Kinetic and Related Models
©American Institute of Mathematical Sciences
Volume **11**, Number **5**, October **2018**

doi:10.3934/krm.2018047

pp. **1211–1234**

A DETERMINISTIC-STOCHASTIC METHOD FOR COMPUTING THE BOLTZMANN COLLISION INTEGRAL IN $\mathcal{O}(MN)$ OPERATIONS

Alexander Alekseenko*

Department of Mathematics
California State University Northridge
Northridge, CA 91330, USA

Truong Nguyen

Department of Mathematics & Statistics
Wright State University
Dayton, OH 45435, USA

Aihua Wood

Department of Mathematics & Statistics
Air Force Institute of Technology
WPAFB, OH 45433, USA

(Communicated by Lorenzo Pareschi)

Abstract. We developed and implemented a numerical algorithm for evaluating the Boltzmann collision integral with $O(MN)$ operations, where N is the number of the discrete velocity points and $M < N$. At the base of the algorithm are nodal-discontinuous Galerkin discretizations of the collision operator on uniform grids and a bilinear convolution form of the Galerkin projection of the collision operator. Efficiency of the algorithm is achieved by applying singular value decomposition compression of the discrete collision kernel and by approximating the kinetic solution by a sum of Maxwellian streams using a stochastic likelihood maximization algorithm. Accuracy of the method is established on solutions to the problem of spatially homogeneous relaxation.

1. **Introduction.** A number of problems have emerged recently in aerodynamics that require accurate simulations of gas flows when the gas is in non-continuum state and when the flow is time-dependent or when the flow velocities are smaller than the gas mean thermal velocity. Such flows are formed around leading edges of hypersonic vehicles and in channels of microelectromechanical devices that use gas as its primary working fluids. A number of simulation techniques are available for non-continuum gas flows, including the most widely used direct simulation Monte-Carlo (DSMC) approach see, e.g., [10, 14, 29, 46, 15], the model kinetic equations

2010 *Mathematics Subject Classification.* Primary: 76P05, 76M10; Secondary: 65M60.

Key words and phrases. Kinetic equations, collision integral, nodal-discontinuous Galerkin discretizations, convolution formulation, dynamics of non-continuum gas.

The first author was supported by NSF grant DMS-1620497 and HPTi PETTT grant PP-SAS-KY06-001. The third author was supported by AFOSR Grant No. F4FGA04296J00.

* Corresponding author: alexander.alekseenko@csun.edu.

approach [9, 28, 45, 44, 51, 43], the moment method and the extended hydrodynamics approach [34, 47], the lattice-Boltzmann method [48], as well as other methods [4, 23, 27]. However, most of these approaches are difficult to apply to simulations of time dependent and slow non-continuum flows, albeit for different reasons. The physically accurate DSMC approach becomes prohibitively expensive due to the large simulation time necessary to resolve the statistical noise. The other methods are simplified descriptions of gas and have limited accuracy in the non-continuum regime.

A possible alternative to DSMC is to employ distributed velocities in collision selection and modeling as opposed to the point measure approximation to the distribution function inherent to DSMC. To this end, distributional Monte Carlo methods have been proposed using kernel density estimation for space homogeneous Boltzmann equation [42, 41].

Another alternative to DSMC methods is given by the deterministic solution of the kinetic Boltzmann equation. For a review on recent developments in this area see, e.g., [17, 38, 6]. The difficulty in solving the Boltzmann equation is the evaluation of the five-fold collision integral describing interactions of the gas molecules. Methods have been proposed to solve the Boltzmann equation using the direct discretization of the collision operator in the velocity variable (see, e.g., [6, 7, 39, 8]). Computational costs of the direct methods grow very rapidly, usually, at least as $O(n^8)$, where n is the number of velocity points in one velocity dimension. As a result, direct fully deterministic methods can only be applied to problems that do not require a large number of spatial discretization points.

An efficient method for solution of the Boltzmann equation was derived in [49] by formally applying a Galerkin discretization in the velocity variable using a basis of Dirac delta-functions defined at the nodes of a uniform velocity grid. Efficient evaluation of the collision integral is achieved by introducing quasi-stochastic Korobov integration. The method has been applied to simulation of multicomponent gas mixtures and gases with internal energies in multidimensional applications [50, 32]. However, the method is still difficult to use for simulation of flows with a very high degree of non-equilibrium, e.g., high Mach number shock waves.

A number of approaches to solve the Boltzmann equation were obtained by applying Galerkin spectral discretizations in the velocity variable. In [40] a Galerkin discretization was constructed using the Fourier basis functions. By exploring properties of exponentials, the authors were able to reduce the complexity of evaluation of the collision operator to only $O(k^6)$ operations, where k is the number of the Fourier basis functions used in each velocity dimension. A closely related approach based on an application of the Fourier transform to the collision integral can be found in [31, 24, 25]. The method has computational complexity $O(k^6)$ and, similarly to the Galerkin spectral methods, is derived using properties of exponentials.

In many applications of non-continuum flows, e.g., three dimensional flows and flows of multicomponent gases, additional efficiency is highly desired in discretizations of the collision operator. In [40] it is pointed out that integrals in the Fourier image of the discrete collision operator have reduced dimensionality in the case of Maxwell interaction potential. The bulk of the computational costs in this case is associated with the evaluation of the discrete Fourier transform, leading to a method with $O(k^3 \log k)$ operations. In [13, 12] Carleman representation of the collision operator was used to separate one-dimensional convolutions of the distribution function inside multidimensional integral leading to an approach with

$O(Mk^3 \log k)$ operations. Here M is related to the number of pairs of directions of one-dimensional convolutions used in approximation of the collision integral. The idea of using Carleman representation was extended in [36, 20, 19] leading to an approach applicable to multidimensional simulations [52], flows of gas mixtures [53], and flows of gas with internal energies [37]. A drawback of the methods is lack of adaptivity in the velocity space since methods use global Fourier basis. In [22] a hyperbolic cross approximation of the solution in the frequency space was proposed, further increasing efficiency of the approach using Carleman representation and introducing adaptivity in spectral methods. However, incomplete spectral representation is hard to combine with the use of fast Fourier transform which may reduce the method's speed.

An approach using discontinuous-Galerkin (DG) discretizations of the Boltzmann equation in the velocity variable was proposed in [35, 2, 3, 26]. The high order DG approximations are well suited for approximating discontinuous and high gradient solutions. In addition, the method is easy to parallelize to a large number of processors and it requires only $O(n^5)$ units of memory storage for the components of the pre-computed kernel of the collision operator. However, the approach is computationally expensive. It requires $O(n^8)$ arithmetic operations, which is significantly slower than the spectral Galerkin methods. In this paper, we introduce an approximate method that is built on the DG discretization of [3] and that requires at most $O(n^6)$ operations. The new method adds the following three elements. First, the DG velocity discretization of the collision operator on uniform grid is re-written in a form of convolution. Second, singular value decomposition (SVD) is used to develop low rank approximation of the collision operator. Third, the kinetic solution is represented at each time step as a sum of continuum streams, each having homogeneous Gaussian densities. In the new method, the collision operator is replaced with a truncated sum of three-dimensional convolutions of homogeneous Gaussian densities with singular vectors of the collision kernel. The convolutions are pre-computed and stored in a large, $O(kn^6)$, database, where k is the number of tabulated values of temperature. With this database, evaluation of the collision operator is performed in $O(Mn^3)$, $M < n^3$ operations (mostly memory operations) by combining pre-computed results for each pair of interacting Maxwellian densities.

The new approach is very different in spirit from the fast spectral Galerkin approaches and to our knowledge has not been tried before. The first important distinction is the formalism that is used to derive the method. The proposed convolution formulation follows from invariance properties of the collision operator with respect to shift in the velocity variable and does not pose immediate restrictions on the choice of the Galerkin basis. The convolution formulation is incorporated in the DG method on uniform grid and extensions to octree meshes are possible. In contrast, the spectral Galerkin approach uses Carleman representation and global exponential functions to derive the convolution form. In this sense, the new approach is more general. The second distinction is in the way in which the efficiency of the method is achieved. In the new approach, parts of the computations are eliminated by using SVD truncation, i.e., we use insight of linear algebra to achieve efficiency.

Techniques of maximum likelihood estimation are used in our approach to approximate the solution by a sum of Maxwellian densities. It is observed that the projection step is not very accurate, with significant variations present in macroparameters of the approximating streams. However, the resulting approximation is

extremely compact in the sense that the entire kinetic solution is described using only a handful of degrees of freedom. To compensate for errors in the stochastic projection, conservation of mass, momentum and energy is enforced similarly to [13, 25, 26]. The new method shows a considerable acceleration in speed compared to that of [3], however, it is still significantly slower than fast spectral Galerkin methods. Partly it is because the method is not optimized for speed. But perhaps the most important restriction for the approach is the $O(Mn^3)$ memory operations. Memory operations are more expensive than arithmetic operations. As a result, it is not clear if the approach as a whole is promising for simulations of multidimensional flows in the near future. However, approximations of the solution by a sum of Maxwellian streams have a promise to provide a stepping stone for ultra-sparse approximations of kinetic solutions. The new method was implemented in the case of zero spatial dimension and applied to the solution of the problem of spatially homogeneous relaxation. Properties of the eigenvalues and eigenvectors of the collision kernel and the accuracy of the singular value decomposition were investigated. Solutions obtained by full DG velocity discretization [3] were used to validate the continuum stream approximation approach.

The paper is organized as follows. In Section 2 we briefly review the kinetic Boltzmann equation and summarize the nodal-DG velocity discretization of [3]. In Section 3 we formulate the new numerical method. The results of numerical simulation and accuracy analysis are presented in Section 4.

2. The nodal-DG velocity discretization.

2.1. 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

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

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 is given by

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

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 between the pair, b^* is the maximum distance at which interaction between molecules is non-negligible, 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 molecules. 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. The exact dependencies for $\vec{v}'$ and $\vec{v}_1'$ are not important for the formulation of the method. 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.

2.2. **Discontinuous Galerkin velocity discretizations of the collision operator.** The method introduced in this paper is an extension and a generalization of the nodal-DG velocity discretization of [2, 3]. Here we briefly summarize the original method.

We select a rectangular parallelepiped in the velocity space that is sufficiently large so that contributions of the molecular distribution function to the first few moments outside of this parallelepiped are negligible. We partition the region into rectangular parallelepipeds K_j. In this paper, only uniform partitions are considered. However, generalizations of the method can be proposed for octree partitions.

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,

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

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.

Lemma 2.1. [3] *The following identities hold for basis functions $\phi_i^j(\vec{v})$:*

$$\int_{K_j} \phi_p^j(\vec{v})\phi_q^j(\vec{v})\,d\vec{v} = \frac{\Delta\vec{v}^j}{8}\omega_p\delta_{pq}, \qquad \int_{K_j} \vec{v}\phi_p^j(\vec{v})\phi_q^j(\vec{v})\,d\vec{v} = \frac{\Delta\vec{v}^j}{8}\vec{v}_p^j\omega_p\delta_{pq}, \tag{3}$$

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 $\vec{v}_p^j = (\kappa_l^{j,u}, \kappa_m^{j,v}, \kappa_n^{j,w})$.

To derive a discrete form of the collision operator, we assume that on each K_j, the solution can be approximated by

$$f(t, \vec{x}, \vec{v})|_{K_j} = \sum_{i=1}^{s} f_{i;j}(t, \vec{x})\phi_i^j(\vec{v})\,. \tag{4}$$

The DG velocity discretization of the Boltzmann equation results from substituting the representation (4) into (1) and multiplying the result by a test basis function and integrating over K_j. Repeating this for all K_j and using identities (3) we arrive at

$$\partial_t f_{i;j}(t, \vec{x}) + \vec{v}_i^j \cdot \nabla_x f_{i;j}(t, \vec{x}) = I_{\phi_i^j}\,, \tag{5}$$

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

$$I_{\phi_i^j} = \frac{8}{\omega_i \Delta\vec{v}^j}\int_{K_j} \phi_i^j(\vec{v}) I[f](t, \vec{x}, \vec{v})\,d\vec{v}\,. \tag{6}$$

Following the approach of [2, 3, 35], we rewrite the DG projection of the collision operator $I_{\phi_i^j}$ in the form of a bilinear integral operator with a time-independent kernel. Specifically,

$$I_{\phi_i^j} = \frac{8}{\omega_i \Delta \vec{v}^j} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f(t,\vec{x},\vec{v}) f(t,\vec{x},\vec{v}_1) A(\vec{v},\vec{v}_1;\phi_i^j) d\vec{v}_1\, d\vec{v}, \tag{7}$$

where

$$A(\vec{v},\vec{v}_1;\phi_i^j) = \frac{|\vec{g}|}{2} \int_0^{2\pi} \int_0^{b^*} (\phi_i^j(\vec{v}') + \phi_i^j(\vec{v}_1') - \phi_i^j(\vec{v}) - \phi_i^j(\vec{v}_1)) b\, db\, d\varepsilon\,. \tag{8}$$

We notice that the kernel $A(\vec{v},\vec{v}_1;\phi_i^j)$ is independent of time and can be pre-computed. In [3], a number of symmetries of $A(\vec{v},\vec{v}_1;\phi_i^j)$ have been considered that allow to store it in $O(n^5)$ memory units in the cases when partitions of the velocity domain are uniform and the same basis functions are used on each element. Evaluation of the collision integral (7) in [3] requires $O(n^8)$ operations for each spatial point, which is slow. However, an efficient parallelization allowed the authors to achieve resolution of 61 points in each velocity dimension in the case of zero spatial dimensions solutions. In the next section, we will describe an approach to make evaluation of the collision integral more efficient, with only $O(n^6)$ operations.

3. Formulation of the accelerated numerical method.

3.1. Re-writing the collision operator in the form of a convolution.

An important generalization of the Galerkin discretization of the collision operator consists of rewriting it in the form of convolution. Let the elements K_j be uniform and let the same basis functions be used on each element. We can select a cell K_c and designate the basis functions $\phi_i^c(\vec{v})$ on K_c as the generating basis functions. Basis functions $\phi_i^j(\vec{v})$ on the rest of the cells can be restored using a substitution of the velocity variable, namely $\phi_i^j(\vec{v}) = \phi_i^c(\vec{v} + \vec{\xi}^j)$ where $\vec{\xi}^j \in \mathbb{R}^3$ is the vector that connects the center of K_j to the center of K_c.

We recall that operator $A(\vec{v}_1,\vec{v},\phi_i^c)$ is invariant with respect to translations [3], and rewrite (7) as follows:

$$\begin{aligned} I_{\phi_i^j} &= \frac{8}{\omega_i \Delta \vec{v}^j} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f(t,\vec{x},\vec{v}) f(t,\vec{x},\vec{v}_1) A(\vec{v}+\vec{\xi}^j, \vec{v}_1+\vec{\xi}^j; \phi_i^j(\vec{u}-\vec{\xi}^j))\, d\vec{v} d\vec{v}_1 \\ &= \frac{8}{\omega_i \Delta \vec{v}^j} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f(t,\vec{x},\vec{v}) f(t,\vec{x},\vec{v}_1) A(\vec{v}+\vec{\xi}^j, \vec{v}_1+\vec{\xi}^j; \phi_i^c(\vec{u}))\, d\vec{v} d\vec{v}_1\,. \end{aligned} \tag{9}$$

Performing the substitutions $\vec{\hat{v}} = \vec{v} + \vec{\xi}^j$ and $\vec{\hat{v}}_1 = \vec{v}_1 + \vec{\xi}^j$ in (9), we have

$$I_{\phi_i^j} = \frac{8}{\omega_i \Delta \vec{v}^j} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f(t,\vec{x},\vec{\hat{v}}-\vec{\xi}^j) f(t,\vec{x},\vec{\hat{v}}_1-\vec{\xi}^j) A(\vec{\hat{v}},\vec{\hat{v}}_1;\phi_i^c(\vec{u}))\, d\vec{\hat{v}} d\vec{\hat{v}}_1\,.$$

We note that the last formula allows to relate Galerkin projections of the collision operator on all basis functions for all cells to projections on the basis functions on cell K_c. This mechanism is used in [2, 3, 26] to reduce the storage requirement for pre-computed values of $A(\vec{v}_1,\vec{v},\phi_i^j)$ by only storing the components of $A(\vec{v}_1,\vec{v},\phi_i^c)$ corresponding to basis functions on cell K_c. In this paper, however, we propose to make an additional step and introduce a bilinear convolution operator

$$I_i(\vec{\xi}) = \frac{8}{\omega_i \Delta \vec{v}} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f(t,\vec{x},\vec{v}-\vec{\xi}) f(t,\vec{x},\vec{v}_1-\vec{\xi}) A(\vec{v},\vec{v}_1;\phi_i^c)\, d\vec{v} d\vec{v}_1\,, \tag{10}$$

and notice that $I_{\phi_i^j}$ can be obtained from (10) as $I_{\phi_i^j} = I_i(\vec{\xi}^j)$. In the following, we will refer to (10) as the convolution form of the Galerkin projection of the collision integral.

We remark that the form (10) of the collision operator is not very instructive for discretizations using global basis. However, it is extremely promising for velocity discretizations using locally supported bases on uniform grids, e.g. finite element methods and discontinuous Galerkin methods. Indeed, on uniform grids, the convolution form can be used to accelerate evaluation of (10) using the fast discrete Fourier transform. The expected number of operations in this case is $O(n^6)$. The development of these methods and their generalizations to octree meshes will be the authors' future work. In this paper, however, we will attempt to develop a method with less than $O(n^6)$ operations that provides only a crude approximation of the solution, but is fast and can be made more accurate at the expense of speed. To design such a method we explore the bilinear structure of (10), construct low rank approximation of (10) and use a special ansatz for the solution.

3.2. **Approximation of the kinetic solution by a linear combination of Maxwellian distributions.** In this section we introduce a further simplification to our model. We will assume that the unknown velocity distribution function $f(t, \vec{x}, \vec{v})$ is reasonably well approximated by a sum of a small number, $p \leq 5$, of Maxwellian streams, namely

$$f(t, \vec{x}, \vec{v}) = \sum_{i=1}^{p} f_{M_i}(t, \vec{x}, \vec{v})\,. \tag{11}$$

The velocity distribution function of a dimensionless Maxwellian stream has the form

$$f_{M_i}(t, \vec{x}, \vec{v}) = \frac{n_i(t, \vec{x})}{\big(\pi\, T_i(t, \vec{x})\big)^{\frac{3}{2}}} \exp\left(- \frac{|\vec{v} - \bar{\vec{u}}_i(t, \vec{x})|^2}{T_i(t, \vec{x})} \right),$$

where $\vec{u} = (u_1, u_2, u_3) \in \mathbb{R}^3$ and $|\vec{u}| = (u_1^2 + u_2^2 + u_3^2)^{\frac{1}{2}}$. The functions $n_i(t, \vec{x})$, $T_i(t, \vec{x})$, and $\bar{\vec{u}}_i(t, \vec{x})$ are the dimensionless density, temperature, and bulk velocity of the i^{th} stream, respectively. Substituting (11) into (10), we obtain:

$$\begin{aligned} I_i(\vec{\xi}) = \frac{8}{\omega_i \Delta \vec{v}} \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} & \left(\sum_{i'=1}^{p} f_{M_{i'}}(t, \vec{x}, \vec{v} - \vec{\xi}) \right) \\ & \times \left(\sum_{i''=1}^{p} f_{M_{i''}}(t, \vec{x}, \vec{v}_1 - \vec{\xi}) \right) A(\vec{v}, \vec{v}_1; \phi_i^c) d\vec{v} d\vec{v}_1 \\ = \frac{8}{\omega_i \Delta \vec{v}} \sum_{\substack{i' \neq i'' \\ i', i''=1}}^{p} & \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f_{M_{i'}}(t, \vec{x}, \vec{v} - \vec{\xi})\, f_{M_{i''}}(t, \vec{x}, \vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \phi_i^c)\, d\vec{v} d\vec{v}_1 \\ = \frac{8}{\omega_i \Delta \vec{v}} \sum_{\substack{i' \neq i'' \\ i', i''=1}}^{p} & I_i^{i'i''}(\vec{\xi})\,, \end{aligned} \tag{12}$$

where

$$I_i^{i'i''}(\vec{\xi}) = \int_{\mathbb{R}^3} \int_{\mathbb{R}^3} f_{M_{i'}}(t, \vec{x}, \vec{v} - \vec{\xi})\, f_{M_{i''}}(t, \vec{x}, \vec{v}_1 - \vec{\xi}) A(\vec{v}, \vec{v}_1; \phi_i^c)\, d\vec{v} d\vec{v}_1\,. \tag{13}$$

Notice that $I_i^{i'i''}(\vec{\xi}) = 0$ when $i' = i''$ since Maxwellian distributions are stationary solutions to the Boltzmann equation.

Similar to the approach of [3], we discretize (13) by replacing the three dimensional integrals with the Gaussian quadratures associated with the DG approximations (4). Notice that the same Gaussian nodes are used for the velocity variables $\vec{v}$ and $\vec{v}_1$. Then (13) becomes:

$$I_i^{i'i''}(\vec{\xi}) = \sum_{\alpha=1}^{N}\sum_{\beta=1}^{N} f_{M_{i'}}(t,\vec{x},\vec{v}_\alpha - \vec{\xi})\, f_{M_{i''}}(t,\vec{x},\vec{v}_\beta - \vec{\xi})\, A^i_{\alpha\beta}\,,$$

where $A^i_{\alpha\beta} = \dfrac{\omega_\alpha \Delta\vec{v}}{8}\dfrac{\omega_\beta \Delta\vec{v}}{8} A(\vec{v}_\alpha, \vec{v}_\beta; \phi_i^c)$. Here, N is the total number of velocity nodes in the velocity domain, and the indices α and β run over all velocity nodes. In the following, we will assume that a correspondence is established between index α and a pair of indices k, j so that $\vec{v}_\alpha = \vec{v}_k^j$. With this assumption, we shall write that $\vec{v}_\alpha$ is contained in the element $K_{j(\alpha)}$ and that k is the index of node $\vec{v}_\alpha = \vec{v}_k^j$ on $K_{j(\alpha)}$.

We recall that operator $A(\vec{v}_1, \vec{v}; \phi_i^c)$ is symmetric with respect to $\vec{v}_1$ and $\vec{v}$. Therefore, the quantity $A^i_{\alpha\beta}$ is a symmetric matrix, i.e., $A^i_{\alpha\beta} = A^i_{\beta\alpha}$, since we agreed that the same velocity nodes are used in the discretization of the integrals in (13). In this case, $A^i_{\alpha\beta}$ has the following representation

$$A^i_{\alpha\beta} = \sum_{l=1}^{N} \lambda_l^i\, \Psi^i_{l\alpha}\, \Psi^i_{l\beta}\,,$$

where λ_l^i, $l = 1, \ldots, N$, are the eigenvalues and $(\Psi^i_{l1}, \Psi^i_{l2}, \ldots, \Psi^i_{lN}) = \vec{\Psi}^i_l$, $l = 1, \ldots, N$, are the corresponding orthonormal eigenvectors. We will assume that the eigenvalues are ordered in the decreasing order by the absolute value, i.e, $|\lambda_1^i| \geq |\lambda_2^i| \geq \ldots \geq |\lambda_N^i|$.

Consider a function $\Psi_l^i(\vec{v})$ that is obtained from vector $\vec{\Psi}_l^i$ by applying the DG interpolation (4), that is

$$\Psi_l^i(\vec{v})|_{K_j} = \sum_{k=1}^{s} \Psi^i_{l,k,j}\, \phi_k^j(\vec{v})\,,$$

where $\Psi^i_{l,k,j} := \Psi^i_{l\alpha}$. Using the single index notation, we rewrite the above equation as

$$\Psi_l^i(\vec{v}) = \sum_{\alpha=1}^{N} \Psi^i_{l\alpha}\, \phi_\alpha(\vec{v})\,,$$

where $\phi_\alpha(\vec{v}) := \phi_k^j(\vec{v})$. Notice that, by construction we have, $\Psi_l^i(\vec{v}_\alpha) = \Psi^i_{l\alpha}$. That is, the values of $\Psi_l^i(\vec{v})$ at the nodes $\vec{v}_\alpha$ are equal to $\Psi^i_{l\alpha}$. Finally, we introduce the discrete representation for $A(\vec{v}, \vec{v}_1; \phi_i^c)$ that we will use:

$$A(\vec{v}, \vec{v}_1; \phi_i^c) = \sum_{k=1}^{N} \lambda_k^i \Psi_k^i(\vec{v}) \Psi_k^i(\vec{v_1}) + \varepsilon_{DG}\,, \tag{14}$$

where the quantity ε_{DG} indicates the truncation errors of nodal DG approximations of $A(\vec{v}, \vec{v}_1; \phi_i^c)$. Up to truncation errors of the Gaussian quadratures and errors of the nodal DG discretization in (4), we have

$$\int_{\mathbb{R}^3}\int_{\mathbb{R}^3} f(t,\vec{x},\vec{v})f(t,\vec{x},\vec{v}_1)A(\vec{v},\vec{v}_1;\phi_i^c)d\vec{v}d\vec{v}_1$$
$$= \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} f(t,\vec{x},\vec{v})f(t,\vec{x},\vec{v}_1)\Big(\sum_{k=1}^{N}\lambda_k^i\Psi_k^i(\vec{v})\Psi_k^i(\vec{v_1})\Big)d\vec{v}d\vec{v}_1\,. \quad (15)$$

Another way to interpret (15) is to introduce bilinear operators

$$\mathcal{A}(f,g) := \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} f(t,\vec{x},\vec{v})g(t,\vec{x},\vec{v}_1)A(\vec{v},\vec{v}_1;\phi_i^c)\,d\vec{v}d\vec{v}_1\,,$$

and

$$\hat{\mathcal{A}}(f,g) := \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} f(t,\vec{x},\vec{v})g(t,\vec{x},\vec{v}_1)\Big(\sum_{k=1}^{N}\lambda_k^i\Psi_k^i(\vec{v})\Psi_k^i(\vec{v_1})\Big)\,d\vec{v}d\vec{v}_1\,.$$

Note that $\mathcal{A} : H\times H\to\mathbb{R}$ where H is an appropriate space of all admissible solutions to the Boltzmann equation. Also, $\hat{\mathcal{A}} : V\times V\to\mathbb{R}$ where V is the finitely dimensional space of the nodal DG approximations. Moreover, λ_k^i and $\Psi_k^i(\vec{v})$ are the eigenvalues and eigenvectors of the linear operator $\hat{\mathfrak{a}} : V\to V$ such that

$$\hat{\mathfrak{a}}(f) := \int_{\mathbb{R}^3}\Big(\sum_{k=1}^{N}\lambda_k^i\Psi_k^i(\vec{v})\Psi_k^i(\vec{v_1})\Big)f(\vec{v})\,d\vec{v}\,.$$

According to (15), $\hat{\mathcal{A}}(f,g)$ can be considered as a finitely dimensional approximation of the bilinear form $\mathcal{A}(f,g)$. Properties of $\hat{\mathcal{A}}(f,g)$ are fully determined by the eigenvalues λ_k^i and eigenvectors $\Psi_k^i(\vec{v})$.

Furthermore, $A(\vec{v},\vec{v}_1;\phi_i^c)$ can be approximated by a subset of eigenvalues and their corresponding eigenvectors as the following (see [30]):

$$A(\vec{v},\vec{v}_1;\phi_i^c) = \sum_{k=1}^{M}\lambda_k^i\Psi_k^i(\vec{v})\Psi_k^i(\vec{v_1}) + \tilde{\varepsilon}_{DG} + \mathrm{O}\Bigg(\sqrt{\frac{\sum_{k=M+1}^{N}|\lambda_k^i|^2}{\sum_{k=1}^{N}|\lambda_k^i|^2}}\Bigg)\,, \quad (16)$$

where M is the number of the eigenvalues used in the approximation. Let us introduce a truncated collision operator

$$\hat{\mathcal{A}}_M(f,g) := \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} f(t,\vec{x},\vec{v})g(t,\vec{x},\vec{v}_1)\Big(\sum_{k=1}^{M}\lambda_k^i\Psi_k^i(\vec{v})\Psi_k^i(\vec{v_1})\Big)d\vec{v}d\vec{v}_1\,. \quad (17)$$

The eigenvalues λ_k^i and the eigenvectors $\vec{\Psi}_k^i$ of $A_{\alpha\beta}^i$ can be computed using available linear algebra packages such as LAPACK and PROPACK (see [5] and [33]). Combining relations (13), (15), and (16), we obtain the approximation to $I_i^{i'i''}(\vec{\xi})$ as follows

$$\begin{aligned} I_i^{i'i''}(\vec{\xi}) &= \int_{\mathbb{R}^3}\int_{\mathbb{R}^3} f_{M_{i'}}(t,\vec{v}-\vec{\xi})\,f_{M_{i''}}(t,\vec{v_1}-\vec{\xi})\Big(\sum_{k=1}^{M}\lambda_k^i\Psi_k^i(\vec{v})\Psi_k^i(\vec{v_1})\Big)\,d\vec{v}d\vec{v_1}\\ &= \sum_{k=1}^{M}\lambda_k^i\int_{\mathbb{R}^3}\int_{\mathbb{R}^3} f_{M_{i'}}(t,\vec{v}-\vec{\xi})\,f_{M_{i''}}(t,\vec{v_1}-\vec{\xi})\,\Psi_k^i(\vec{v})\,\Psi_k^i(\vec{v_1})\,d\vec{v}d\vec{v_1}\\ &= \sum_{k=1}^{M}\lambda_k^i\Big(\int_{R^3} f_{M_{i'}}(t,\vec{v}-\vec{\xi})\Psi_k^i(\vec{v})\,d\vec{v}\Big)\Big(\int_{\mathbb{R}^3} f_{M_{i''}}(t,\vec{v_1}-\vec{\xi})\Psi_k^i(\vec{v_1})\,d\vec{v_1}\Big)\,. \end{aligned} \quad (18)$$

Notice that each integral in the last formula is a convolution of a single Maxwellian density with $\Psi_k^i(\vec{v})$. In the next section, we will develop a method for evaluating these convolutions efficiently.

3.3. **Ansatz for approximating convolutions of Maxwellian densities with eigenvectors.** In order to optimize evaluation of (18), we consider the following quantity

$$F_k^i(\vec{\xi},T)=\frac{1}{(\sqrt{\pi T})^3}\int_{\mathbb{R}^3}\exp\left(-\frac{|\vec{w}-\vec{\xi}|^2}{T}\right)\Psi_k^i(\vec{w})d\vec{w}\,. \tag{19}$$

Notice that the convolution integrals in (18) can be expressed using $F_k^i(\vec{\xi},T)$ as follows

$$\begin{aligned}\int_{\mathbb{R}^3} f_M(t,\vec{v}-\vec{\xi})\Psi_k^i(\vec{v})d\vec{v} &= \frac{n}{(\sqrt{\pi T})^3}\int_{\mathbb{R}^3}\exp\left(-\frac{|(\vec{v}-\vec{\xi})-\bar{\vec{u}}|^2}{T}\right)\Psi_k^i(\vec{v})d\vec{v}\\ &= \frac{n}{(\sqrt{\pi T})^3}\int_{\mathbb{R}^3}\exp\left(-\frac{|\vec{v}-(\vec{\xi}+\bar{\vec{u}})|^2}{T}\right)\Psi_k^i(\vec{v})d\vec{v}\\ &= n\,F_k^i(\vec{\xi}+\bar{\vec{u}},T)\,. \end{aligned}\tag{20}$$

Relation (20) indicates that in order to approximate the convolution integrals in (18), a set of values $F_k^i(\vec{\xi},T)$ may be computed and stored. For the evaluations of $F_k^i(\vec{\xi},T)$ at other points in the velocity space and temperature domain, we will employ the DG approximations (4) in the variable $\vec{\xi}$ and high order DG approximations in variable T as is described next.

Notice that in formulation (7), the distribution function is assumed to be well contained in the velocity domain. This means that the temperatures of the computed solutions have to belong to a certain range $[0,T^*]$ to guarantee accuracy. It was observed experimentally that the method provides a good accuracy if the temperature of the solution is such that the width of the distribution function is at most one-third of the width of the velocity domain. In the simulations presented in this paper, the velocity domain was chosen to be $[-3,3]\times[-3,3]\times[-3,3]$ and the temperature interval was taken to be $[0.01,1]$. To construct the nodal DG approximation in the variable T, we consider the temperature range on logarithmic scale, $\tau=\log_{10}(T)$, i.e., $\tau\in[-2,0]$. The interval $[-2,0]$ is then partitioned into uniform cells and high order nodal DG approximations are used on these cells. In this paper, we used two cells with five Gauss nodes on each cell to discretize $F_k^i(\vec{\xi},T)$ in the variable T.

To approximate $F_k^i(\vec{\xi},T)$, we first compute $F_k^i(\vec{v}_\alpha,T_q)$, at each node $\vec{v}_\alpha$ and each value $T_q=10^{\tau_q}$. The errors of computing $F_k^i(\vec{v}_\alpha,T_q)$ can be made small using adaptive quadrature methods. Values of $F_k^i(\vec{\xi},T)$ that do not fall on the nodes $(\vec{v}_\alpha,T_q)$ are computed by using nodal DG interpolation in the velocity and temperature variables. Notice that the values of $F_k^i(\vec{\xi},T)$ are assumed to be zero for all $\vec{\xi}$ that fall outside the velocity domain.

3.4. **Projection of the kinetic solution onto a sum of Maxwellian streams.** In this section, we will discuss the techniques that were used to compute approximation (11) for a given velocity distribution function $f(t,\vec{x},\vec{v})$.

For the sake of simplicity, we assume that the distribution function has density $n(t,\vec{x})=1$. In general, the velocity distribution function can be normalized before

each projection step to satisfy this requirement. The algorithm to evaluate the approximation (11) consists of the following four steps.

(i) Construction of the training set. In this first step, the training set of stochastic values $\vec{\omega}_k$, $k = 1, \dots, K$, that are distributed with the probability density $f(t, \vec{x}, \vec{v})$ is computed. To obtain the training set, we evaluate the reduced distributions $f_u(t, \vec{x}, \vec{v}) = \int_{\mathbb{R}^2} f(t, \vec{x}, \vec{v})\, dvdw$, $f_v(t, \vec{x}, \vec{v}) = \int_{\mathbb{R}^2} f(t, \vec{x}, \vec{v})\, dudw$, and $f_w(t, \vec{x}, \vec{v}) = \int_{\mathbb{R}^2} f(t, \vec{x}, \vec{v})\, dudv$. Next, we apply the inverse transformation techniques (see [16]) to the reduced distribution functions to construct three sets of one dimensional stochastic values ω_k^u, ω_k^v, and ω_k^w, $k = 1, \dots, K$, that are distributed with probability densities $f_u(t, \vec{x}, \vec{v})$, $f_v(t, \vec{x}, \vec{v})$, and $f_w(t, \vec{x}, \vec{v})$, correspondingly. The desired training set is constructed by randomly selecting K triplets $(\omega_l^u, \omega_m^v, \omega_n^w)$ from the sets $\{\omega_k^u\}, \{\omega_k^v\}$, and $\{\omega_k^w\}$, where $k, l, m, n \in \{1, \dots, K\}$.

To recover macroparameters $n_i(t, \vec{x})$, $T_i(t, \vec{x})$, and $\bar{\vec{u}}_i(t, \vec{x})$ that define the Maxwellian streams in (11), we use a version of the expectation maximization algorithm, described in [18, 11]. The recovery of $n_i(t, \vec{x})$, $T_i(t, \vec{x})$, and $\bar{\vec{u}}_i(t, \vec{x})$ is performed in steps (ii) - (iv).

(ii) Initialization step. The initial values for the variables $n_i(t, \vec{x})$ are assigned in such a way that $\sum_{i=1}^{p} n_i(t, \vec{x}) = 1$. Also, the initial values $\bar{\vec{u}}_i(t, \vec{x})$ are chosen randomly from the training set. The temperatures $T_i(t, \vec{x})$ are calculated as follows

$$T_i(t, \vec{x}) = \frac{2}{3K} \left(\sum_{k=1}^{K} \left(\vec{\omega}_k - \bar{\vec{u}}_i(t, \vec{x}) \right)^T \left(\vec{\omega}_k - \bar{\vec{u}}_i(t, \vec{x}) \right) \right).$$

(iii) Expectation step. The following auxiliary quantities are computed

$$\gamma_{i,k} = \frac{f_{M_i}(t, \vec{x}, \vec{\omega}_k)}{\sum_{j=1}^{p} f_{M_j}(t, \vec{x}, \vec{\omega}_k)}.$$

(iv) Maximization step. Values of the variables $n_i(t, \vec{x})$, $T_i(t, \vec{x})$ and $\bar{\vec{u}}_i(t, \vec{x})$ are updated using the following formulas

$$\bar{\vec{u}}_i(t, \vec{x}) = \left(\sum_{k=1}^{K} \gamma_{i,k}\, \vec{\omega}_k \right) \Big/ \left(\sum_{k=1}^{K} \gamma_{i,k} \right), \qquad n_i(t, \vec{x}) = \frac{1}{K} \sum_{k=1}^{K} \gamma_{i,k},$$

$$T_i(t, \vec{x}) = \frac{2}{3} \sum_{k=1}^{K} \left(\gamma_{i,k} \left(\vec{\omega}_k - \bar{\vec{u}}_i(t, \vec{x}) \right)^T \left(\vec{\omega}_k - \bar{\vec{u}}_i(t, \vec{x}) \right) \right) \Big/ \sum_{k=1}^{K} \gamma_{i,k}.$$

The expectation step and the maximization step are repeated until the macroparameters $n_i(t, \vec{x})$, $T_i(t, \vec{x})$, and $\bar{\vec{u}}_i(t, \vec{x})$ converge to a desired accuracy. It is expected that the algorithm will not reach the global minimum during the maximization steps. Therefore, the entire algorithm is repeated several times and the L_1 norm of the difference between the representation (11) and the solution $f(t, \vec{x}, \vec{v})$ is computed for each run of the algorithm, namely

$$\varepsilon_{L_1} = \frac{1}{n(t, \vec{x})} \int_{\mathbb{R}^3} \left| f(t, \vec{x}, \vec{v}) - \sum_{i=1}^{p} f_{M_i}(t, \vec{x}, \vec{v}) \right| d\vec{v}. \tag{21}$$

The values of $n_i(t, \vec{x})$, $T_i(t, \vec{x})$, and $\bar{\vec{u}}_i(t, \vec{x})$ that correspond to the smallest ε_{L_1} are accepted.

3.5. **Computing solutions to the Boltzmann equation in the spatially homogeneous case.** Assume that discrete velocity solution $f_{i;j}(t)$ is known at some time t. The solution at time $t + \Delta t$ is obtained using the following steps.

(i) The solution $f(t, \vec{v})$ is reconstructed from $f_{i;j}(t)$ using (4) and approximated by a sum of Maxwellian streams (11) using the algorithm described in Section 3.4.

(ii) Convolutions in (18) are obtained from $F_k^i(\vec{\xi}, T)$ using (20). An approximation of the Boltzmann collision integral is evaluated using relations (18) and (12).

(iii) The solution is computed at the next time step using Euler's method:

$$f_{i;j}(t + \Delta t) = f_{i;j}(t) + \Delta t\, I_i(\vec{\xi^j})\,.$$

We notice that in order to evaluate the collision integral, we have to perform interpolation of $F_k^i(\vec{\xi}, T)$ for each value of i and for each cellular shift $\vec{\xi^j}$. This results in the total of N interpolations, where N is the total number of points on the velocity grid. Each interpolation will take the same number of operations which is proportional to the number $s_u s_v s_w$ of Gauss nodes on a velocity cell where s_u, s_v, and s_w are the numbers of nodes in each cell in each velocity dimension. Therefore, we estimate the number of operations for the interpolations to be $O(N)$. Interpolations are repeated for each of the M eigenvectors used in (18). Therefore, our new approach allows to compute the Boltzmann collision integral in $O(MN)$ operations. We notice that this estimate does not include the cost of memory transfer operations which may be substantial.

3.6. **Enforcing conservation of mass, momentum, and energy.** The stochastic algorithm described in Section 3.4 returns approximations that do not have the same density, momentum, and energy as the full discrete solution. The differences in density and momentum are minute and differences in the energy are small. Nevertheless these small errors can lead to a significant error in the solution after a large number of time steps. To avoid this error, the obtained approximations are post-processed by dividing the discrepancies in density, bulk velocity, and temperature evenly among the approximating Maxwellian streams so as to enforce conservation.

An additional violation of conservation laws occurs during the evaluation of the collision operator using formula (18). To enforce conservation laws at this step we use the approach of [13, 25, 26] in which a linear least squares problem subject to constrains of zero mass, momentum, and energy in the discrete collision operator is solved at every evaluation step. The result of this procedure is a conservative discrete collision operator that is closest, in the l^2 norm, to the one computed from formula (18). It has been noticed, however, that if the algorithm is applied to update values of the collision operator at all velocity nodes, the least squares procedure assigns corrections to the values to the collision operator at all velocity nodes, including the nodes where the values of the collision operator are expected to be zero. While the absolute values of the corrections may be small, their relative errors are infinite. When the solution is updated to include these corrections, small negative values appear in the solution where the solution is expected to be zero. These negative values rapidly lead to a failure of the stochastic approximation algorithm. To circumvent this issue, the collision operator was only corrected at points where the solution is significantly above zero. This allows for enforcing the conservation laws in the collision operator and does not produce spurious negative values in the solution.

An approach to enforce conservation laws in Fourier spectral discretizations of the collision operator was proposed in [21] using a decomposition of the solution into a sum of the local Maxwellian density and the deviation from the local Maxwellian. By substituting this decomposition into the collision operator and using the fact that Maxwellians nullify the collision operator, the authors were able to re-write the method to achieve better conservation properties. The same decomposition is also used in the DG velocity discretizations of the collision operator in [2, 3] to reduce roundoff cancellation errors in solutions that are near Maxwellian. It was observed too, that conservation properties of the decomposed solutions are better, especially when solutions are near their steady states. The decomposition, however, is not directly applicable to the scheme presented in this paper. We note, however, that similar to the decomposition, only cross-interactions of the streams are computed in the evaluation of the collision operator for a sum of Maxwellian densities.

4. **Numerical simulations.** An important parameter in the presented method is M, the number of eigenvectors $\Psi_k(\vec{v})$ used to approximate $A(\vec{v}_1, \vec{v}; \phi_i^c)$ in (16). This number has to be large enough so that the error term in (16) is small. On the other hand, having a large number of eigenvectors would result in more expensive calculations. Therefore, it is important to understand how to choose M to achieve the desired accuracy while maintaining efficiency. The number of eigenvectors can be selected by observing how fast the magnitudes of eigenvalues of $A^i_{\alpha\beta}$ decrease. We evaluated eigenvalues of the matrix $A^i_{\alpha\beta}$ for the two cases of nodal DG discretizations: using uniform grids with 15 and 33 cells in each velocity dimension. In each case, one Gauss node per cell was used, i.e., $s_u = s_v = s_w = 1$. Thus, these cases correspond to $N = 15^3$ and $N = 33^3$ velocity points, respectively. We note that in the case of a single Gauss node per cell, the index i in (4) takes only on a single value, $i = 1$. Therefore we will omit index i in the discussion of numerical results. The magnitudes of the eigenvalues for the cases $N = 15^3$ and $N = 33^3$ are shown in Figures 1a and 1b, respectively, using base ten logarithmic scale. Note that the total number of eigenvalues is N in both cases. Figure 1a shows approximately fifty percent of the eigenvalues and Figure 1b shows approximately forty percent.

According to [30], the Frobenius norm of the relative error in the approximation of $A_{\alpha\beta}$ is given by the quantity $\varepsilon_{rel} = ((\sum_{k=M+1}^{N} |\lambda_k|^2)/(\sum_{k=1}^{N} |\lambda_k|^2))^{1/2}$. In particular, if one percent accuracy is desired, this quantity should be less than or equal to 0.01. It was found experimentally that, in the cases of $N = 15^3$ and $N = 33^3$,

$$\sqrt{\frac{\sum_{k=800}^{N=15^3} |\lambda_k|^2}{\sum_{k=1}^{N=15^3} |\lambda_k|^2}} \approx 0.0099 \quad \text{and} \quad \sqrt{\frac{\sum_{k=3000}^{N=33^3} |\lambda_k|^2}{\sum_{k=1}^{N=33^3} |\lambda_k|^2}} \approx 0.0099\,,$$

respectively. These results imply that in order to keep the relative errors in the approximation of the matrix $A_{\alpha\beta}$ within 10^{-2}, it is sufficient to choose $M \approx 800$ and $M \approx 3000$. Similarly, to keep the relative errors within 10^{-3}, it is suggested to select $M \approx 1165$ and $M \approx 11300$ for the cases $N = 15^3$ and $N = 33^3$, respectively. To keep the relative errors within 10^{-4}, it will require using $M \approx 1270$ and $M \approx 13000$. Note that errors in the representation (16) are compounded from the errors in the approximation of $A_{\alpha\beta}$ and the errors of the DG approximations. Thus, the accuracy in the approximation of $A(\vec{v}, \vec{v}_1, \phi^c)$ is not guaranteed by the value of ε_{rel} alone. However, our calculations suggest that ε_{rel} may be used as a satisfactory indicator of the relative error.

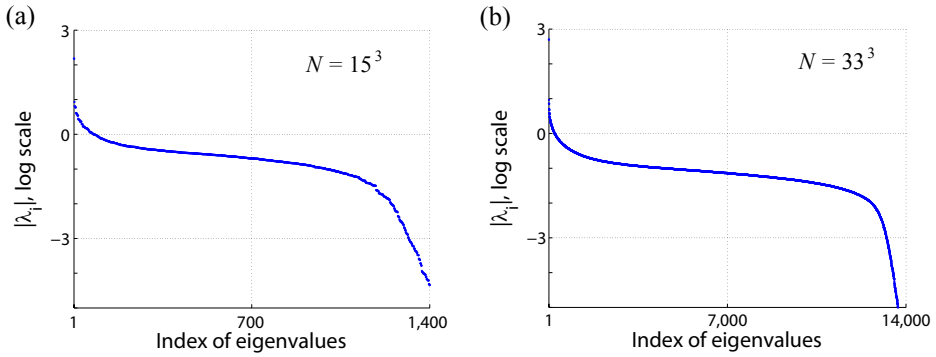

FIGURE 1. Magnitudes of eigenvalues of $A_{\alpha\beta}$ for the cases $s_u = s_v = s_w = 1$, $N = 15^3$ (a) and $s = 1$, $N = 33^3$ (b).

4.1. **Convergence of the truncated eigenvector expansion (17).** In this section, we perform simulations to verify the choice of parameter M describing the number of eigenvectors retained in expansion (17). The simulations use uniform grids with 15 and 33 cells in each velocity dimension. We will denote these cases as $N = 15^3$ and $N = 33^3$, correspondingly. One node per cell is used in both cases, i.e., $s_u = s_v = s_w = 1$. To estimate errors that are due to truncation of $\hat{\mathcal{A}}(f, f)$, we compute numerically the left side of (15) for given velocity distribution functions. The numerical integration is performed using adaptive quadrature methods and the integration errors are kept small. Also, $\hat{\mathcal{A}}_M(f, f)$ is evaluated for the same distribution functions using adaptive numerical integration. We then compare the two results. Since adaptive integration is used to compute the two quantities, the bulk of the error is expected to be due to the truncation of the eigenvalue expansion defining $\hat{\mathcal{A}}_M(f, f)$. The results are summarized in Tables 1 and 2, for the $N = 15^3$ and $N = 33^3$ cases, respectively.

Results of the comparison for the case of $N = 15^3$ grid are given in Table 1. In the first column, the numbers M of eigenvectors used in (17) are shown. In the second column, the absolute values of the last retained eigenvalues $|\lambda_M|$ are shown. Note that eigenvectors that are truncated correspond to eigenvalues smaller than $|\lambda_M|$ by their absolute values. In the third column, differences between $\mathcal{A}(f_1, f_1)$ and $\hat{\mathcal{A}}_M(f_1, f_1)$ are shown for the case of Maxwellian distribution $f_1(t, \vec{x}, \vec{v})$ with the density $n(t, \vec{x}) = 1$, bulk velocity $\bar{\vec{u}}(t, \vec{x}) = (0, 0, 0)$, and temperature $T(t, \vec{x}) = 0.59$. Specifically, ε_1 is defined as follows:

$$\varepsilon_1 = ||\mathcal{A}(f, f) - \hat{\mathcal{A}}_M(f, f)||_\infty . \tag{22}$$

We recall that Maxwellian distributions are stationary solutions to the Boltzmann equation. Thus, the value of $\mathcal{A}(f, f)$ should be zero in this case. However, since (15) contains errors of the DG approximations of $A(\vec{v}, \vec{v}_1; \phi^c)$, the computed quantity will not be zero. It is, however, expected that as M approaches N, values of $\hat{\mathcal{A}}_M(f, f)$ will approach those of $\mathcal{A}(f, f)$. Differences between $\mathcal{A}(f_i, f_i)$ and $\hat{\mathcal{A}}_M(f_i, f_i)$, $i = 2, 3$ are shown in the fourth and fifth columns. Here $f_{2,3}(t, \vec{x}, \vec{v})$ are sums of two Maxwellian streams, that is, $f_{2,3}(t, \vec{x}, \vec{v}) = f_{M_1}(t, \vec{x}, \vec{v}) + f_{M_2}(t, \vec{x}, \vec{v})$. In the case of $f_2(t, \vec{x}, \vec{v})$, Maxwellian streams with densities $n_1(t, \vec{x}) = n_2(t, \vec{x}) = 1$, bulk velocities $\bar{\vec{u}}_1(t, \vec{x}) = (0.4, 0, 0)$ and $\bar{\vec{u}}_2(t, \vec{x}) = (-0.4, 0, 0)$, and temperatures

M	$\lvert\lambda_M\rvert$	ε_1	ε_2	ε_3
10	4.1E+0	1.7E-1	3.1E-1	2.1E-1
80	1.0E+0	3.7E-2	5.8E-1	5.3E-2
160	5.7E-1	1.6E-2	2.9E-2	2.0E-2
320	3.6E-1	1.5E-2	2.5E-2	1.8E-2
640	2.2E-1	3.3E-3	8.4E-3	4.4E-3
960	1.1E-1	1.3E-3	5.7E-3	2.2E-3
1120	5.8E-2	2.3E-3	5.9E-3	2.3E-3
1280	4.1E-3	1.7E-5	2.1E-3	3.1E-5
1400	4.6E-5	2.2E-7	2.0E-3	3.9E-7

TABLE 1. Convergence of errors in the approximation of $\mathcal{A}(f,f)$ by $\hat{\mathcal{A}}_M(f,f)$ for the case $N = 15^3$ with respect to the number M of retained eigenvectors: $|\lambda_M|$ is the magnitude of the M^{th} eigenvalue; ε_1 is the absolute error between $\mathcal{A}(f,f)$ and $\hat{\mathcal{A}}_M(f,f)$ when $f(t,\vec{x},\vec{v})$ is a single Maxwellian stream with density $n = 1$, bulk velocity $\bar{\vec{u}} = (0,0,0)$ and temperature $T = 0.59$; ε_2 is the relative error between $\mathcal{A}(f,f)$ and $\hat{\mathcal{A}}_M(f,f)$ when $f(t,\vec{x},\vec{v})$ is a sum of two Maxwellian streams with densities $n_1 = n_2 = 1$, bulk velocities $\bar{\vec{u}}_1 = (0.4,0,0)$ and $\bar{\vec{u}}_2 = (-0.4,0,0)$ and temperatures $T_1 = T_2 = 0.89$; ε_3 is the relative error between $\mathcal{A}(f,f)$ and $\hat{\mathcal{A}}_M(f,f)$ when $f(t,\vec{x},\vec{v})$ is a sum of two Maxwellian streams with densities $n_1 = n_2 = 1$, bulk velocities $\bar{\vec{u}}_1 = (0.4,0,0)$ and $\bar{\vec{u}}_2 = (-0.4,0,0)$ and temperatures $T_1 = T_2 = 0.31$.

$T_1(t,\vec{x}) = T_2(t,\vec{x}) = 0.89$ are used. In the case of $f_3(t,\vec{x},\vec{v})$, streams with densities $n_1(t,\vec{x}) = n_2(t,\vec{x}) = 1$, bulk velocities $\bar{\vec{u}}_1(t,\vec{x}) = (0.4,0,0)$ and $\bar{\vec{u}}_2(t,\vec{x}) = (-0.4,0,0)$, and temperatures $T_1(t,\vec{x}) = T_2(t,\vec{x}) = 0.31$ are used. The relative errors ε_2 and ε_3 are defined by

$$\varepsilon_{2,3} = \frac{||\mathcal{A}(f_{M_1}+f_{M_2}, f_{M_1}+f_{M_2}) - \hat{\mathcal{A}}_M(f_{M_1}+f_{M_2}, f_{M_1}+f_{M_2})||_\infty}{||\mathcal{A}(f_{M_1}+f_{M_2}, f_{M_1}+f_{M_2})||_\infty}\,. \tag{23}$$

As is seen in Table 1, the values of the absolute errors ε_1 drop as the values of M increase. However, there are moments when the values of M increase but the values of ε_1 increase as well. This may be caused by the existence of groups of eigenvectors of comparable influence. In this case, the truncation may be capturing only a portion of the group and therefore is not achieving an improvement in accuracy. Similar observations can be made about relative errors ε_2 and ε_3. We notice that in both cases, use of $M \geq 500$ is required to keep errors less than one percent.

In Table 2, results of the comparison for the case of $N = 33^3$ grid are presented. In the third column, the absolute errors ε_1 are shown for the case of a Maxwellian distribution with the density $n(t,\vec{x}) = 1$, bulk velocity $\bar{\vec{u}}(t,\vec{x}) = (0,0,0)$ and temperature $T(t,\vec{x}) = 0.89$. In the fourth column, the errors are shown for the sum of Maxwellian distributions with the densities $n_1(t,\vec{x}) = n_2(t,\vec{x}) = 1$, bulk velocities $\bar{\vec{u}}_1(t,\vec{x}) = (0.\overline{36},0,0)$ and $\bar{\vec{u}}_2(t,\vec{x}) = (-0.\overline{36},0,0)$, and temperatures $T_1(t,\vec{x}) = T_2(t,\vec{x}) = 0.89$. In the last column, errors are given for the case of the sum of two Maxwellians with the densities $n_1(t,\vec{x}) = n_2(t,\vec{x}) = 1$, bulk velocities $\bar{\vec{u}}_1(t,\vec{x}) = (0.\overline{36},0,0)$ and $\bar{\vec{u}}_2(t,\vec{x}) = (-0.\overline{36},0,0)$, and temperatures $T_1(t,\vec{x}) = T_2(t,\vec{x}) = 0.31$.

M	$\lvert\lambda_M\rvert$	ε_1	ε_2	ε_3
10	5.0E+0	1.0E-1	5.2E-1	3.7E-1
100	1.7E+0	4.1E-2	2.0E-1	3.5E-1
500	5.4E-1	5.0E-3	2.3E-2	2.5E-2
1000	2.9E-1	3.3E-3	1.6E-2	1.5E-2
2000	1.7E-1	1.6E-3	7.9E-3	4.7E-3
2900	1.3E-1	1.5E-3	7.5E-3	3.9E-3
4800	9.6E-2	9.2E-4	4.6E-3	2.3E-3
9900	4.3E-2	1.6E-4	3.5E-3	5.3E-4
14400	2.8E-12	2.8E-5	3.5E-3	9.2E-7

TABLE 2. Convergence of errors in the approximation of $\mathcal{A}(f,f)$ by $\hat{\mathcal{A}}_M(f,f)$ for the case $N = 33^3$ with respect to the number M of retained eigenvectors: $|\lambda_M|$ is the magnitude of the M^{th} eigenvalue; ε_1 is the absolute error between $\mathcal{A}(f,f)$ and $\hat{\mathcal{A}}_M(f,f)$ when $f(t,\vec{x},\vec{v})$ is a single Maxwellian stream with density $n = 1$, bulk velocity $\bar{\vec{u}} = (0,0,0)$ and temperature $T = 0.89$. ε_2 is the relative error between values of $\mathcal{A}(f,f)$ and values $\hat{\mathcal{A}}_M(f,f)$ when $f(t,\vec{x},\vec{v})$ is a sum of two Maxwellian streams with densities $n_1 = n_2 = 1$, bulk velocities $\bar{\vec{u}}_1 = (0.\overline{36},0,0)$ and $\bar{\vec{u}}_2 = (-0.\overline{36},0,0)$ and temperatures $T_1 = T_2 = 0.89$. The quantity ε_3 is the relative error between values of $\mathcal{A}(f,f)$ and values $\hat{\mathcal{A}}_M(f,f)$ when $f(t,\vec{x},\vec{v})$ is a sum of two Maxwellian streams with densities $n_1 = n_2 = 1$, bulk velocities $\bar{\vec{u}}_1 = (0.\overline{36},0,0)$ and $\bar{\vec{u}}_2 = (-0.\overline{36},0,0)$ and temperatures $T_1 = T_2 = 0.31$.

As is seen from the results presented in Table 2, the values of the errors get smaller as M increases and the magnitudes of eigenvalues of the truncated eigenvectors decrease. It is observed that using $M = 1000$ eigenvectors results in errors of approximately 1.6 percent. Also, using $M \geq 2000$ is necessary to have relative errors less than one percent.

4.2. **Computational efficiency of the new method.** In this section we summarize the computational time needed in the new method. Simulations were performed for the cases of $N = 15^3$ and $N = 33^3$ grids on a 2.4 GHz processor. In Table 3, CPU times to complete one time step of the solution are listed as functions of M, the number of eigenvectors retained in expansion (17). The bottom row of the table lists the CPU times used in the evaluation of the full collision integral in the approach of [3]. The speedup coefficient is the ratio of the time used in the new method and the time used in the approach of [3].

It can be seen from Table 3 that computational times grow linearly with the number M of eigenvectors used in (17). We note that the step in which the solution is approximated by a sum of Maxwell distributions is included in the time measurements. Since the approximation step is the same for all values of M and since it does not seem to impact the linear growth of computational time with respect to M, we conclude that the time necessary to compute the approximation is small as compared to the time necessary to compute $\hat{A}_M$.

According to Tables 1 and 2, values $M = 1400$ and $M = 14000$ are sufficient to achieve less than a percent error in the evaluation of collision integral using the new method for the girds $N = 15^3$ and $N = 33^3$, respectively. Therefore, the

$n = 15$, $s_u = s_v = s_w = 1$			$n = 33$, $s_u = s_v = s_w = 1$		
M	CPU Time, s	Speedup	M	CPU Time, s	Speedup
160	2.0	16.4	4000	444.1	40.0
320	3.8	8.7	6000	668.10	26.6
480	5.4	6.1	8000	890.2	19.9
640	7.0	4.8	10000	1113.4	15.9
960	10.3	3.2	12000	1333.6	13.3
1400	16.0	2.1	14000	1559.4	11.4
DG Boltzmann CPU Time: 33.18 s			DG Boltzmann CPU Time: 17758 s		

TABLE 3. CPU time in seconds for computing one time step of the spatially homogeneous solution on a 2.4 GHz processor. Results of simulations for the $N = 15^3$ and $N = 33^3$ grids are presented. M is the number of eigenvectors used in (17). The bottom row gives the time for computing one time step in the method of [3]. Speedup is the ratio of the time used in the new method and the time used by the algorithm of [3].

new method is about twice as fast as the full Boltzmann solver in the case of the $N = 15^3$ grid and more than eleven times as fast in the case of the $N = 33^3$ grid. Moreover, since the full evaluation of the collision operator requires $O(N^{8/3})$ operations and the evaluation of the $\hat{A}_M$ requires $O(MN)$ operations, even larger speedup is anticipated for larger values of N.

4.3. **Approximations by Maxwellian densities in solutions to spatially homogeneous relaxation.** In this section, we will verify that a solution to the problem of spatially homogeneous relaxation can be approximated by a sum of a small number of Maxwellian distributions. We consider a solution to the full Boltzmann equation computed by the method of [3] and apply the approximation algorithm of Section 3.4 to obtain an approximation to the solution by a sum of Maxwellian distributions. Our benchmark solution uses 33 velocity cells in each velocity dimension and one node per cell, i.e., $s_u = s_v = s_w = 1$. Thus, the solution has the total of $N = 33^3$ velocity points. The maximum likelihood algorithm of Section 3.4 is performed using $K = 75000$ sample points. The number of sample points is chosen higher than the number of points in the solution so as to guarantee abundant data for the approximation. In particular, using 75000 sample points results in consistent small values of the L^1 norm of the approximation error. The algorithm performs maximization of the likelihood function several times and the macroparameters of the approximation with the smallest L^1 norm of the error are accepted.

The results of the density estimation simulations are given in Tables 4 and 5. The initial data for the spatially homogeneous solution is a sum of two Maxwellian streams. The dimensionless densities, bulk velocities and temperatures of the two initial streams are $n_1 = 1.609$, $\vec{u}_1 = (0.775, 0, 0)$, $T_1 = 0.3$ and $n_2 = 2.863$, $\vec{u}_2 = (0.436, 0, 0)$, $T_2 = 0.464$. The reference density for this simulation is $N_{ref} =$ 10E+21, and the reference temperature is $T_{ref} = 1000$ K. The characteristic length scale is $L_{ref} = 1$ m, the value of the gas constant is $R = 208.13$, and the molecular mass is 6.63E-26 kg. The two Maxwellian streams correspond to downstream and upstream conditions of a stationary normal shock wave with Mach number 1.55.

Results presented in Table 4 correspond to approximations using two Maxwellian streams, i.e., the case $p = 2$ in (11). The first column gives the values of the time of

the computed solution normalized to the mean free time. Columns two to six give the computed values of the dimensionless macroparameters of the first approximating Maxwellian stream. In particular, the second column gives the dimensionless density n_1, the third column gives the dimensionless temperature T_1 and the fourth, fifth and sixth columns give the u, v, and w-components of the dimensionless bulk velocity $\vec{\vec{u}}_1$, respectively. Similarly columns seven to eleven give the macroparameters of the second approximating Maxwellian. The L^1 norm of the approximation error, ε_{L^1}, is shown in the last column of the table.

It can be seen in Table 4 that macroparameters of the two Maxwellian streams making up the initial data were identified correctly by the algorithm. In general, errors of up to five percent were observed in the reconstructed macroparameters of a mixture of Maxwellians. The L_1 error of the reconstructed solution, however, is expected to be small. In particular, in all instances of approximations shown in Table 4, the L^1 norm of the difference between the solution and its approximation is less than a percent. Approximations were computed for the solution at different stages in the relaxation process. In can be seen that for about three mean free times, two Maxwellian streams with different temperatures were identified. To make the results of the table easier to read, one of the identified streams is highlighted in grey. As time increases, the temperatures of the two streams are getting closer in magnitude. After six mean free times, the solution almost reached the steady state and the streams become difficult to distinguish. In this case, the algorithm returns two Maxwellian streams with close values of bulk velocity and temperature. In other words, the algorithm identified two instances of the same Maxwellian with different densities.

Next, the approximation procedure is applied again, this time using $p = 3$ Maxwellian streams. The results are summarized in Table 5. Similar to Table 4, the columns of Table 5 contain macroparameters of the approximating Maxwellian streams. Columns containing the w-component of the bulk velocities $\vec{\vec{u}}_i$, $i = 1, 2, 3$ were removed for brevity. The values of the w-components, however, are very close to those of the v-components.

The behavior of approximating Maxwellian streams shown in Table 5 is very similar to the behavior seen in Table 4. However, at the initial time and at several moments later, two of the three approximating streams appear to be pertubations of a single Maxwellian stream. In particular, at the initial moment, the first and the second Maxwellian streams are perturbations of a single Maxwellian stream with the combined density of the two streams. One can see that the identified streams contain $3\% - 5\%$ errors in the temperature as compared to the Maxwellian stream with the temperature $T_2 = 0.464$ used in the initial data. Similarly, after a few moments we can still see two streams that may be perturbed copies of the same stream. Overall, it appears that throughout the entire relaxation process, two Maxwellian streams can be observed, with one being duplicated by the approximation algorithm. Similar observations were made when the solution was approximated using four Maxwellian streams. Therefore, we conclude that using two Maxwellian streams in the approximation step is sufficient in this problem.

4.4. **Solution of the problem of spatially homogeneous relaxation.** The algorithm described in Section 3.5 for evaluating the Boltzmann collision integral using truncated discrete kernels was applied to the solution of the problem of spatially homogeneous relaxation of a sum of two Maxwellian streams. Two cases of initial data were considered. In the first case, the two streams have dimensionless

t/τ	n_1	T_1	$\bar{u}_1$	$\bar{v}_1$	$\bar{w}_1$	n_2	T_2	$\bar{u}_2$	$\bar{v}_2$	$\bar{w}_2$	ε_{L_1}
0.0	1.54	0.30	0.78	1.1E-2	3.6E-4	2.94	0.46	0.44	-7.4E-3	-2.9E-3	8.5E-3
0.04	1.66	0.30	0.77	1.9E-3	-7.4E-3	2.81	0.47	0.43	-2.5E-4	9.7E-4	8.3E-3
0.45	3.31	0.46	0.48	5.8E-4	1.2E-3	1.16	0.29	0.78	6.3E-3	4.7E-4	7.8E-3
0.87	1.19	0.30	0.76	1.9E-3	1.0E-2	3.28	0.45	0.48	3.3E-4	-2.0E-3	5.8E-3
1.7	1.30	0.33	0.69	6.6E-4	1.5E-2	3.17	0.46	0.50	6.6E-4	-8.9E-3	7.8E-3
2.9	2.93	0.45	0.51	-3.7E-3	7.8E-3	1.54	0.37	0.65	4.5E-3	-9.3E-3	7.1E-3
3.8	1.29	0.36	0.65	-1.6E-2	-1.2E-2	3.18	0.44	0.52	9.0E-3	5.2E-3	6.9E-3
4.2	3.53	0.44	0.53	-4.0E-3	-3.2E-3	0.94	0.36	0.66	1.2E-2	1.3E-2	5.9E-3
4.5	2.21	0.44	0.52	-1.2E-2	-2.7E-2	2.26	0.40	0.60	1.3E-2	2.8E-2	7.0E-3
6.2	2.34	0.42	0.53	2.6E-2	-8.2E-3	2.13	0.42	0.59	-2.7E-2	9.9E-3	6.1E-3
7.9	1.58	0.42	0.53	4.4E-2	-4.3E-2	2.89	0.42	0.57	-2.4E-2	2.3E-2	4.6E-3

TABLE 4. Results of approximating the solution to the problem of spatially homogeneous relaxation [3] by a sum of two Maxwellian distributions at different stages of the relaxation process. The values of the time are normalized to the mean free time. The initial data for the problem is a sum of two Maxwellians with dimensionless densities, bulk velocities and temperatures, $n_1 = 1.609$, $\bar{\vec{u}}_1 = (0.775, 0, 0)$, $T_1 = 0.3$ and $n_2 = 2.863$, $\bar{\vec{u}}_2 = (0.436, 0, 0)$, T_2=0.464, respectively. Columns n_i, T_i, $\bar{u}_i$, $\bar{v}_i$, and $\bar{w}_i$, $i = 1, 2$ show the density, temperature, and the components of the bulk velocity of the approximating Maxwellians, respectively; ε_{L^1} is the L^1 norm error given by (21).

t/τ	n_1	T_1	$\bar{u}_1$	$\bar{v}_1$	n_2	T_2	$\bar{u}_2$	$\bar{v}_2$	n_3	T_3	$\bar{u}_3$	$\bar{v}_3$	ε_{L_1}
0.0	1.35	0.48	0.38	-2.8E-2	1.51	0.44	0.48	3.0E-3	1.61	0.30	0.78	-5.9E-3	6.6E-3
.04	1.97	0.43	0.51	1.0E-2	1.17	0.48	0.38	-1.4E-2	1.33	0.29	0.79	-9.6E-3	6.2E-3
.45	1.86	0.33	0.73	3.6E-3	0.68	0.47	0.45	-4.9E-2	1.93	0.47	0.43	1.3E-2	7.9E-3
.87	0.89	0.46	0.40	-9.0E-2	2.47	0.44	0.53	3.9E-2	1.11	0.31	0.75	-1.1E-2	8.9E-3
1.7	0.92	0.29	0.73	5.5E-3	0.68	0.44	0.52	1.0E-1	2.87	0.45	0.51	-2.7E-2	7.0E-3
2.9	1.59	0.42	0.58	1.4E-2	0.92	0.34	0.67	-1.9E-2	1.96	0.45	0.49	5.7E-2	7.0E-3
3.8	2.01	0.39	0.62	-3.0E-3	1.03	0.43	0.52	5.2E-2	1.43	0.45	0.50	-3.0E-2	6.2E-3
4.2	1.64	0.43	0.54	-2.5E-2	1.03	0.38	0.64	-2.7E-2	1.80	0.43	0.52	-6.7E-3	7.6E-3
4.5	1.50	0.45	0.49	-4.1E-2	0.84	0.37	0.65	-4.0E-2	2.13	0.42	0.57	4.2E-2	5.4E-3
6.2	1.90	0.41	0.56	-2.3E-2	1.81	0.44	0.53	4.3E-2	0.76	0.39	0.63	-3.0E-2	8.0E-3
7.9	1.34	0.42	0.60	2.9E-4	2.27	0.42	0.57	-2.3E-2	0.84	0.41	0.47	6.1E-2	5.9E-3

TABLE 5. Results of approximating the solution to the problem of spatially homogeneous relaxation by a sum of three Maxwellian distributions at different stages of the relaxation process. The initial data for the problem is a sum of two Maxwellians with dimensionless densities, bulk velocities and temperatures, $n_1 = 1.609$, $\bar{\vec{u}}_1 = (0.775, 0, 0)$, $T_1 = 0.3$ and $n_2 = 2.863$, $\bar{\vec{u}}_2 = (0.436, 0, 0)$, T_2=0.464, respectively.

densities, bulk velocities and temperatures $n_1 = 1.609$, $\vec{u}_1 = (0.775, 0, 0)$, $T_1 = 0.3$ and $n_2 = 2.863$, $\vec{u}_2 = (0.436, 0, 0)$, $T_2 = 0.464$. Solutions with this initial data are plotted in Figures 2(a) and (b). The reference number density for this simulation is $N_{\mathbf{ref}} = 10\text{E}{+}21$ and the reference temperature is $T_{\mathbf{ref}} = 1000$ K. In all simulations, the characteristic length scale is $L_{\mathbf{ref}} = 1$ m, the value of the gas constant is $R = 208.13$, and the molecular mass is $6.63\text{E}{-}26$ kg. The value of the exponent in the power viscosity law is $\gamma = 0.5$, which corresponds to hard spheres gas. We notice that the selected Maxwellian streams are downstream and upstream conditions for a stationary normal shock wave with Mach number 1.55. In the second case, the two streams have dimensionless densities, bulk velocities and temperatures $n_1 = 1$, $\vec{u}_1 = (1.225, 0, 0)$, $T_1 = 0.2$ and $n_2 = 3$, $\vec{u}_2 = (0.408, 0, 0)$, $T_2 = 0.733$. These solutions are plotted in Figures 2(c) and (d). The reference number density is $N_{\mathbf{ref}} = 10\text{E}{+}20$ and the reference temperature is $T_{\mathbf{ref}} = 1500$ K. These values correspond to downstream and upstream conditions of a normal shock wave with Mach number 3.

In Figure 2 relaxations of the directional temperatures are presented for the two cases of initial data. It can be seen in Figures 2(a) and (c) that the algorithm using truncated kernels is reasonably accurate in capturing the dynamics of the solution for the first three mean free times. Some oscillations can be seen in the solutions, especially noticeable in the case of Mach 3.0 initial data. The authors believe that these oscillations are primarily caused by the errors of stochastic approximation of the solution by a sum of Maxwellian streams. Therefore, a more efficient approximation algorithm is needed to reduce the oscillations. Additional improvement in the solution can be achieved by increasing the numbers of velocity and temperature points in the pre-computed kernel (19).

Shortly after the time mark of three mean free times, the algorithm using truncated kernels fails. It was observed that at this point the approximation algorithms start to return two Maxwellian streams in which one has macroparameters close to that of the steady state solution and another Maxwellian having small values of density and large values of temperature. At the same time, values presented in Tables 4 and 5 do not exhibit presence of a high temperature Maxwellian component suggesting that this artifact might be due to an instability intrinsic to the new method. On the other hand, results presented in Tables 4 and 5 suggest a high level of uncertainty in the values of macroparameters of the approximating Maxwellian streams for the period from three to six mean free times. In this period, distinctions between the two populations of molecules begin to blur. It may be the case that a small number of Maxwellian streams are not sufficient to capture accurately small perturbations of the steady state solution or that maximization of the likelihood function leads to a sensitive reconstruction of the parameters of weak contributions.

A possible way to circumvent the problem is to use model collision operators once the stochastic approximation algorithm fails. In Figures 2(b) and (d), solutions are presented for the two cases of initial data in which the new model is used to compute the solution for the first three mean free times. After that the solution is computed using the classical ellipsoidal-statistical Bhatnagar-Gross-Crook method (ES-BGK)[28]. The solutions obtained by this hybrid approach approximate the solution to the full Boltzmann equation reasonably well. In particular, the solution presented in Figures 2(d) shows an improvement as compared to the solution obtained by the ES-BGK model presented in [1].

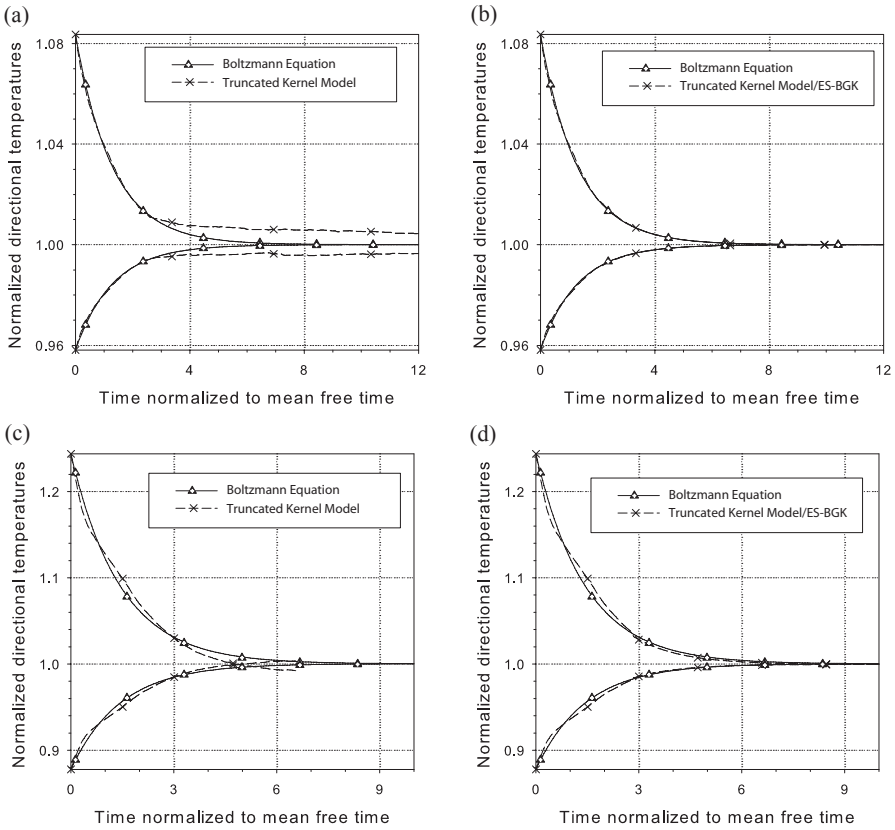

FIGURE 2. Directional temperatures in the solutions to the problem of spatially homogeneous relaxation for the initial data corresponding to Mach 1.55 (a), (b) and Mach 3.0 (c), (d) shock waves. The solutions shown in Figures (a) and (c) are obtained by using the truncated kernel method for evaluating the collision integral. The solid line corresponds to the solution using the full Boltzmann integral. The new model looses accuracy at about three mean free times for both Mach 1.55 and Mach 3.0 solutions. The solutions shown in Figures (b) and (d) are obtained by using the truncated kernel model for about three mean free times and by using the ellipsoidal-statistical Bhatnagar-Gross-Krook model after.

5. **Conclusions.** We developed an algorithm for computing the Boltzmann collision operator approximately in $O(MN)$ operations where N is the number of discrete velocity points and $M < N$. The new method is based on a truncated singular value decomposition of DG velocity discretizations of the collision operator on uniform grids. It employs deterministic and stochastic methods to evaluate the collision integral and shows a significant improvement in the solution speed as compared to the DG velocity method presented in [3]. The method, however, has significantly larger memory requirements than the original approach of [3] and appears to be slower than fast Fourier spectral methods. Accuracy of the method

has been evaluated using the solution to the problem of spatially homogeneous relaxation as well as other tests. The method shows good results for solutions that strongly deviate from continuum. It was observed, however, that the method fails to approximate accurately solutions that are near continuum. However, for such solutions, simpler models of collision operator may be used. We introduce and briefly study a discrete bilinear convolution form of the Galerkin projection of the collision operator. It is anticipated that further study of the discrete convolution form and the properties of the convolution kernel will lead to more efficient approximate methods for evaluating the Boltzmann collision integral.

Acknowledgments. The first author was supported by the NSF DMS-1620497, HPTi PETTT Project PP-SAS-KY06-001. The second author was supported by the Interdisciplinary Research Institute for the Sciences at California State University, Northridge and by the Oak Ridge Institute for Science and Education. The third author was supported by AFOSR Grant No. F4FGA04296J003. Computer resources were provided by the Extreme Science and Engineering Discovery Environment, supported by National Science Foundation Grant No. OCI-1053575 and by the Department of Defense High Performance Computing Defense Shared Resource Center at AFRL, Wright-Patterson AFB, Ohio. Additional computational resources were provided by the Rosen Center for Advanced Computing at Purdue University and by the Department of Mathematics at Purdue University.

REFERENCES

[1] A. Alekseenko and C. Euler, A Bhatnagar–Gross–Krook kinetic model with velocity-dependent collision frequency and corrected relaxation of moments, *Continuum Mechanics and Thermodynamics*, **28** (2016), 751–763.

[2] A. Alekseenko and E. Josyula, Deterministic solution of the Boltzmann equation using a discontinuous Galerkin velocity discretization, in *28th International Symposium on Rarefied Gas Dynamics, 9-13 July 2012, Zaragoza, Spain*, AIP Conference Proceedings, American Institute of Physics, 2012, 8pp.

[3] A. Alekseenko and E. Josyula, Deterministic solution of the spatially homogeneous Boltzmann equation using discontinuous Galerkin discretizations in the velocity space, emphJournal of Computational Physics, **272** (2014), 170–188, URL http://www.sciencedirect.com/science/article/pii/S0021999114002186.

[4] L. Andallah and H. Babovsky, A discrete Boltzmann equation based on a cub-octahedron in $\mathbb{R}^3$, *SIAM Journal on Scientific Computing*, **31** (2009), 799–825.

[5] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney and D. Sorensen, *LAPACK Users' Guide*, 3rd edition, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999.

[6] V. V. Aristov, *Direct Methods for Solving the Boltzmann Equation and Study of Nonequilibrium Flows*, Fluid Mechanics and Its Applications, Kluwer Academic Publishers, 2001.

[7] V. V. Aristov and S. A. Zabelok, A deterministic method for the solution of the Boltzmann equation with parallel computations, *Zhurnal Vychislitel'noi Tekhniki i Matematicheskoi Physiki*, **42** (2002), 425–437.

[8] H. Babovsky, Kinetic models on orthogonal groups and the simulation of the Boltzmann equation, *AIP Conference Proceedings*, **1084** (2008), 415–420, URL http://scitation.aip.org/content/aip/proceeding/aipcp/10.1063/1.3076513.

[9] P. L. Bhatnagar, E. P. Gross and M. Krook, A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems, *Phys. Rev.*, **94** (1954), 511–525.

[10] G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford Engineering Science Series, Oxford University Press, New York, USA, 1995.

[11] C. M. Bishop, *Neural Networks for Pattern Recognition*, Advanced Texts in Econometrics, Clarendon Press, 1995.

[12] A. V. Bobylev and S. Rjasanow, Difference scheme for the Boltzmann equation based on the fast Fourier transform., *European Journal of Mechanics - B/Fluids*, **16** (1997), 293–306.
[13] A. V. Bobylev and S. Rjasanow, Fast deterministic method of solving the Boltzmann equation for hard spheres, *European Journal of Mechanics - B/Fluids*, **18** (1999), 869–887, URL http://www.sciencedirect.com/science/article/pii/S0997754699001211.
[14] I. D. Boyd, Vectorization of a Monte Carlo simulation scheme for nonequilibrium gas dynamics, *Journal of Computational Physics*, **96** (1991), 411–427, URL http://www.sciencedirect.com/science/article/pii/002199919190243E.
[15] J. M. Burt, E. Josyula and I. D. Boyd, Novel Cartesian implementation of the direct simulation Monte Carlo method, *Journal of Thermophysics and Heat Transfer*, **26** (2012), 258–270.
[16] L. Devroye, General principles in random variate generation, in *Non-Uniform Random Variate Generation*, Springer New York, 1986, 27–82.
[17] G. Dimarco and L. Pareschi, Numerical methods for kinetic equations, *Acta Numerica*, **23** (2014), 369–520.
[18] I. D. Dinov, Expectation maximization and mixture modeling tutorial, in *Statistics Online Computational Resource*, UCLA: Statistics Online Computational Resource, 2008, URL http://escholarship.org/uc/item/1rb70972.
[19] F. Filbet and C. Mouhot, Analysis of spectral methods for the homogeneous Boltzmann equation, *Transactions of the American Mathematical Society*, **363** (2011), 1947–1980, URL http://www.jstor.org/stable/41104652.
[20] F. Filbet, C. Mouhot and L. Pareschi, Solving the Boltzmann equation in $N \log_2 N$, *SIAM Journal on Scientific Computing*, **28** (2006), 1029–1053.
[21] F. Filbet, L. Pareschi and T. Rey, On steady-state preserving spectral methods for homogeneous Boltzmann equations, *Comptes Rendus Mathematique*, **353** (2015), 309–314, URL http://www.sciencedirect.com/science/article/pii/S1631073X15000412.
[22] E. Fonn, P. Grohs and R. Hiptmair, Hyperbolic cross approximation for the spatially homogeneous Boltzmann equation, *IMA Journal of Numerical Analysis*, **35** (2015), 1533–1567.
[23] R. O. Fox and P. Vedula, Quadrature-based moment model for moderately dense polydisperse gas-particle flows, *Industrial and Engineering Chemistry Research*, **49** (2010), 5174–5187.
[24] I. M. Gamba and S. H. Tharkabhushanam, Spectral-Lagrangian methods for collisional models of non-equilibrium statistical states, *J. Comput. Phys.*, **228** (2009), 2012–2036.
[25] I. M. Gamba and S. H. Tharkabhushanam, Shock and boundary structure formation by spectral-lagrangian methods for the inhomogeneous Boltzmann transport equation, *Journal of Computational Mathematics*, **28** (2010), 430–460.
[26] I. M. Gamba and C. Zhang, A conservative discontinuous Galerkin scheme with $O(n^2)$ operations in computing Boltzmann collision weight matrix, in *29th International Symposium on Rarefied Gas Dynamics, July 2014, China*, AIP Conference Proceedings, American Institute of Physics, 2014, 8pp.
[27] B. I. Green and P. Vedula, Validation of a collisional lattice Boltzmann method, in *20th AIAA Computational Fluid Dynamics Conference, 27-30 June 2011, Honolulu Hawaii*, AIP Conference Proceedings, American Institute of Physics, 2011, 14pp.
[28] L. H. Holway, New statistical models for kinetic theory: Methods of construction, *Phys. Fluids*, **9** (1966), 1658–1673.
[29] M. Ivanov, A. Kashkovsky, S. Gimelshein, G. Markelov, A. Alexeenko, Y. Bondar, G. Zhukova and S. Nikiforov, SMILE system for 2D/3D DSMC computations, in *25th International Symposium on Rarefied Gas Dynamics, 21–28 July 2006, St. Petersburg, Russia*, AIP Conference Proceedings, Publishing House of the Siberian Branch of the Russian Academy of Sciences, Novosibirsk, Russia, 2007, 8pp.
[30] D. Kalman, A singularly valuable decomposition: The SVD of a matrix, *College Math Journal*, **27** (1996), 2–23.
[31] R. Kirsch and S. Rjasanow, A weak formulation of the Boltzmann equation based on the Fourier transform, *Journal of Statistical Physics*, **129** (2007), 483–492.
[32] Y. Y. Kloss, F. G. Tcheremissine and P. V. Shuvalov, Solution of the Boltzmann equation for unsteady flows with shock waves in narrow channels, *Computational Mathematics and Mathematical Physics*, **50** (2010), 1093–1103.
[33] R. Larsen, PROPACK: Computing the singular value decomposition of large and sparse or structured matrices. Computer software, 2005.
[34] C. D. Levermore, Moment closure hierarchies for kinetic theories, *Journal of Statistical Physics*, **83** (1996), 1021–1065.

[35] A. Majorana, A numerical model of the Boltzmann equation related to the discontinuous Galerkin method, *Kinetic and Related Models*, **4** (2011), 139–151.
[36] C. Mouhot and L. Pareschi, Fast algorithms for computing the Boltzmann collision operator, *Mathematics of Computation*, **75** (2006), 1833–1852, URL http://www.jstor.org/stable/4100126.
[37] A. Munafò, J. R. Haack, I. M. Gamba and T. E. Magin, A spectral-lagrangian Boltzmann solver for a multi-energy level gas, *Journal of Computational Physics*, **264** (2014), 152–176, URL http://www.sciencedirect.com/science/article/pii/S0021999114000631.
[38] A. Narayan and A. Klöckner, Deterministic numerical schemes for the Boltzmann equation, arXiv:0911.3589.
[39] V. A. Panferov and A. G. Heintz, A new consistent discrete-velocity model for the Boltzmann equation, *Mathematical Methods in the Applied Sciences*, **25** (2002), 571–593.
[40] L. Pareschi and B. Perthame, A Fourier spectral method for homogeneous Boltzmann equations, *Transport Theory and Statistical Physics*, **25** (1996), 369–382.
[41] C. R. Schrock and A. W. Wood, Convergence of a distributional Monte Carlo method for the Boltzmann equation, *Advances in Applied Mathematics and Mechanics*, **4** (2012), 102–121.
[42] C. R. Schrock and A. W. Wood, Distributional Monte Carlo solution technique for rarefied gasdynamics, *Journal of Thermophysics and Heat Transfer*, **26** (2012), 185–189.
[43] N. Selden, C. Ngalande, N. Gimelshein, S. Gimelshein and A. Ketsdever, Origins of radiometric forces on a circular vane with a temperature gradient, *Journal of Fluid Mechanics*, **634** (2009), 419–431, URL http://journals.cambridge.org/article_S0022112009007976.
[44] E. M. Shakhov, Approximate kinetic equations in rarefied gas theory, *Fluid Dynamics*, **3** (1968), 112–115.
[45] E. M. Shakhov, Generalization of the Krook kinetic relaxation equation, *Fluid Dynamics*, **3** (1968), 95–96.
[46] K. Stephani, D. Goldstein and P. Varghese, Generation of a hybrid DSMC/CFD solution for gas mixtures with internal degrees of freedom, in *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Aerospace Sciences Meetings, American Institute of Aeronautics and Astronautics, 2012, p648.
[47] H. Struchtrup, *Macroscopic Tansport Equations for Rarefied Gas Flows. Approximation Methods in Kinetic Theory*, Interaction of Mechanics and Mathematics Series, Springer, Heidelberg, 2005.
[48] S. Succi, *The Lattice Boltzmann Equation: For Fluid Dynamics and Beyond*, Numerical Mathematics and Scientific Computation, Clarendon Press, Oxford, 2013, URL https://books.google.com/books?id=0C0Sj_xgnhAC.
[49] F. G. Tcheremissine, Solution to the Boltzmann kinetic equation for high-speed flows, *Computational Mathematics and Mathematical Physics*, **46** (2006), 315–329.
[50] F. G. Tcheremissine, Method for solving the Boltzmann kinetic equation for polyatomic gases, *Computational Mathematics and Mathematical Physics*, **52** (2012), 252–268.
[51] V. A. Titarev, Efficient deterministic modelling of three-dimensional rarefied gas flows, *Communications in Computational Physics*, **12** (2012), 162–192.
[52] L. Wu, C. White, T. J. Scanlon, J. M. Reese and Y. Zhang, Deterministic numerical solutions of the Boltzmann equation using the fast spectral method, *Journal of Computational Physics*, **250** (2013), 27–52, URL http://www.sciencedirect.com/science/article/pii/S0021999113003276.
[53] L. Wu, J. Zhang, J. M. Reese and Y. Zhang, A fast spectral method for the Boltzmann equation for monatomic gas mixtures, *Journal of Computational Physics*, **298** (2015), 602–621, URL http://www.sciencedirect.com/science/article/pii/S0021999115004167.

Received March 2016; revised September 2017.

E-mail address: alexander.alekseenko@csun.edu
E-mail address: nguyen.142@wright.edu
E-mail address: Aihua.Wood@afit.edu