece-wise constant DG approximations perform inferior to the piece-wise quadratic approximations. This observation is consistent with studies performed in [34]. It is believed that the fast convergence of the piece-wise constant approximations in the problem of relaxation of two Maxwellian distributions was due to the solution smoothness and to the fact that kinetic solutions exponentially decrease at infinity. As can be seen from the second example, accuracy of integration in piece-wise constant approximations deteriorates dramatically if the approximated solution is not smooth. One can see that growth of numerical errors in piece-wise constant DG approximations is fastest at the initial stages of the relaxation, when the solution is still far from being smooth. As the solution is getting smoother, however, the accuracy of quadrature rules increases and the conservation laws are satisfied with a better accuracy. As a result, we observe almost no change in the errors once the solutions reach the steady state.

7. Convergence of $s = 1$ and $s = 3$ DG approximations

Analysis of errors in numerical solutions to kinetic equations is complicated because it requires to consider several contributing factors including the accuracy of approximating the distribution function, the accuracy of approximating the nonlinear collision integral and the stability and physical validity of the resulting discrete system. Some theoretical results concerning the accuracy of deterministic methods for solving the Boltzmann equation can be found in [13–15,22,26,37,48,49]. However, in practice, accuracy of numerical solutions is verified by comparing them to a few known analytical solutions or to solutions obtained by other numerical techniques. Such comparison studies may also get complicated if high accuracy benchmark solutions are not available or if it is impossible to reach the asymptotic regime of convergence due to limitations in computing power. Detailed studies, as a rule, are carried out in one- and two-spatial dimensions with the expectation that errors behave similarly in the full three-dimensional cases (see, e.g., [17,35]).

In this section we consider some additional detail on the convergence of the DG approximations in the problem of spatially homogeneous relaxation of two Maxwellian streams. The initial data for simulations presented in this section is the same as for simulations shown in Figs. 1 and 2 in the previous section. We compare two sets of discrete solutions. The first set corresponds to piece-wise constant, $s_u = s_v = s_w = 1$, and the second set to piece-wise quadratic, $s_u = s_v = s_w = 3$, DG velocity approximations on uniform partitions. We note that our piece-wise constant DG approximations are equivalent to the discrete ordinate approach on a uniform grid. The number M of velocity cells in each velocity dimension ranges from 9 to 61 for the piece-wise constant case and from 3 to 11 for the piece-wise quadratic case (corresponding to 9 and 33 nodal velocity points in each velocity dimension, respectively). To select the time step, simulation were run for about 2 μs for $s_u = s_v = s_w = 1$ and $M_u = M_v = M_w = 41$ and solutions were compared. The time step was reduced in each simulation until the discrete velocity solution converged within nine digits at its maximum. Thus we expect that the bulk of the error in the computed solutions can be attributed to the velocity discretizations. The results are summarized in Tables 3–8.

In Table 3 the L^2 norms of the relative errors $\varepsilon = \|f_{DG} - f_I\|_{L^2}$ are compared, where f_I is the initial data for the problem and f_{DG} is a DG approximation of the initial data. Integral norms are computed using a uniform velocity mesh and using Gauss quadratures inside each velocity cell. The total of 33 cells in each velocity dimension were used for integration and nine quadrature nodes in each velocity dimension were used inside each cell. This provided sufficient resolution to guarantee accuracy of the error up to two decimal digits in all simulations. The estimated order of convergence α is computed by assuming a power law for the error, $\varepsilon = CM^{-\alpha}$, and by comparing consecutive approximations. Here M is the number of velocity cells in one velocity dimension and C is a constant independent of M. As is expected, the L^2 errors converge

Table 3
Convergence of the L^2 norms of the relative error in approximations of the sum of two Maxwellian streams by piece-wise constant, $s_u = s_v = s_w = 1$, and piece-wise quadratic, $s_u = s_v = s_w = 3$, DG approximations. Here α is the estimated order of convergence. Note, however, that values of the discrete solutions at the velocity nodes are accurate up to roundoff errors.

DG vel. approx. $s = 1$			DG vel. approx. $s = 3$		
Num. cells, M	L^2 rel. error	α	Num. cells, M	L^2 rel. error	α
9	0.50		3	0.60	
15	0.30	1.00	5	0.14	2.85
21	0.22	0.95	7	0.05	2.98
33	0.14	0.90	11	0.01	2.81
41	0.11	1.13			
61	0.08	0.81			

Table 4
Convergence of the L^1 and L^∞ norms of the relative error in the DG solutions at time $t = 15.6$ μs. Solutions for the piece-wise quadratic DG velocity discretizations, $s_u = s_v = s_w = 3$, are compared to the target solution obtained for $s_u = s_v = s_w = 1$ on 61 velocity cells in each velocity dimension. The norms are computed using the mesh of the target solution.

DG vel. approx. $s = 3$				
Num. cells, M	L^1 rel. error	α	L^∞ rel. error	α
3	0.43		0.49	
5	7.1E−2	3.5	1.1E−1	3.0
7	1.8E−2	4.0	2.3E−2	4.6
11	6.2E−3	2.4	9.8E−3	1.9

with approximately first order in the piece-wise constant case, $s_u = s_v = s_w = 1$, and with approximately third order in the piece-wise quadratic case, $s_u = s_v = s_w = 3$. These results, however, need to be considered with a caution, because in both cases the approximate solutions are accurate within the machine precision at the velocity nodal points. Thus, with some proper interpolation, a better approximation may be achieved in the piece-wise constant case. Also, we notice that the selected resolution can hardly be considered accurate. In particular, the error of the piece-wise constant approximation is above 10% in most simulations. This is a consequence of the fact that our grid discretization has not been adapted to the problem and thus is not efficient. In the following, we will show that the accuracy performance may look different if one is interested only in a limited number of quantities computed from the distribution function, rather than in the distribution function itself. In particular, it is known that the density, bulk velocity and energy computed from a smooth distribution function using rectangle (or trapezoidal) rule on a uniform grid can appear to converge exponentially with respect to the grid size if the domain is sufficiently large.

Let us consider solutions computed for $s_u = s_v = s_w = 3$ at the time 15.6 μs which is approximately a third of the time required to reach the steady state. Since piece-wise quadratic DG approximations are third order accurate on smooth functions, one expects that the L^2 norm of the difference between the approximate solution and the exact solution will decrease with the third order or better. However, the exact solution is not known. Instead, we will use the solution corresponding to $s_u = s_v = s_w = 1$ and $M_u = M_v = M_w = 61$ as the most accurate solution. This solution will be denoted as f^*. We wish to compare piece-wise quadratic approximate solutions to the piece-wise constant (most accurate) solution f^*. The discussion of Table 3 suggests that the true L^2 norm of the difference will not be informative for establishing the accuracy of the method. Instead, we choose to evaluate the DG solutions at the nodal points of f^* and to compute the following two quantities,

$$\varepsilon_{L^1} = \frac{\sum_{i,j} |f^*_{i;j}(t) - f(t, v^j_i)|}{\sum_{i,j} |f^*_{i;j}(t)|} \quad \text{and} \quad \varepsilon_{L^\infty} = \frac{\max_{i,j} |f^*_{i;j}(t) - f(t, v^j_i)|}{\max_{i,j} |f^*_{i;j}(t)|},$$

where indices i and j run over all velocity nodes in the solution f^*. The results are summarized in Table 4. One can see that in two instances the convergence is faster than third order. This may be explained by the fact that differences are evaluated near the nodal points of the discrete solution where convergence may be accelerated. The lower than third order of convergence for $M_u = M_v = M_w = 11$ may be an indication that the simulations still did not reach the asymptotic regime of convergence.

Next we consider the convergence of the DG solutions to the exact steady state solution. Again, we evaluate the L^1 and L^∞ norms of the relative error. However, this time, we compute the errors using the solutions nodal points (thus eliminating the interpolation errors). The results are summarized in Table 5. It can be seen that errors are smaller at the steady state than the corresponding errors at the time 15.6 μs. This may be explained by the following two reasons. First, errors at the nodal points are expected to be smaller than errors at the interpolated points. Second, solutions to the problem of spatially homogeneous relaxation have a property of repairing itself. Indeed, as long as the density, bulk velocity and energy are preserved with a good accuracy in the approximate solutions, the steady state solution will be computed very accurately, even if it takes a longer (or a shorter) time to reach the steady state than that of the true solution. The loss of convergence

Table 5
Convergence of the L^1 and L^∞ norms of the relative error in the DG approximations of the steady state solution. Norms are computed using the solutions native meshes.

DG vel. approx. $s=1$					DG vel. approx. $s=3$				
Num. cells, N	L^1 rel. error	α	L^∞ rel. error	α	Num. cells, N	L^1 rel. error	α	L^∞ rel. error	α
9	2.4E−2		2.1E−2		3	2.9E−1		1.2E−1	
15	3.2E−3	3.9	5.1E−3	2.8	5	1.7E−2	5.6	2.7E−2	2.9
21	2.2E−3	1.1	3.6E−3	1.0	7	1.1E−3	8.2	1.1E−3	9.7
33	1.1E−3	1.5	1.9E−3	1.4	11	4.7E−4	1.8	7.5E−4	0.8
41	8.9E−4	1.1	1.4E−3	1.2					

Table 6
Relative errors in approximating the density, n, the directional temperature, T_u, and the moments $M_u^i = \int (u-\bar{u})^i f(t,u,v,w)\,du\,dv\,dw$, $i=4,6$ in the initial data. Here $\bar{u}$ is the first component of the vector of bulk velocity $\vec{\bar{u}} = (\bar{u},\bar{v},\bar{w})$. Moments are computed using the solutions native meshes.

N	n	α	T_u	α	M_u^4	α	M_u^6	α
DG velocity approximations $s=1$								
9	1.5E−2		8.7E−3		1.1E−2		3.2E−3	
15	7.4E−7	19.4	8.9E−5	9.0	4.8E−4	6.1	1.8E−3	1.1
21	6.7E−6	−6.6	8.5E−5	0.1	5.5E−4	−0.4	2.0E−3	−0.3
33	7.5E−6	−0.2	9.2E−5	−0.1	5.9E−4	−0.1	2.1E−3	−0.1
41	7.7E−6	−0.1	9.4E−5	−0.1	5.9E−4	−0.1	2.1E−3	−0.05
61	7.9E−6	−0.06	9.6E−5	−0.05	6.0E−4	−0.04	2.1E−3	−0.03
DG velocity approximations $s=3$								
3	1.2E−1		6.1E−2		8.6E−2		5.1E−3	
5	1.2E−2	4.5	3.6E−3	5.6	9.7E−3	4.3	4.9E−4	4.6
7	8.5E−4	8.0	2.1E−4	8.4	8.9E−4	7.2	5.4E−4	−0.3
11	2.2E−5	8.1	1.4E−4	0.8	5.6E−4	1.0	2.2E−3	−3.1

Table 7
Relative errors in approximating the density, n, the directional temperature, T_u, and the moments M_u^4 and M_u^6 in the solution at time $t=15.6$ μs. Moments are computed using the solutions native meshes. The solutions are compared to the solution using piece-wise constant approximations, $s_u=s_v=s_w=1$, on 61 velocity cells in each velocity dimension.

N	n	α	T_u	α	M_u^4	α	M_u^6	α
DG velocity approximations $s=1$								
9	1.5E−2		5.3E−3		1.0E−2		4.5E−2	
15	3.5E−5	11.9	2.7E−3	1.4	8.4E−3	0.36	1.9E−2	1.6
21	2.2E−5	1.4	1.4E−3	1.8	4.2E−3	2.1	8.6E−3	2.8
33	1.8E−5	0.5	5.5E−4	2.1	1.7E−3	2.0	3.6E−3	1.9
41	1.4E−5	1.0	2.6E−4	3.4	7.9E−4	3.5	1.7E−3	3.6
DG velocity approximations $s=3$								
3	1.3E−1		4.3E−2		5.9E−2		3.4E−1	
5	1.3E−2	4.6	4.7E−3	4.4	1.8E−2	2.3	1.1E−2	6.7
7	1.1E−2	7.3	3.4E−5	14.7	1.3E−3	7.7	4.7E−3	2.6
11	1.6E−4	4.4	1.1E−4	−2.7	4.5E−4	2.4	1.3E−3	2.8

for the solution $s_u=s_v=s_w=3$ and $M_u=M_v=M_w=11$ is most likely due to an accidental high accuracy of the solution $s_u=s_v=s_w=3$ and $M_u=M_v=M_w=7$. Also, not shown here, the convergence of the L^2 norms of the relative errors was very similar to results presented in Table 3. The L^2 errors were evaluated using $M_u=M_v=M_w=33$ velocity cells and nine Gauss points in each velocity dimension in each cell.

In conclusion, we consider the accuracy in approximating the density, $n(t)$, the directional temperature, $T_u(t)$, and the moments $M_u^i(t) = \int (u-\bar{u}(t))^i f(t,u,v,w)\,du\,dv\,dw$, $i=4,6$ in the computed solutions. Here $\bar{u}$ is the first component of the vector of bulk velocity $\vec{\bar{u}} = (\bar{u},\bar{v},\bar{w})$. All quantities of interest are computed from the approximate solutions using the solutions velocity nodes. The moments are compared at times $t=0.0$ μs, $t=15.6$ μs and $t=96.7$ μs. We notice that the solution has reached the steady state by the time $t=45$ μs. The results are summarized in Tables 6–8.

In Table 6, the relative errors in approximating the moments in the initial data are considered. Moments in the piece-wise constant approximations, $s_u=s_v=s_w=1$, quickly converge to their exact values within several digits and after that the convergence is lost. In contrast, approximations by piece-wise quadratic DG methods, $s_u=s_v=s_w=3$, converge slower, but still with a high order. A similar observation was made in [34,35] where the convergence of DG approximations was studied in the case of one velocity dimension. The fact that the density and the directional temperature only converged to five and four digits, respectively, is most likely caused by the truncation of the velocity domain. In the one-dimensional study [34,35] a much larger velocity domain was used and the moments in piece-wise constant DG approximations converged exponentially to the machine precision.

Table 8
Relative errors in approximating the density, n, the directional temperature, T_u, and the moments M_u^4 and M_u^6 in the final state solution. Moments are computed using the solutions native meshes.

N	n	α	T_u	α	M_u^4	α	M_u^6	α
DG velocity approximations $s=1$								
9	1.5E−2		4.2E−4		5.2E−4		5.9E−3	
15	6.5E−5	10.7	3.2E−3	−4.0	4.9E−3	−4.4	2.1E−3	2.1
21	4.6E−5	1.0	2.0E−3	1.3	2.4E−3	2.1	1.9E−3	0.2
33	3.8E−5	0.4	8.1E−4	2.0	8.3E−5	7.5	5.9E−3	−2.5
41	3.3E−5	0.6	5.1E−4	2.1	7.0E−4	−9.8	6.9E−3	−0.7
DG velocity approximations $s=3$								
3	1.5E−1		2.7E−2		0.1		5.7E−3	
5	1.3E−2	4.7	4.3E−3	3.6	2.2E−2	3.1	3.3E−2	−3.5
7	1.2E−3	7.1	3.1E−4	7.8	2.5E−3	6.5	9.2E−3	3.9
11	1.9E−4	4.1	1.6E−4	1.6	1.4E−3	1.2	8.0E−3	0.30

In Table 7, the accuracy of approximating the moments at the time 15.6 μs is summarized. Solutions with $s_u = s_v = s_w = 1$ and $s_u = s_v = s_w = 3$ are compared to the solution with $s_u = s_v = s_w = 1$ and $M_u = M_v = M_w = 61$. In this case, piece-wise constant solutions are still better in approximating the density. However, the piece-wise quadratic solutions are more efficient in approximating the directional temperature T_u and moments $M_u^{4,6}$. In particular the errors for $s_u = s_v = s_w = 3$ and $M_u = M_v = M_w = 11$ are less than the errors for $s_u = s_v = s_w = 1$ and $M_u = M_v = M_w = 33$. We believe that the piece-wise quadratic approximations generate a smoother kernel $A(\vec{v}, \vec{v}_1, \phi)$ than the piece-wise constant approximations. As a result, integration of the discrete collision operator is more accurate in the piece-wise quadratic case, thus giving an advantage to higher order methods.

In Table 8, the accuracy of approximating the moments in the steady state solution is considered. Again, piece-wise constant solutions are more efficient in approximating the density and the piece-wise quadratic solutions are more efficient in approximating the directional temperature T_u and the moments $M_u^{4,6}$. Overall, this comparison speaks in favor of using high order methods in applications where accuracy of higher moments is important.

8. Conclusion

We developed a discontinuous Galerkin discretization of the spatially homogeneous Boltzmann equation in the velocity space using a symmetric bilinear form of the Galerkin projection of the collision operator. The time-independent kernel of the bilinear operator carries the information about the geometry of the velocity discretization and about the collision model. Properties of the kernel were studied and several practical statements about the kernel symmetries were formulated. Evaluation of the kernel was implemented using an MPI parallelization algorithm that scales to a large number of processors. The approach was applied to the problem of spatially homogeneous relaxation. Discretizations with up to $M = 33$ degrees of freedom per velocity dimension were successfully tested on a single processor and discretizations up to $M = 61$ degrees of freedom were successfully tested on MPI parallel processors. DG solutions to the problem of spatially homogeneous relaxation showed an excellent agreement with the solution obtained by an established DSMC code. The solutions conserve mass momentum and temperature with a good accuracy.

The main obstacles to increasing the velocity resolution to hundreds of nodes per velocity dimension are the large amount of components in the pre-computed collision kernel and the large number of resulting arithmetic operations to evaluate the collision integral. These numbers were demonstrated to grow as $O(M^5)$ and $O(M^8)$, respectively for a single spatial cell, where M is the number of degrees of freedom in one velocity dimension. To overcome this problem, the authors are developing algorithms for parallelization of the evaluation of the collision integral to up to thousands of processors. An additional improvement is expected by developing adaptive Galerkin basis so as to minimize the total number of degrees of freedom in the velocity variable.

The discrete Galerkin form of the Boltzmann equation offers a unique insight to its numerical properties. In the future, the authors will apply techniques of numerical linear algebra and will explore the eigenstructure of the bilinear discrete collision kernel. They expect that this will give a new perspective on the familiar numerical properties of the solutions to the Boltzmann equation and will eventually lead to the development of new and more efficient approximation techniques.

Acknowledgements

The first author was supported by the NRC Resident Associate Program in 2011–2013. The authors would like to thank Professor I. Boyd and Dr. J. Burt for their interest in this work and for inspiring discussions. The authors also thank Professors S. Gimelshein, J. Shen, B. Lucier and V. Panferov for their interest in this work. The computer resources were provided by the U.S. Department of Defence, High Performance Computing, Defence Shared Resource Center at AFRL, Wright-Patterson AFB, Ohio. Additional computational resources were provided by the Rosen Center for Advanced Computing at Purdue University, by the Department of Mathematics at Purdue University and by the Extreme Science and Engineering Discovery Environ-

ment, supported by National Science Foundation Grant No. OCI-1053575. The authors cordially thank Dr. L.L. Foster for her help in proofreading the paper.

Appendix A. Sketches of the proofs of Lemmas 4.1–4.3

Lemma 2.1. *Let operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ be defined by (11) with all gas particles having the same mass and the potential of the particles interaction being spherically symmetric. Then $A(\vec{v}, \vec{v}_1; \phi_i^j)$ is symmetric with respect to $\vec{v}$ and $\vec{v}_1$, that is*

$$A\big(\vec{v}, \vec{v}_1; \phi_i^j\big) = A\big(\vec{v}_1, \vec{v}; \phi_i^j\big), \quad \forall \vec{v}, \vec{v}_1 \in \mathbb{R}^3.$$

Also,

$$A\big(\vec{v}, \vec{v}; \phi_i^j\big) = 0, \quad \forall \vec{v} \in \mathbb{R}^3.$$

Proof. Formula (13) follows immediately from (11) by noticing that if $\vec{v} = \vec{v}_1$ then $\vec{g} = 0$. Therefore, (11) is automatically zero.

To prove (12) we recall that in the case when all particles have the same mass, the interchange of pre-collision velocities $\vec{v}$ and $\vec{v}_1$ will result in the interchange of post-collision velocities $\vec{v}'$ and $\vec{v}_1'$. In particular, no different post-collision velocities result from the interchange of $\vec{v}$ and $\vec{v}_1$. The statement follows by noticing the symmetry of the expression under the integral (12) is both pairs of velocities.

For a more "mathematical" proof one may recall that in the case of a spherically symmetric potential the pre-collision velocities $\vec{v}$ and $\vec{v}_1$ and the post-collision velocities $\vec{v}'$ and $\vec{v}'$ of the molecules are located on a sphere with the center at $(\vec{v} + \vec{v}_1)/2$ and the radius of $|\vec{g}|/2$, see for example [42]. Furthermore, from some geometrical and physical considerations one concludes that $\vec{v}$, $\vec{v}_1$, $\vec{v}'$ and $\vec{v}'$ form a plane rectangle. Then the statement follows by observing that the rectangle is determined by $\vec{v}$ and $\vec{v}_1$ in a unique way. □

Lemma 2.2. *Let operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ be defined by (11) and let the potential of molecular interaction be dependent only on the distance between the particles. Then $\forall \xi \in \mathbb{R}^3$*

$$A\big(\vec{v} + \vec{\xi}, \vec{v}_1 + \vec{\xi}; \phi_i^j(\vec{v} - \vec{\xi})\big) = A\big(\vec{v}, \vec{v}_1; \phi_i^j\big).$$

Proof. Consider $A(\vec{v} + \vec{\xi}, \vec{v}_1 + \vec{\xi}; \phi_i^j(\vec{v} - \vec{\xi}))$. We clarify that these notations mean that vectors $\vec{v}$ and $\vec{v}_1$ in (11) are replaced with $\vec{v} + \vec{\xi}$ and $\vec{v}_1 + \vec{\xi}$ correspondingly and that basis function $\phi_i^j(\vec{v})$ is replaced with a "shifted" function $\phi_i^j(\vec{v} - \vec{\xi})$. We notice that the relative speed of the molecules with velocities $\vec{v} + \vec{\xi}$ and $\vec{v}_1 + \vec{\xi}$ is still $\vec{g} = \vec{v} + \vec{\xi} - (\vec{v}_1 + \vec{\xi}_1) = \vec{v} - \vec{v}_1$. Since the potential of molecular interaction depends only on the distance between the particles and in particular it does not depend on the particles individual velocities, the post-collision velocities will be $\vec{v}' + \vec{\xi}$ and $\vec{v}_1' + \vec{\xi}$. The rest of the statement follows by a direct substitution:

$$\begin{aligned}
&A\big(\vec{v} + \vec{\xi}, \vec{v}_1 + \vec{\xi}; \phi_i^j(\vec{v} - \vec{\xi})\big) \\
&= \frac{|\vec{g}|}{2} \int_0^{2\pi} \int_0^{b_*} \big(\phi_i^j\big(\big(\vec{v}' + \vec{\xi}\big) - \vec{\xi}\big) + \phi_i^j\big(\big(\vec{v}_1' + \vec{\xi}\big) - \vec{\xi}\big) - \phi_i^j\big((\vec{v} + \vec{\xi}) - \vec{\xi}\big) - \phi_i^j\big((\vec{v}_1 + \vec{\xi}) - \vec{\xi}\big)\big) b \, db \, d\varepsilon \\
&= \frac{|\vec{g}|}{2} \int_0^{2\pi} \int_0^{b_*} \big(\phi_i^j\big(\vec{v}'\big) + \phi_i^j\big(\vec{v}_1'\big) - \phi_i^j(\vec{v}) - \phi_i^j(\vec{v}_1)\big) b \, db \, d\varepsilon \\
&= A\big(\vec{v}, \vec{v}_1; \phi_i^j\big). \qquad \square
\end{aligned}$$

Lemma 2.3. *Let operator $A(\vec{v}, \vec{v}_1; \phi_i^j)$ be defined by (11) and let the potential of molecular interaction be dependent only on the distance between the particles. Let $S : \mathbb{R}^3 \to \mathbb{R}^3$ be a linear isometry of $\mathbb{R}^3$. Then*

$$A\big(S\vec{v}, S\vec{v}_1; \phi_i^j\big(S^{-1}\vec{u}\big)\big) = A\big(\vec{v}, \vec{v}_1; \phi_i^j\big).$$

Proof. *(We only present a sketch of the proof.)* We begin with a remark that will be helpful to highlight the mathematical structure of the proof. In the kinetic theory there is a connection between the velocity space and the physical space in that velocity vectors are describing motion in the physical space and that displacement vectors in physical space are naturally identified with points in the velocity space. By identifying this connection, we implicitly introduce an angle and orientation preserving affine transformation that maps the first axis of the positive coordinate triple in the physical space to the first

axis of the positive coordinate triple in the velocity space, the second axis into the second one and the third axis into the third one. As a result, we obtain a way to map vectors from the physical space to the velocity space and vice-versa. Connection between physical and velocity spaces can also be seen in definition (11). Indeed, post-collision velocities $\vec{v}'$ and $\vec{v}_1'$ are functions of the pre-collision velocities, $\vec{v}$ and $\vec{v}_1$, the impact parameters b and ε and the collision model. While the former belongs to the velocity space, the latter belongs to the physical space. Moreover, to define parameters b of the molecular shortest approach distance and ε of the relative angle of the collision plane, we must move the vector of relative velocity $\vec{g} = \vec{v} - \vec{v}_1$ into the physical space, e.g., [41], Section 1.3. Specifically, we define b as the distance from the center of the second molecule to the line coming through the center of the first molecule in the direction of their relative velocity $\vec{g}$. Also, the parameter ε is the angle between the collision plane formed by the image of the vector $\vec{g}$ in the physical space and the centers of the colliding molecules and a reference coordinate plane. Thus, to evaluate (11) we need to map vectors between physical and velocity spaces many times. Applying this reasoning to the statement of the theorem in question, we notice that the left side of (15) is evaluated relative to velocities $S\vec{v}$ and $S\vec{v}_1$ while the right side relative to velocities $\vec{v}$ and $\vec{v}_1$. Therefore separate sets of impact parameters are introduced for each case. What we are intended to show is that the result is the same for both cases.

Applying definition (11) to the left side of (15) we have

$$\begin{aligned} &A\big(S\vec{v}, S\vec{v}_1; \phi_i^j\big(S^{-1}\vec{u}\big)\big) \\ &\quad= \frac{|\vec{\tilde{g}}|}{2}\int_0^{2\pi}\int_0^{b_*}\big[\phi_i^j\big(S^{-1}(S\vec{v})'\big) + \phi_i^j\big(S^{-1}(S\vec{v}_1)'\big) - \phi_i^j\big(S^{-1}S\vec{v}\big) - \phi_i^j\big(S^{-1}S\vec{v}_1\big)\big]\tilde{b}\,d\tilde{b}\,d\tilde{\varepsilon} \\ &\quad= \frac{|\vec{g}|}{2}\int_0^{2\pi}\int_0^{b_*}\big[\phi_i^j\big(S^{-1}(S\vec{v})'\big) + \phi_i^j\big(S^{-1}(S\vec{v}_1)'\big) - \phi_i^j(\vec{v}) - \phi_i^j(\vec{v}_1)\big]\tilde{b}\,d\tilde{b}\,d\tilde{\varepsilon}, \end{aligned} \tag{A.1}$$

where $\vec{\tilde{g}} = S\vec{v} - S\vec{v}_1$ and $\tilde{b}$ and $\tilde{\varepsilon}$ are the impact parameters defined in the local coordinate system of the molecules with velocities $S\vec{v}$ and $S\vec{v}_1$. Notice that because S is linear, we have $\vec{\tilde{g}} = S\vec{v} - S\vec{v}_1 = S(\vec{v} - \vec{v}_1) = S\vec{g}$. Also, S is an isometry, therefore, $|\vec{\tilde{g}}| = |S\vec{g}| = |\vec{g}|$.

Let us now describe the local coordinate systems that are introduced for both pairs of molecules: the pair with velocities $S\vec{v}$ and $S\vec{v}_1$ and the pair with velocities $\vec{v}$ and $\vec{v}_1$. Let us consider the molecules with velocities $S\vec{v}$ and $S\vec{v}_1$, first. According to the formalism described above, we define $\tilde{b}$ as the distance from molecule with velocity $S\vec{v}_1$ to the line passing through the center of molecule with velocity $S\vec{v}$ in the direction of $\vec{\tilde{g}}$. We let the coordinate system be introduced for the pair of colliding molecules with velocities $S\vec{v}$ and $S\vec{v}_1$ so that its origin located at the center of the molecule with velocity $S\vec{v}$ and its $\tilde{\xi}_1$ axis directed along $\vec{\tilde{g}}$. Then the parameter $\tilde{b}$ is the distance to the $\tilde{\xi}_1$ axis. The parameter $\tilde{\varepsilon}$ of the angle of the collision plane is defined relevant to a reference plane. We let axes $\tilde{\xi}_2$ and $\tilde{\xi}_3$ be selected to make a right triple and designate the plane $\tilde{\xi}_1\tilde{\xi}_2$ as the reference plane. We note that the collision plane contains the $\tilde{\xi}_1$ axis and let the angle $\tilde{\varepsilon}$ between the reference plane and the collision plane be measured from $\tilde{\xi}_2$ axis toward the $\tilde{\xi}_3$ axis. Notice that these assumptions do not limit the generality of the argument because any admissible parametrization should produce the correct value of the integral above.

We will show now that a choice of the parameter ε can be made in a local system of coordinates corresponding to the pair of molecules colliding with velocities $\vec{v}$ and $\vec{v}_1$, such that $\vec{v}' = S^{-1}(S\vec{v})'$ and $\vec{v}_1' = S^{-1}(S\vec{v}_1)'$ as long as $b = \tilde{b}$ and $\varepsilon = \tilde{\varepsilon}$. We let the origin of the second set of coordinates be located at the center of molecule with velocity $\vec{v}$ and its ξ_1 axis directed along $\vec{g}$. Then parameter b gives the distance from the molecule with velocity $\vec{v}_1$ to the axis ξ_1. We define axes ξ_2 and ξ_3 to be the images of $\tilde{\xi}_2$ and $\tilde{\xi}_3$, respectively, under the action of S^{-1}. Specifically, using the mapping between the physical space and the velocity space we consider the unit vectors giving the directions of the axes $\tilde{\xi}_2$ and $\tilde{\xi}_3$ in the velocity space. Let these vectors be $\vec{\tilde{\tau}}_2$ and $\vec{\tilde{\tau}}_3$, respectively. Applying the inverse transformation S^{-1} to these vectors and moving back to physical space, we require that $\vec{\tau}_2 = S^{-1}\vec{\tilde{\tau}}_2$ and $\vec{\tau}_3 = S^{-1}\vec{\tilde{\tau}}_3$ be the unit vectors of the axes ξ_2 and ξ_3. It is a simple check that $\vec{\tau}_1 = S^{-1}\vec{\tilde{\tau}}_1$, where $\vec{\tilde{\tau}}_1 = \vec{\tilde{g}}/|\vec{\tilde{g}}|$ defines the $\tilde{\xi}_1$ axis.

Since the molecular potential is spherically symmetric, the relative velocity undergoes a specular reflection on the impact, see e.g. [42], Section 1.2. Furthermore, the trajectories of the colliding molecules are contained in the collision plane, see e.g., [41], Section 1.3. Thus the post-collision relative velocity belongs to the collision plane. Let us denote by $O[b, \varepsilon, |\vec{g}|](\vec{g}) : \mathbb{R}^3 \to \mathbb{R}^3$ the operator of flat rotation of the vector of relative velocity in collision plane. Notice that the rotation angle depends on the approach distance b, the collision model and on the norm of the relative velocity. It, however, does not depend on the direction of $\vec{g}$. Re-writing slightly the familiar formulas for the post-collision velocities (see, e.g., [42], Section 1.2), we have

$$\begin{aligned} \vec{v}' &= \vec{v} + \big(O\big[b, \varepsilon, |\vec{g}|\big](\vec{g}) - \vec{g}\big)/2, \\ \vec{v}_1' &= \vec{v}_1 - \big(O\big[b, \varepsilon, |\vec{g}|\big](\vec{g}) - \vec{g}\big)/2. \end{aligned}$$

Similarly the molecules with velocities $S\vec{v}$ and $S\vec{v}_1$ we have

$$(S\vec{v})' = S\vec{v} + \left(O\left[\tilde{b},\tilde{\varepsilon},|\vec{\tilde{g}}|\right](\vec{\tilde{g}}) - \vec{\tilde{g}}\right)/2,$$
$$(S\vec{v}_1)' = S\vec{v}_1 - \left(O\left[\tilde{b},\tilde{\varepsilon},|\vec{\tilde{g}}|\right](\vec{\tilde{g}}) - \vec{\tilde{g}}\right)/2.$$

Recalling that $|\vec{\tilde{g}}| = |\vec{g}|$, we obtain

$$(S\vec{v})' = S\vec{v} + \left(O\left[\tilde{b},\tilde{\varepsilon},|\vec{g}|\right](\vec{\tilde{g}}) - \vec{\tilde{g}}\right)/2,$$
$$(S\vec{v}_1)' = S\vec{v}_1 - \left(O\left[\tilde{b},\tilde{\varepsilon},|\vec{g}|\right](\vec{\tilde{g}}) - \vec{\tilde{g}}\right)/2. \tag{A.2}$$

We now consider the two vectors $O[\tilde{b},\tilde{\varepsilon},|\vec{g}|](\vec{\tilde{g}})$ and $S(O[b,\varepsilon,|\vec{g}|](\vec{g}))$. Because S is a linear isometry, it preserves angles between vectors. Therefore, for any $0 < n < b^*$ the plane formed by the vectors $S(O[b,\varepsilon,|\vec{g}|](\vec{g}))$, $\vec{\tilde{g}} = S\vec{g}$ will be making angle ε with the plane formed by vectors $\vec{\tilde{\tau}}_2 = S\vec{\tau}_2$ and $\vec{\tilde{\tau}}_3 = S\vec{\tau}_3$. Moreover, $S(O[b,\varepsilon,|\vec{g}|](\vec{g}))$ will make the same angle with vector $\vec{\tilde{g}} = S\vec{g}$ as $O[b,\varepsilon,|\vec{g}|](\vec{g})$ is making with $\vec{g}$. Recalling that the angle of flat rotation in (A.2) only depends on $\tilde{b}$ we conclude that if $\tilde{b} = b$ then $O[\tilde{b},\tilde{\varepsilon},|\vec{g}|](\vec{\tilde{g}})$ will make the same angle with $\vec{\tilde{g}}$ as $O[b,\varepsilon,|\vec{g}|](\vec{g})$ is making with $\vec{g}$. Noting that as long as $\tilde{\varepsilon} = \varepsilon$, both $S(O[b,\varepsilon,|\vec{g}|](\vec{g}))$ and $O[\tilde{b},\tilde{\varepsilon},|\vec{g}|](\vec{\tilde{g}})$ lay in the same plane, we conclude that they are the same vectors. Therefore, $S(O[b,\varepsilon,|\vec{g}|](\vec{g})) = O[\tilde{b},\tilde{\varepsilon},|\vec{g}|](\vec{\tilde{g}})$ and $O[b,\varepsilon,|\vec{g}|](\vec{g}) = S^{-1}(O[\tilde{b},\tilde{\varepsilon},|\vec{g}|](\vec{\tilde{g}}))$ as long as $\tilde{b} = b$ and $\tilde{\varepsilon} = \varepsilon$. By applying S^{-1} to both sides of (15) we conclude that $\vec{v}' = S^{-1}(S\vec{v})'$ and $\vec{v}_1' = S^{-1}(S\vec{v}_1)'$ as long as $b = \tilde{b}$ and $\varepsilon = \tilde{\varepsilon}$. The rest of the statement follows by introducing the substitution $\tilde{b} = b$ and $\tilde{\varepsilon} = \varepsilon$ in (A.1) and replacing $S^{-1}(S\vec{v})'$ and $S^{-1}(S\vec{v}_1)'$ with $\vec{v}'$ and $\vec{v}_1'$, respectively. □

We notice that in the case of a general molecular interaction potential, the angle of rotation in the collision plane included in $O[\tilde{b},\tilde{\varepsilon},|\vec{g}|]$, depends on the relative velocity. Because of this, it is unlikely that Lemma 4.3 can be extended to a contraction mapping S. We however anticipate that it is possible to develop analogs of formulas (15) by introducing an appropriate scaling parameter in the case of molecular potentials that do not depend on the relative velocity, e.g., for hard spheres. Such formulas would alleviate evaluation and storage of $A(\vec{v},\vec{v}_1;\phi_i^j)$.

References

[1] G. Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Clarendon Press, Oxford, 1994.
[2] A. Mohamad, Lattice Boltzmann Method: Fundamentals and Engineering Applications with Computer Codes, 2011.
[3] C. Buet, A discrete-velocity scheme for the Boltzmann operator of rarefied gas-dynamics, Transp. Theory Stat. Phys. 25 (1996) 33–60.
[4] L. Mieussens, Discrete-velocity models and numerical schemes for the Boltzmann-BGK equation in plane and axisymmetric geometries, J. Comput. Phys. 162 (2000) 429–466.
[5] V.A. Titarev, Efficient deterministic modeling of three-dimensional rarefied gas flows, Commun. Comput. Phys. 162 (2012) 162–192.
[6] H. Struchtrup, Macroscopic Transport Equations for Rarefied Gas Flows Approximation Methods in Kinetic Theory, Interaction of Mechanics and Mathematics Series, Springer, Heidelberg, 2005.
[7] F.G. Tcheremissine, Solution to the Boltzmann kinetic equation for high-speed flows, Comput. Math. Math. Phys. 46 (2006) 315–329.
[8] V. Kolobov, R. Arslanbekov, V. Aristov, A. Frolova, S. Zabelok, Unified solver for rarefied and continuum flows with adaptive mesh and algorithm refinement, J. Comput. Phys. 223 (2007) 589–608.
[9] V. Aristov, A. Frolova, S. Zabelok, Parallel algorithms of direct solving the Boltzmann equation in aerodynamics problems, in: A. Ecer, N. Satofuka, J. Periaux, P. Fox (Eds.), Parallel Computational Fluid Dynamics 2003, Elsevier, Amsterdam, 2004, pp. 49–56.
[10] E. Josyula, P. Vedula, W.F. Bailey, C.J. Suchyta III, Kinetic solution of the structure of a shock wave in a nonreactive gas mixture, Phys. Fluids 23 (2011) 017101.
[11] V.V. Aristov, S.A. Zabelok, A deterministic method for the solution of the Boltzmann equation with parallel computations, Zh. Vychisl. Mat. Mat. Fiz. 42 (2002) 425–437.
[12] L. Pareschi, B. Perthame, A Fourier spectral method for homogeneous Boltzmann equations, Transp. Theory Stat. Phys. 25 (1996) 369–382.
[13] L. Pareschi, G. Russo, On the stability of spectral methods for the homogeneous Boltzmann equation, Transp. Theory Stat. Phys. 29 (2000) 431–447.
[14] L. Pareschi, G. Russo, Numerical solution of the Boltzmann equation I: spectrally accurate approximation of the collision operator, SIAM J. Numer. Anal. 37 (2000) 1217–1245.
[15] A. Bobylev, S. Rjasanow, Fast deterministic method of solving the Boltzmann equation for hard spheres, Eur. J. Mech. B, Fluids 18 (1999) 869–887.
[16] C. Mouhot, L. Pareschi, Fast methods for the Boltzmann collision integral, C. R. Math. 339 (2004) 71–76.
[17] C. Mouhot, L. Pareschi, Fast algorithms for computing the Boltzmann collision operator, Math. Comput. 75 (2006) 1833–1852.
[18] L. Pareschi, Computational methods and fast algorithms for Boltzmann equation, in: N. Bellomo, R. Gatignol (Eds.), Lecture Notes on the Discretization of the Boltzmann Equation, in: Series on Advances in Mathematics and Applied Sciences, vol. 63, World Scientific, Singapore, 2003, pp. 159–202.
[19] F. Filbet, G. Russo, High order numerical methods for the space non-homogeneous Boltzmann equation, J. Comput. Phys. 186 (2003) 457–480.
[20] F. Filbet, C. Mouhot, L. Pareschi, Solving the Boltzmann equation in $n \log 2n$, SIAM J. Sci. Comput. 28 (2006) 1029–1053.
[21] C. Mouhot, L. Pareschi, T. Rey, Convolutive decomposition and fast summation methods for discrete-velocity approximations of the Boltzmann equation, Math. Model. Numer. Anal. 47 (2013) 1515–1531.
[22] R. Kirsch, S. Rjasanow, A weak formulation of the Boltzmann equation based on the Fourier transform, J. Stat. Phys. 129 (2007) 483–492.
[23] I.M. Gamba, S.H. Tharkabhushanam, Spectral-lagrangian methods for collisional models of non-equilibrium statistical states, J. Comput. Phys. 228 (2009) 2012–2036.
[24] I.M. Gamba, S.H. Tharkabhushanam, Shock and boundary structure formation by spectral-Lagrangian methods for the inhomogeneous Boltzmann transport equation, J. Comput. Math. (2010).
[25] J.R. Haack, I.M. Gamba, High performance computing with a conservative spectral Boltzmann solver, in: 28th International Symposium on Rarefied Gas Dynamics, AIP Conference Proceedings, Zaragoza, Spain, 9–13 July 2012, American Institute of Physics, 2012, pp. 334–341.
[26] V. Aristov, Direct Methods for Solving the Boltzmann Equation and Study of Nonequilibrium Flows, Fluid Mechanics and Its Applications, Kluwer Academic Publishers, 2001.

[27] P.L. Varghese, Arbitrary post-collision velocities in a discrete velocity scheme for the Boltzmann equation, in: M. Ivanov, A. Rebrov (Eds.), 25th International Symposium on Rarefied Gas Dynamics, 21–28 July 2006, Saint-Petersburg, Russia, Publishing House of Siberian Branch of RAS, Novosibirsk, Russia, 2007, pp. 227–232.
[28] V.A. Panferov, A.G. Heintz, A new consistent discrete-velocity model for the Boltzmann equation, Math. Models Methods Appl. Sci. 25 (2002) 571–593.
[29] E.A. Malkov, M.S. Ivanov, The phase space PIC method for solving the Boltzmann equation. Part I, in: 42nd AIAA Thermophysics Conference, in: AIP Conference Proceedings , vol. 3626, Honolulu, Hawaii, 27–30 June 2011, American Institute of Physics, 2011, p. 12.
[30] A. Morris, P. Varghese, D. Goldstein, Monte Carlo solution of the Boltzmann equation via a discrete velocity model, J. Comput. Phys. 230 (2011) 1265–1280.
[31] E.A. Malkov, M.S. Ivanov, S.O. Poleshkin, Discrete velocity scheme for solving the Boltzmann equation accelerated with the GP GPU, in: 28th International Symposium on Rarefied Gas Dynamics, AIP Conference Proceedings, Zaragoza, Spain, 9–13 July 2012, American Institute of Physics, 2012, pp. 318–325.
[32] G. Ghiroldi, L. Gibelli, P. Dagna, A. Invernizzi, Linearized Boltzmann equation: a preliminary exploration of its range of applicability, in: 28th International Symposium on Rarefied Gas Dynamics, AIP Conference Proceedings, Zaragoza, Spain, 9–13 July 2012, American Institute of Physics, 2012, pp. 735–741.
[33] A. Alekseenko, E. Josyula, Deterministic solution of the equation using a discontinuous Galerkin velocity discretization, in: 28th International Symposium on Rarefied Gas Dynamics, AIP Conference Proceedings, Zaragoza, Spain, 9–13 July 2012, American Institute of Physics, 2012, pp. 279–286.
[34] A.M. Alekseenko, Numerical properties of high order discrete velocity solutions to the BGK kinetic equation, Appl. Numer. Math. 61 (2011) 410–427.
[35] A. Alekseenko, N. Gimelshein, S. Gimelshein, An application of discontinuous Galerkin space and velocity discretisations to the solution of a model kinetic equation, Int. J. Comput. Fluid Dyn. 26 (2012) 145–161.
[36] J. Hesthaven, T. Warburton, Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications, Texts in Applied Mathematics, Springer, 2007.
[37] A. Bobylev, Moment inequalities for the Boltzmann equation and applications to spatially homogeneous problems, J. Stat. Phys. 88 (1997) 1183–1214.
[38] I.D. Boyd, Vectorization of a Monte Carlo simulation scheme for nonequilibrium gas dynamics, J. Comput. Phys. 96 (1991) 411–427.
[39] A. Majorana, A numerical model of the Boltzmann equation related to the discontinuous Galerkin method, Kinet. Relat. Models 4 (2011) 139–151.
[40] M.K. Gobbert, T.S. Cale, 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 (2007) 237–273.
[41] M. Kogan, Rarefied Gas Dynamics, Plenum Press, New York, USA, 1969.
[42] C. Cercignani, Rarefied Gas Dynamics: From Basic Concepts to Actual Calculations, Cambridge University Press, Cambridge, UK, 2000.
[43] R.O. Fox, P. Vedula, Quadrature-based moment model for moderately dense polydisperse gas-particle flows, Ind. Eng. Chem. Res. 49 (2010) 5174–5187.
[44] B.I. Green, P. Vedula, Validation of a collisional lattice Boltzmann method, in: 20th AIAA Computational Fluid Dynamics Conference, AIP Conference Proceedings, vol. 3403, Honolulu Hawaii, 27–30 June 2011, American Institute of Physics, 2011, p. 14.
[45] V.K. Gupta, M. Torrilhon, Automated Boltzmann collision integrals for moment equations, in: 28th International Symposium on Rarefied Gas Dynamics, AIP Conference Proceedings, Zaragoza, Spain, 9–13 July 2012, American Institute of Physics, 2012, pp. 67–74.
[46] M. Frigo, C. Leiserson, H. Prokop, S. Ramachandran, Cache oblivious algorithms, in: 40th IEE Annual Symposium on Foundations of Computer Science, New York, October 1999, pp. 285–297.
[47] R. Burden, J. Faires, Numerical Analysis, Thomson Brooks/Cole, 2005.
[48] A. Palczewski, J. Schneider, A.V. Bobylev, A consistency result for a discrete-velocity model of the Boltzmann equation, SIAM J. Numer. Anal. 34 (1997) 1865–1883.
[49] T. Platkowski, R. Illner, Discrete velocity models of the Boltzmann equation: a survey on the mathematical aspects of the theory, SIAM Rev. 30 (1988) 213–255.