Journal of Computational Physics 230 (2011) 1265–1280

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Monte Carlo solution of the Boltzmann equation via a discrete velocity model

A.B. Morris *, P.L. Varghese, D.B. Goldstein

ASE-EM Department, University of Texas at Austin, 1 University Station, C0600, Austin, TX 78712, USA

ARTICLE INFO

Article history:
Received 18 February 2010
Received in revised form 27 October 2010
Accepted 28 October 2010
Available online 5 November 2010

Keywords:
Boltzmann equation
Discrete velocity model
Non-equilibrium gas flows
Monte Carlo method

ABSTRACT

A new discrete velocity scheme for solving the Boltzmann equation is described. Directly solving the Boltzmann equation is computationally expensive because, in addition to working in physical space, the nonlinear collision integral must also be evaluated in a velocity space. Collisions between each point in velocity space with all other points in velocity space must be considered in order to compute the collision integral most accurately, but this is expensive. The computational costs in the present method are reduced by randomly sampling a set of collision partners for each point in velocity space analogous to the Direct Simulation Monte Carlo (DSMC) method. The present method has been applied to a traveling 1D shock wave. The jump conditions across the shock wave match the Rankine–Hugoniot jump conditions. The internal shock wave structure was compared to DSMC solutions, and good agreement was found for Mach numbers ranging from 1.2 to 10. Since a coarse velocity discretization is required for efficient calculation, the effects of different velocity grid resolutions are examined. Additionally, the new scheme's performance is compared to DSMC and it was found that upstream of the shock wave the new scheme performed nearly an order of magnitude faster than DSMC for the same upstream noise. The noise levels are comparable for the same computational effort downstream of the shock wave.

1. Introduction

Direct numerical simulation of the Boltzmann equation via a discrete velocity model can be used to describe a non-equilibrium gas flow, but computational costs limit its use. The Direct Simulation Monte Carlo (DSMC) method [1] is commonly used to solve the Boltzmann equation, but the macroscopic flow variables are subject to substantial statistical fluctuations. Since there is little statistical noise associated with the macroscopic flow variables in a direct solution of the Boltzmann equation via a discrete velocity model, such a method may be more efficient than the DSMC method, particularly for transient problems.

In a discrete velocity model, the infinite velocity space is truncated and discretized. Instead of having a continuum of velocities, a fixed set of discrete velocities approximate the velocity space. For most problems of interest, the combined phase space will have at least 6 dimensions, three components for physical space and three for velocity space. If additional physics are specified, such as chemistry or internal energy, the dimensionality of the problem will be larger. This leads to expensive calculations and consequently the velocity space must be limited to a relatively coarse discretization for efficient calculations.

* Corresponding author.
E-mail address: morria2@gmail.com (A.B. Morris).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.10.037

Discrete velocity methods have been used in schemes such as the Hicks–Yen–Nordsieck (HYN) method, [2,3]. Similar numerical strategies have been developed by Tcheremissine in [4]. In the HYN scheme, the collision integral is approximated by a Monte Carlo method. To approximate the collision integral, a mean value of the integral is computed by sampling a sufficiently large set of N values of the integrand. Since the collision integral is only approximated this scheme does not enforce mass, momentum, and energy conservation. Consequently, the HYN method requires a correction scheme to enforce conservation, but these corrections may limit the accuracy of the solution. Alternatively, a discrete velocity model is exemplified by the $\Delta - \varepsilon$ model [5]. In this model, the collisions are performed by a Monte Carlo-like method. However, in this model the post-collision velocities are restricted to those that lie on the grid. Similar schemes are discussed in [6,7]. Although these schemes enforce conservation of mass, momentum, and energy, the restricted set of grid-dependent post collision velocities can result in too many null collisions, i.e. the post and pre-collision velocities are identical, when a coarse discretization is used in velocity space. Both the HYN method and the $\Delta - \varepsilon$ model can be applied to mixed continuum and rarefied flow problems but the computational costs are high, especially for high Mach number flows where the velocity space needs to be large. Different regions of the flow can have vastly different bulk velocities, and simply coarsening the domain is not feasible because the mesh size must be similar for both the low and high velocity regions. Consequently, many grid points are required and this leads to a high computational cost.

The discrete velocity scheme described here is a further development of previous work [8,9]. Traditional discrete velocity Monte Carlo schemes are limited by the restriction that post-collision velocities must lie on the grid and there are large associated computational costs. The present scheme attempts to address both of these issues. For collision pairs with small relative velocities, there exist few post-collision velocities that lie on the grid. To overcome this problem, we pick an arbitrary orientation of the post-collision velocity vector and use an interpolation scheme to map those points back onto the grid. This interpolation routine precisely conserves mass, momentum, and energy and can be extended to points that lie entirely outside of the finite sub-set of velocity space that is used in the computation. Similar interpolation schemes were developed by Tcheremissine [10–12] by interpolating to nearby nodes that are selected such that the symmetry ensures that momentum is automatically conserved. Then, an additional constraint is enforced such that the energy is conserved. Although conservative, this interpolation scheme is valid only for uniform velocity grids and does not readily accommodate post-collision velocities that lie outside of the velocity space. Other methods, e.g. [13], have generalized Tcheremissine's interpolation method to non-uniform grids but these schemes introduce slight errors in conservation of mass, momentum, and energy. To eliminate these errors, a correction scheme was implemented.

The nonlinear collision integral is responsible for most of the computational effort. Varghese showed that accurate evaluation of the replenishing portion of the integral can be obtained using a coarse approximation, only replenishing to several pairs [8]. In that work the depleting integral was evaluated by considering collisions between each point in velocity space with every other point in velocity space. Since accurate results were obtained using a coarse description of the replenishing integral, a coarser description of the depleting integral was also developed [9]. To coarsen evaluation of the depleting integral, collisions are performed for *each* point in velocity space with a *representative* set of partners that are chosen randomly. It was shown that accurate results could be obtained for significantly less computational effort by performing collisions with a representative set of partners instead of all possible collision partners. By varying how many collision partners are sampled, one can control how accurately to evaluate the depleting collision integral depending on noise requirements.

A traveling shock wave is used as the test case to evaluate the performance of our scheme. Normal shock waves are commonly used to validate and verify a numerical scheme and related work can be found in [14,15]. To generate the shock wave, a gas initially moving left is brought to rest by a specular wall. A shock forms at the wall and propagates upstream, bringing the flow to rest while increasing the temperature and density. At the right boundary, the distribution function is set by a uniform inflow boundary condition. In order to assess the accuracy of the numerical solution, comparisons to analytic values are made outside of the shock. Comparisons are made to DSMC inside the shock. Jump conditions for density, pressure, and temperature across the shock, are determined by the Rankine–Hugoniot equations and the shock propagation speed is determined from continuum dynamics. The internal shock structure depends on molecular cross sections and in the following cases pseudo-Maxwell molecules are used.

2. Formulation

The Boltzmann equation describes the time evolution of the distribution function, φ, and for isotropic scattering without body forces can be written in dimensional form as:

$$\frac{\partial \varphi}{\partial t} + \eta_i \frac{\partial \varphi}{\partial x_i} = \int_\eta \left[\varphi(\eta_i')\varphi(\zeta_i') - \varphi(\eta_i)\varphi(\zeta_i)\right] g \sigma_T \, dV_\eta \tag{1}$$

The left hand side of this Eq. (1) represents the total derivative of the (density weighted) velocity distribution function and the right hand side represents the changes in the distribution function resulting from collisions. It is often convenient to work with scaled variables and in the present work velocities are scaled by the most probable thermal speed $\eta_r \equiv \sqrt{2k_b T_r/m}$, distances are scaled by a reference mean length L, and time is scaled by a characteristic flow time. The scaled variables are denoted by hats and are defined as:

$$\hat{x} \equiv \frac{x}{L}, \quad \hat{t} \equiv \frac{L}{\eta_r}, \quad \left(\hat{g}, \hat{\eta}, \hat{\zeta}\right) \equiv \frac{(g, \eta, \zeta)}{\eta_r}, \quad \hat{\sigma}_T \equiv \frac{\sigma}{\sigma_r}, \quad \hat{\varphi} = \frac{\eta_r^3}{n_r}\varphi \tag{2}$$

The reference temperature and density are defined as T_r and n_r. In non-dimensional form, Eq. (1) can be rewritten as:

$$\frac{\partial \hat{\varphi}}{\partial \hat{t}} + \hat{\eta}_i \frac{\partial \hat{\varphi}}{\partial \hat{x}_i} = \frac{1}{Kn} \int_{\hat{\eta}} \left[\hat{\varphi}(\hat{\eta}_i')\hat{\varphi}(\hat{\zeta}_i') - \hat{\varphi}(\hat{\eta}_i)\hat{\varphi}\left(\hat{\zeta}_i\right)\right] \hat{g}\hat{\sigma}_T \, d\vec{V}_{\hat{\eta}} \tag{3}$$

The Knudsen number, $Kn \equiv \lambda_r/L$, is the ratio between the reference mean free path and a characteristic length scale of the problem and is used to quantify the rarefaction of a flow. The reference mean free path is computed via $\lambda_r = 1/(n_r\sigma_r)$ where σ_r is a reference total collision cross section. When comparing computed shock wave profiles to those computed via DSMC, it is common to use the upstream mean free path as the characteristic length scale, L. The upstream mean free path for a VHS gas is defined in Appendix A. For elastic collisions and isotropic scattering, the direction of the post-collision relative velocity vector is distributed uniformly on the sphere centered about the center of mass velocity. The relative speed, $\hat{g}$, is conserved in a collision to preserve conservation of energy, and to preserve conservation of momentum the center of mass velocity, $\widehat{G}_i$, is constant. The primed values indicate post-collision velocities, and for elastic collisions the post-collision velocities are any velocities that satisfy Eqs. (4a) and (4b).

$$\left|\hat{\eta}_i' - \hat{\zeta}_i'\right| = \left|\hat{\eta}_i - \hat{\zeta}_i\right| \equiv \hat{g} \tag{4a}$$

$$\hat{\eta}_i' + \hat{\zeta}_i' = \hat{\eta}_i + \hat{\zeta}_i \equiv 2\overline{G}_i \tag{4b}$$

The collision integral is nonlinear and can be decomposed into depleting and replenishing collisions. This decomposition is convenient because the proposed method computes estimates for both integrals simultaneously by pairing depleting collisions with their corresponding replenishing collisions while ensuring conservation of mass, momentum, and energy:

$$\widehat{I}_{coll} = \widehat{I}_{repl} - \widehat{I}_{depl} \tag{5}$$

$$\widehat{I}_{depl} = \int_{\hat{\eta}} \left[\hat{\varphi}(\hat{\zeta}_i)\hat{\varphi}(\hat{\eta}_i)\right] \hat{g}\hat{\sigma}_T \, dV_{\hat{\eta}} \tag{6}$$

$$\widehat{I}_{repl} = \int_{\hat{\eta}} \left[\hat{\varphi}(\hat{\zeta}_i')\hat{\varphi}(\hat{\eta}_i')\right] \hat{g}\hat{\sigma}_T \, dV_{\hat{\eta}} \tag{7}$$

There are numerous models that are used to describe the total cross section, such as Lennard-Jones [16], hard sphere models, variable hard spheres (VHS) [17], and variable soft spheres (VSS) [18]. In the present numerical scheme, a VHS model is used. The cross section in a VHS model, written in scaled variables, is given by $\hat{\sigma}_T = \hat{g}^{-\kappa}$ where κ is typically fractional. The cross section is scaled by a reference cross section and the relative speed is scaled by η_r. When $\kappa = 0$ the collision cross section reduces to a hard sphere model. When $\kappa = 1$ the total cross section corresponds to pseudo-Maxwell molecules. In this work, we only consider the specific case for pseudo-Maxwell molecules. Pseudo-Maxwell molecules are not identical to "real" Maxwell molecules because the scattering for "real" Maxwell molecules is not isotropic whereas in the VHS model the scattering is isotropic. If pseudo-Maxwell molecules are used, $\hat{\sigma}_T\hat{g} = 1$. Introducing a simple first order time discretization the contribution of the collision integral to the change in the scaled distribution function is:

$$\Delta\hat{\varphi}_{coll} = \overline{I}_{coll}\Delta\hat{t} = \frac{1}{Kn}\left[\Delta\hat{\varphi}_{repl} - \Delta\hat{\varphi}_{depl}\right] \tag{8}$$

where,

$$\Delta\hat{\varphi}_{depl}(\hat{\zeta}_i) = \Delta\hat{t}\hat{\varphi}(\hat{\zeta}_i) \int_{\hat{\eta}_i} \hat{\varphi}(\hat{\eta}_i) d\hat{\eta}_i = \Delta\hat{t}\hat{\varphi}(\hat{\zeta}_i)\hat{n} \tag{9}$$

$$\Delta\hat{\varphi}_{repl}(\hat{\zeta}_i) = \Delta\hat{t} \int_{\hat{\eta}_i} \hat{\varphi}(\hat{\zeta}_i')\hat{\varphi}(\hat{\eta}_i') d\hat{\eta}_i \tag{10}$$

In Eq. (9), $\hat{n}$ is the scaled number density. Eq. (9) can be used to calculate the incremental depletion at grid point $\hat{\eta}_i$ as a result of depleting collisions with all other velocities. Eq. (10) represents the incremental gain at grid point $\hat{\eta}_i$ as a result of replenishing collisions. The incremental depletion is a function of the pre-collision velocities and can therefore be evaluated directly.

3. Numerical strategy

First, the physical space is divided into cells and corresponding velocity distributions are initialized for each physical cell. The cell size for rarefied flows should be of the same order as the mean free path. The distribution function for each cell is

initialized by evaluating a continuous Maxwellian with a specified density, velocity, and temperature at each discrete velocity. Since the collision integral is computationally expensive, a rather coarse discrete representation must be used. When creating a velocity grid, both the range and the resolution of the grid are important. The nominally infinite velocity space is truncated where the contributions to the first several moments of the distribution function become vanishingly small. The resolution of the grid affects several different integration routines and the collision routine. In our representation, the velocity distribution is viewed as a set of delta functions and integration is performed by summation over the delta functions. Although a continuous Maxwellian has a specified temperature, velocity and density, the computed density, velocity, and temperature from the corresponding discrete distribution will have errors associated with the numerical integration. A coarse resolution also restricts the potential velocities that the molecules can have. Since real gas molecules have a continuum of velocities, a limited set of discrete velocities can restrict the accuracy of the computations. The characteristic velocity step used to discretize the velocity space is denoted by β, where $\beta \equiv \Delta\eta/\eta_r$. Typical values used for the scaled velocity spacing, β, range from 0.6 to 0.9 and the velocity domain typically ranges over $\pm 2.5\eta_r$. This discretization results in Gaussian distributions being represented by approximately 11 velocities in each direction. The velocity domain needs to be expanded as the gas temperature increases.

The collision integral and the convective terms are computed separately when solving the Boltzmann equation.

$$\Delta\hat{\varphi} \equiv \frac{\partial\hat{\varphi}}{\partial\hat{t}}\Delta\hat{t} = \Delta\hat{\varphi}_{collision} + \Delta\hat{\varphi}_{convection} \tag{11}$$

$$\Delta\hat{\varphi}_{convection} = -\hat{\eta}_i \frac{\partial\hat{\varphi}}{\partial\hat{x}_i}\Delta\hat{t} \tag{12}$$

$$\Delta\hat{\varphi}_{collision} = \bar{I}_{coll}\Delta\hat{t} \tag{13}$$

As discussed earlier (Eqs. (8)–(10)), the collision integral can be decomposed into paired depleting and replenishing collisions that allows the two terms to be approximated simultaneously while ensuring that mass momentum, and energy are strictly conserved.

4. Evaluation of the collision integral

For pseudo-Maxwell molecules the depletion at each velocity is given by Eq. (9). The integrand of Eq. (9) can be interpreted as the depletion associated with collisions between velocities $\hat{\eta}_i$ and $\hat{\zeta}_j$. In previous work [8], the collision integral was evaluated by considering collisions between each point $\hat{\eta}_i$ and all other points $\hat{\zeta}_j$. Although this results in correct evaluation of the depleting integral, it is computationally expensive and of order N_V^2, where N_V is the number of points in velocity space. To reduce the computational effort in this work, the depleting integral is instead evaluated statistically by considering collisions between each point $\hat{\eta}_i$ and a set of randomly chosen collision partners, M_j as illustrated in Fig. 1. For efficiency, collision partners are selected using importance sampling.

Physically speaking the distribution function must be positive, but the value of $\hat{\varphi}$ occasionally becomes slightly negative at a few locations in velocity space because of the approximate nature of the computation of the collision integral (see below). This typically occurs near the fringes of the computational domain in velocity space where $\hat{\varphi}$ is small. To account for negative values of the distribution function, collision partners are selected based on magnitude of the value of the distribution function. Collisions between identical velocities are forbidden, $\hat{\zeta}_j \neq \hat{\eta}_i$, because such a case corresponds to a null collision. The probability of selecting a velocity $\hat{\zeta}_j$ to collide with $\hat{\eta}_i$ is therefore given as:

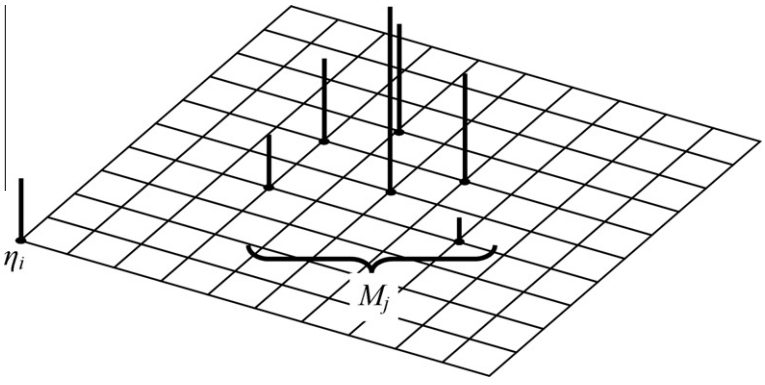

Fig. 1. A set of random collision partners M_j is chosen to collide with η_i. The selection of collision partners is biased toward those that contribute more to the collision integral.

$$P(\hat{\eta}|\hat{\zeta}) = \frac{\left|\hat{\varphi}(\hat{\zeta})\right|}{\sum_{\hat{\zeta}^* \neq \hat{\eta}} \left|\hat{\varphi}\left(\hat{\zeta}^*\right)\right|} \tag{14}$$

The amount depleted from each partner for collisions between $\hat{\eta}_i$ and each velocity $\hat{\zeta}_j$ in M_j is

$$\Delta\hat{\varphi}_{depl}\left(\hat{\eta}_i|\hat{\zeta}_j\right) = \frac{1}{2M_j}\hat{\varphi}(\hat{\eta}_i)\Delta\hat{t}\beta^3\left(\sum_{\hat{\zeta}_j^* \neq \eta_i}\left|\hat{\varphi}\left(\hat{\zeta}_j^*\right)\right|\right)\mathrm{sgn}\left(\hat{\varphi}\left(\hat{\zeta}_j\right)\right) \tag{15}$$

where

$$\mathrm{sgn}(x) = \begin{cases} 1 & x \geqslant 0 \\ -1 & x < 0 \end{cases} \tag{16}$$

The factor of 0.5 is used to prevent double counting of the depletion, and it can be shown that the expectation value of Eq. (15) correctly matches the analytic depletion [21].

4.1. Picking collision partners

Many points in velocity space contain trace amounts of mass, especially near the fringes of the domain and consequently they contribute little to the collision integral. The goal of the proposed selection algorithm is to estimate the collision integral efficiently by selecting collision partners that make the largest contribution. For Maxwell molecules, the contribution from each point in velocity space to the depleting collision integral is directly proportional to the value of the distribution function. This can be implemented via an acceptance–rejection algorithm. In traditional acceptance–rejection schemes two random numbers are drawn. First a random point in velocity space is selected. A second random number is generated after the first random point is selected and that point is accepted if the local value of the distribution function is larger than the second random number. If it is not accepted, a new set of random numbers is drawn. This scheme can be computationally expensive because of the costs associated with generating random numbers. An alternative method that reduces the number of random number calls by using a cumulative distribution function has been developed and is discussed in Appendix B.

The cumulative scheme avoids having to reject randomly sampled points, based on another random number call, and thus reduces the number of random number calls in two ways. However, as noted in Appendix B, a search is required. Collision partners can be selected more than once and for a given test point $\hat{\eta}_i$, partners are selected until the set of M_j randomly chosen collision partners has a cumulative mass fraction larger than a cutoff mass fraction. The cutoff fraction is referred to as the depleting fraction, *DF*. A depleting fraction of 1.0 considers collisions with all other points in velocity space and similarly a depleting fraction of 0.1 samples collisions from only 10% of the velocity space. This is a form of importance sampling and, to further improve the efficiency, a linearly varying depleting fraction is used. For test points with small $\hat{\varphi}(\hat{\eta}_i)$, e.g. points near the fringes, a smaller depleting fraction is used for the collision partners because those collisions contribute less to the collision integral than points with large $\hat{\varphi}$. This is implemented by imposing a minimum and maximum depleting fraction. The minimum depleting fraction is used for the point with the smallest absolute value of $\hat{\varphi}$ and the maximum depleting fraction is used for the point with the largest $\hat{\varphi}$; at all other points the depleting fraction chosen varies linearly between the minimum and maximum.

4.2. Replenishing collisions

The inverse replenishing collisions are computed simultaneously with the corresponding direct depleting collisions. For elastic collisions mass, momentum, and energy are conserved and, in a discrete velocity framework, all possible collision outcomes lie on the sphere of diameter g centered about the center of mass velocity, drawn in 2D in Fig. 2.

In the two-dimensional Fig. 2 the depleting collisions are performed at η and ζ. To find the post collision velocities, the relative velocity vector g is arbitrarily rotated about the center of mass velocity. These replenishing velocities correspond to the dashed and dotted vectors, g' and g''. In principle, the replenishing integral is evaluated by considering all possible post-collision velocities. In the current approach, the integral is approximated by sampling several replenishing pairs. The total amount replenished equals the amount depleted for each collision, ensuring mass, momentum, and energy conservation. The replenishment is then shared equally among all post-collision velocities. In a discrete representation, the only permissible velocities are those that lie precisely on the velocity grid. Fig. 2 shows that this restriction can greatly limit the number of post collision velocities because only the dashed vector, g', lies on the grid. In general, as the relative velocity becomes smaller, the number of potential post-collision velocities that lie on the grid decreases [22]. For collisions between molecules with very small relative velocities, there may be no post-collision velocities that lie on the grid. In Fig. 2, this null type of collision is represented by the dashed circle in the upper right corner. The reduction of possible collision outcomes can artificially restrict the approach to equilibrium. Consequently, we present an approach wherein the post-collision velocities that lie off the grid (e.g. g'' in Fig. 2) are projected back onto the grid nodes while ensuring that mass, momentum, and energy are strictly conserved [8].

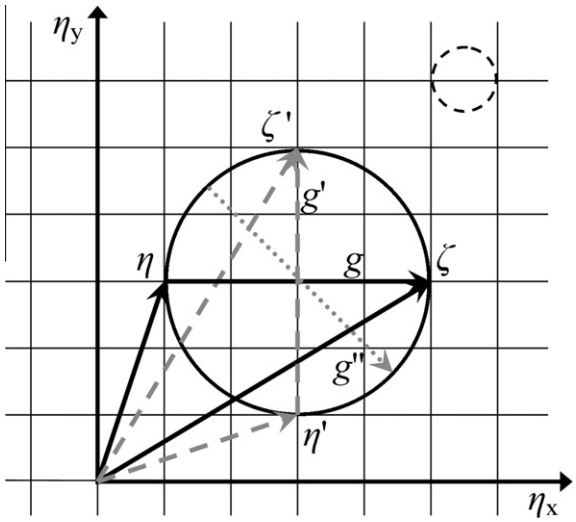

Fig. 2. Collision circle drawn in 2D. Elastic collisions between η and ζ result in post collision velocities that lie on the black circle. The solid black lines represent the pre-collision velocities. The dashed grey lines are post-collision velocities that lie on the velocity grid and the dotted grey lines are post-collision velocities that lie off the grid.

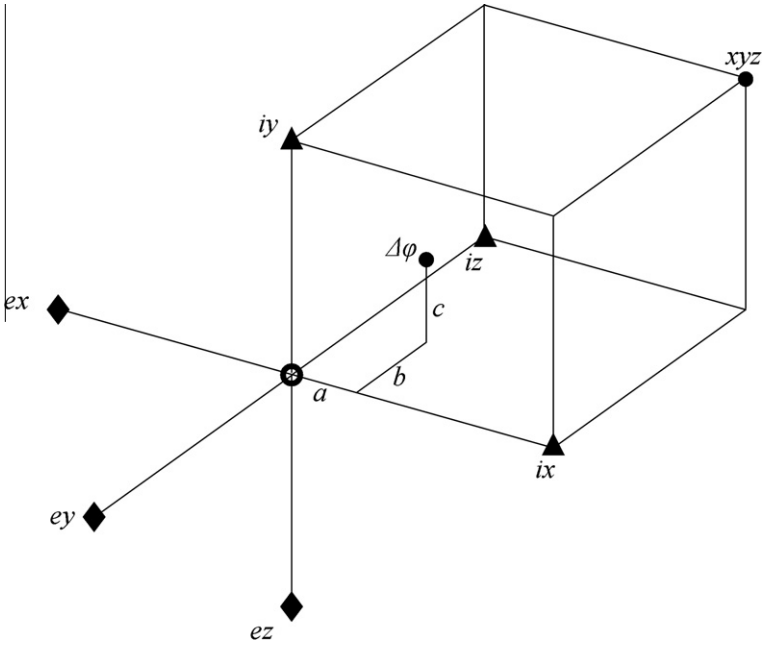

Fig. 3. A 7-point symmetric stencil when the point to be interpolated lies within the cubic velocity domain. Points *ix*, *iy*, and *iz* are interior points while *ex*, *ey* and *ez* are exterior lattice points.

4.3. Interpolation scheme

Suppose a post-collision relative velocity vector similar to g'' in Fig. 2 places an incremental change to the distribution function off the grid. While projecting distribution function changes, $\Delta\varphi$, that lie off the grid back onto the grid, the 5 conservation equations must be satisfied. First, we need to define a stencil on which the contribution $\Delta\varphi$ is. We now continue this discussion for a three dimensional interpolation suitable for a 3-D velocity space. The origin of the stencil in Fig. 3, is taken as the velocity lattice point closest to the increment to be interpolated. Suppose that the increment is located at coordinates *a*, *b*, and *c* relative to this lattice point. The origin and three other points that lie on the cube surrounding the interpolating point make up the 'interior points' that are denoted by the circle and triangles, respectively, in Fig. 3.

Although it would appear logical to use a fifth point located on the cube surrounding the interpolating point, e.g. point *xyz*, the resulting matrix describing mass, momentum, and energy conservation would be singular. It can be easily shown that this is true for any five points chosen on the corners of the cube surrounding the point. However, it is possible to obtain a unique interpolation onto the grid by using four 'internal' points that lie on the cube surrounding the point to be interpolated (the triangles and circle) and at least one 'external' point outside of the box. Although one 'external' point *ex*, *ey*, or *ez* would suffice, three external points are generally used for symmetry, and the computed change is distributed equally to all three of these 'external' points. Because the contributions at the external points are usually negative, an additional advantage of this strategy is that it reduces the possibility that the value of φ at any point becomes negative. The conservation equations for such a symmetric 7-point stencil with three 'external' points are:

$$\begin{matrix} mass \\ x-mom \\ y-mom \\ z-mom \\ energy \end{matrix}\begin{pmatrix} 1 & 1 & 1 & 1 & 3 \\ 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 1 & 1 & 1 & 3 \end{pmatrix}\begin{pmatrix} f_o \\ f_{ix} \\ f_{iy} \\ f_{iz} \\ f_{ext} \end{pmatrix} = \begin{pmatrix} 1 \\ a \\ b \\ c \\ a^2+b^2+c^2 \end{pmatrix} \tag{17}$$

The parameters a, b, and c are the components of the relative velocity vector from the stencil origin at o to the point to be interpolated. The values f_o, f_{ix}, f_{iy}, f_{iz}, and f_{ext} are the fractional density changes that are interpolated to the corresponding points at the origin, at the interior points, and at the exterior points. This matrix is invertible analytically, and the corresponding interpolation equations for such a symmetric 7-point stencil are:

$$\begin{aligned} &f_o = 1 - a^2 - b^2 - c^2 \geqslant 0; \quad -0.5 \leqslant a,b,c \leqslant 0.5 \\ &f_{ext} = -(a+b+c-a^2-b^2-c^2)/6 \leqslant 0 \\ &f_{ix} = (a + f_{ext}) \\ &f_{iy} = (b + f_{ext}) \\ &f_{iz} = (c + f_{ext}) \end{aligned} \tag{18}$$

For the stencil shown above, the interpolation equations always result in negative changes, f_{ext}, at the external points of the stencil. The negative change is necessary to satisfy the energy constraint because the only way to cancel the relative kinetic energy introduced by distributing the density at a single velocity to the same density at several velocities is to introduce a negative density somewhere. The interpolation equations, Eq. (18), were derived for the common case where the point to be interpolated lies within the velocity space domain. Occasionally collisions yield velocities that are outside the domain, as is the case for point η' in Fig. 4.

Using a similar procedure, it is also possible to map these points to the nearest stencil on the grid. To construct the stencil, the nearest point in velocity space that is not a corner of the cubic velocity domain is selected as the stencil origin. A corner of the velocity domain is not an acceptable stencil origin because it only has three points as neighbors and a 5 point stencil is required at a minimum. The internal points are then selected as the next three nearest points that form the cube that is clos-

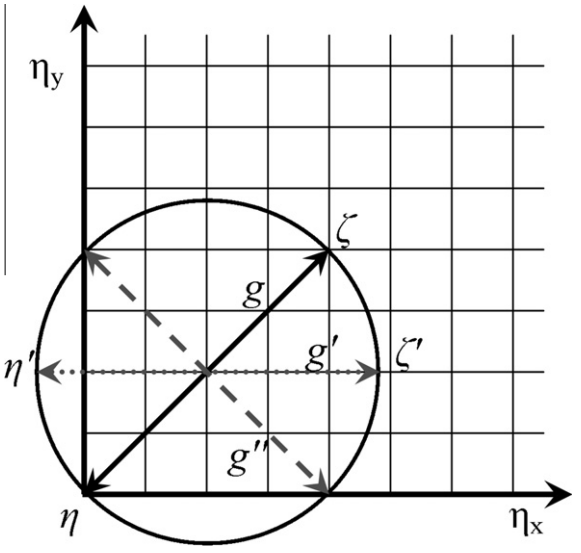

Fig. 4. A collision (dotted line) that results in a post collision velocity that lies outside of the domain.

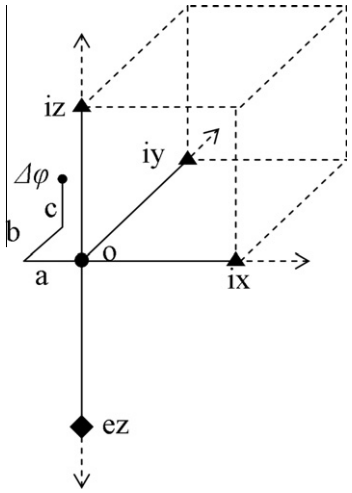

Fig. 5. A 5-point stencil (o, ix, iy, iz, ez) where the point to be interpolated lies outside of the velocity domain and is mapped back into the defined velocity space. The line connecting points *iz* and *ez* represent an edge of the velocity domain.

est to the point to be interpolated. Only one or two 'exterior' points are used if the stencil origin is on an edge of the velocity domain or a face boundary of the domain. For example, in Fig. 5, the origin corresponds to a point that is on the edge of the velocity domain and only one exterior point is used.

The corresponding matrix describing mass, momentum, and energy conservation is:

$$\begin{matrix} mass \\ x-mom \\ y-mom \\ z-mom \\ energy \end{matrix} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} f_o \\ f_{ix} \\ f_{iy} \\ f_{iz} \\ f_{ez} \end{pmatrix} = \begin{pmatrix} 1 \\ a \\ b \\ c \\ a^2+b^2+c^2 \end{pmatrix} \tag{19}$$

Similar to the previous stencil, a, b, and c are the components of the relative velocity vector from the stencil origin to the interpolating point. In Fig. 5, $a < 0$, and $b, c > 0$. For the stencil above, the interpolation equations are:

$$\begin{aligned} f_o &= 1 - a^2 - b^2 - c^2 \\ f_{ez} &= -\left(a + b + c - a^2 - b^2 - c^2\right)/2 \\ f_{ix} &= a \\ f_{iy} &= b \\ f_{iz} &= c + f_{ez} \end{aligned} \tag{20}$$

Larger negative changes occur when interpolating post-collision points located outside the computational domain in velocity space back onto the grid inside the computational domain. Since the point to be interpolated does not lie within a cube defined by the 'interior' points and origin, the negative changes can occur at the external points or interior points. One can map any point to the above 5-point stencil and conserve mass, momentum, and energy regardless of the orientation of the interpolating point relative to the stencil origin. However, larger negative corrections are needed to satisfy the conservation equations as the distance from the interpolating point to the stencil origin increases. Therefore, when an interpolating point is far from an origin, i.e. for points well outside of the domain, appreciable negative changes in density are computed. In general this is not a serious problem because collisions that result in velocities that lie outside of the grid boundaries are infrequent and the fractional densities moved back onto the grid are very small, of the order 10^{-6}.

After the replenishing velocities are mapped back onto the grid, collisions between η_i and the remainder of the random collision partner set M_j are considered. The process is then repeated for all η_i points in velocity space.

5. Convection of the distribution function

In the time-splitting approach the change in the distribution function due to convection is computed after the collision step. The boundary conditions are applied during the convective step. Since most of the computational effort is spent performing collisions, a more expensive explicit 4th order convection scheme [7] is used with little penalty. The fourth order scheme is

$$\varphi(x_i, t+\Delta t) = \alpha_{-2}\varphi(x_{i-2}, t) + \alpha_{-1}\varphi(x_{i-1}, t) + \alpha_0\varphi(x_i, t) + \alpha_1\varphi(x_{i+1}, t) + \alpha_2\varphi(x_{i+2}, t) \tag{21}$$

where the coefficients are:

$$\begin{aligned}
\alpha_{-2} &= -c(1-c^2)(2+c)/24 \\
\alpha_{-1} &= c(1+c)(4-c^2)/6 \\
\alpha_0 &= (1-c^2)(4-c^2)/4 \\
\alpha_1 &= -c(1-c)(4-c^2)/6 \\
\alpha_2 &= c(1-c^2)(2-c)/24
\end{aligned} \tag{22}$$

and $c = \eta\Delta t/\Delta x$ is the CFL number. A lower order convective scheme was tried at first, but in order for the convective step to converge a finer spatial resolution is required, and consequently the computational cost increases.

6. Numerical results

6.1. Velocity grid resolution

The discretization of the velocity space must be relatively coarse for efficient calculations, so understanding the limits on velocity grid size is important if the scheme is to be optimized. Three velocity grids with different resolutions are used to solve a traveling Mach 2.0 shock wave with a scaled upstream temperature of 1. Each velocity grid tested extends to $\pm 4\eta_r$ but the scaled velocity spacing is varied from $\beta = 0.7$ to 1.33, where we scale based on η_r computed at the upstream conditions. A variable depleting fraction ranging from 1% to 10% was used along with two replenishing pairs per collision for each calculation. In Figs. 6 and 7, the scaled density and temperature profiles are plotted at $\hat{t} = 50$ for three different values of β. At this time, the shock is far enough from the left hand wall so that boundary effects are confined near the wall and not seen in the shock wave.

When $\beta < 1.0$, the solutions show good agreement indicating that the computation has achieved grid independence with respect to velocity resolution. However when $\beta = 1.33$, the density and temperature (to a larger extent) undershoot upstream of the shock wave and the shock propagates at a slightly slower speed. The undershoot in front of the Mach 2 shock wave computed on a coarse velocity grid is not caused by the convection step or the statistical evaluation of the collision integral. This was verified by refining the spatial and temporal discretizations and evaluating the collision integral with more samples. For a stronger Mach 4 shock wave with the same upstream temperature, the undershoot occurs when a coarser velocity spacing of 2.25 is used. This indicates that the critical velocity spacing that causes undershoot to occur is Mach number dependant. Numerical experiments show that the gross errors in the upstream portion of the shock wave arise from the interpolation of post-collision velocities on very coarse grids.

To show that the interpolation scheme is responsible for the upstream undershoot, the post-collision velocities were constrained to lie within a small distance, ε, of a velocity grid point. Although inefficient, due to a large number of rejected

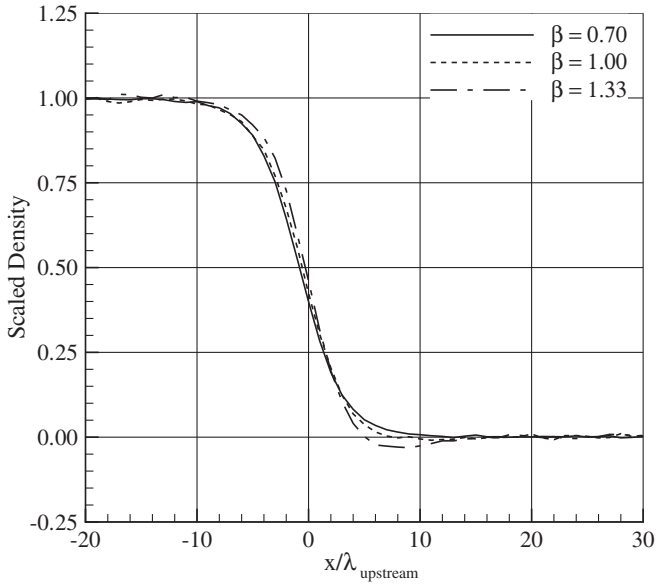

Fig. 6. Density profiles for a Mach 2 shock wave using various velocity grid resolutions.

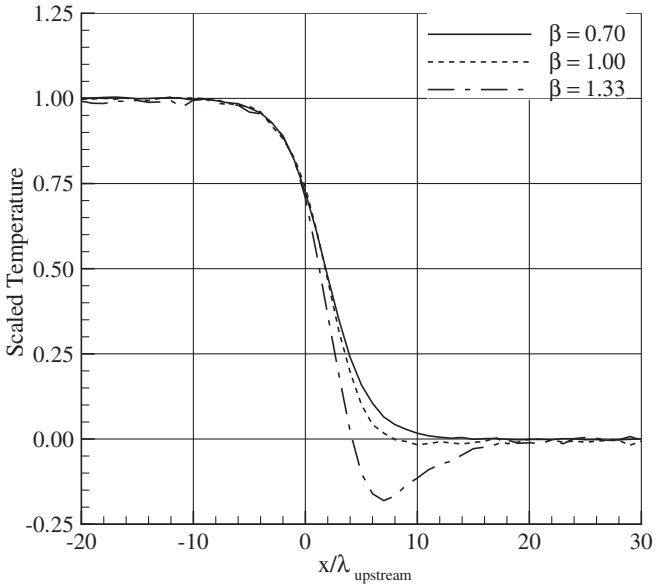

Fig. 7. Temperature profiles for a Mach 2 shock wave using various velocity grid resolutions.

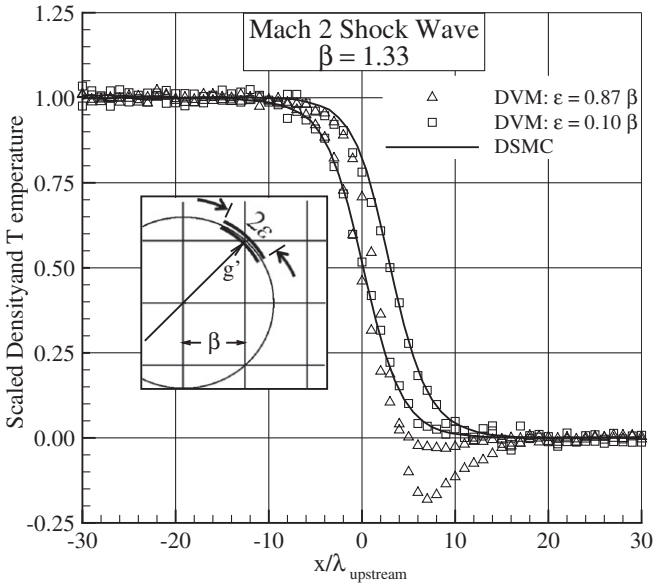

Fig. 8. Comparison of Mach 2 shock wave profiles when the post-collision velocities are restricted to those within 0.1β of a velocity grid point and when the post-collision velocities are unrestricted.

post-collision velocities, this constraint dampens the effect of interpolating floating velocities over large ranges. In Fig. 8, Mach 2 shock wave profiles are computed when the post-collision velocities are restricted to those within 0.1β of a velocity grid point and when the post-collision velocities are not restricted, i.e. $\varepsilon = \left(\sqrt{3}/2\right)\beta$. Both are compared to DSMC profiles.

Fig. 8 shows excellent agreement with DSMC is obtained when the post-collision velocities are restricted to those near grid points and poor agreement when the post-collision velocities are unrestricted. In [22] the post-collision velocities were restricted to grid points and it was found that at very coarse discretizations the shock tends to thicken because there are a significant number of null collisions, e.g. the post-collision velocities are identical to the pre-collision velocities. For a Mach 2 shock wave, the effect of null collisions is seen on grids even coarser than those when interpolation errors become apparent.

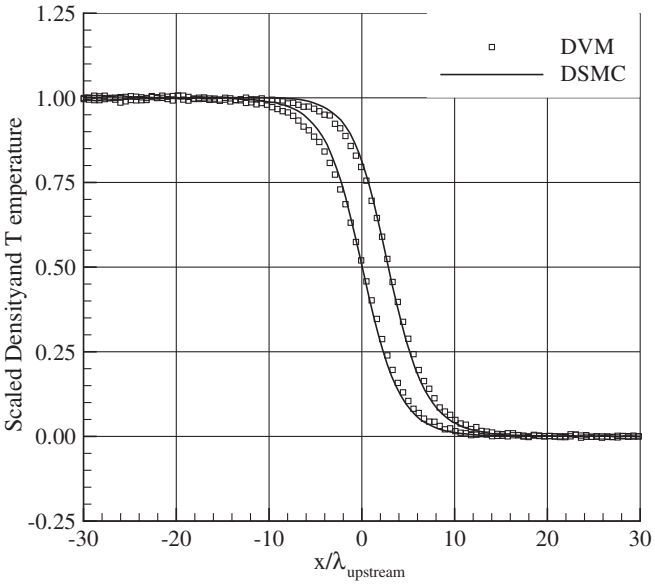

Fig. 9. Mach 2.0 shock profiles comparing discrete velocity solutions to DSMC solutions.

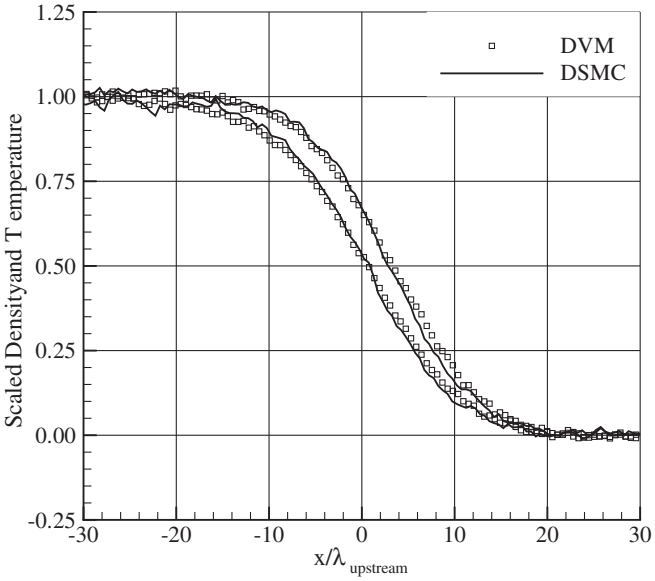

Fig. 10. Mach 1.2 shock profiles comparing discrete velocity and DSMC solutions.

6.2. Internal shock structure

The flow is not in equilibrium inside the shock wave and the shock structure depends on how the gas equilibrates. The computed internal shock structure is compared to steady state DSMC calculations for three different cases: a Mach 2.0 shock wave, a weak Mach 1.2 shock wave, and a stronger Mach 10 shock wave. The DSMC solutions were generated using Bird's DSMC1S code [1]. The DSMC method has been shown to converge to a solution for the Boltzmann equation [23] and has also been shown to agree well with experimental data [24]. A converged DSMC solution, such as those compared to in the subsequent calculations, can thus be treated as a sufficiently accurate solution to the Boltzmann equation.

In Fig. 9, a Mach 2.0 traveling shock computed by the discrete velocity model is compared to a stationary Mach 2.0 computed by DSMC. The traveling shock wave profile was taken when the shock was 70 mean free paths from the wall. A velocity space spanning to $\pm 4\eta_r$ with a scaled velocity spacing of 0.8 was used. A time step $\Delta\hat{t} = 0.05$ and a scaled physical spacing $\alpha = 0.5$ were used. A depleting fraction ranging from 0.1 to 0.4 was used along with 2 replenishing pairs per collision.

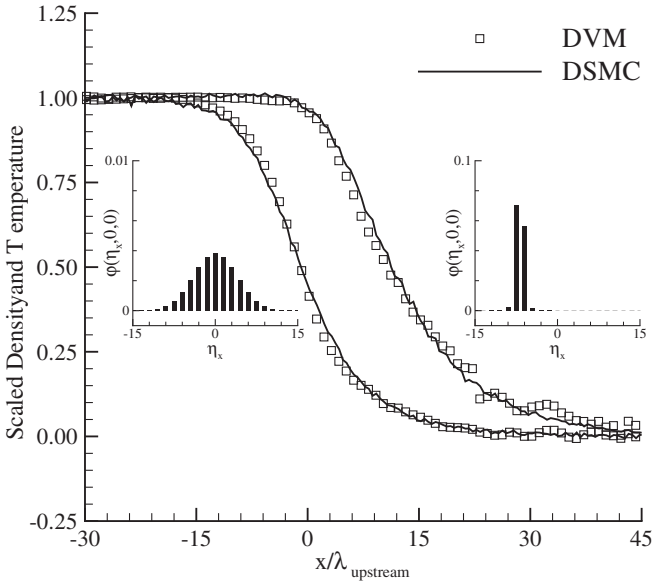

Fig. 11. Mach 10.0 shock profiles comparing discrete velocity and DSMC solutions. Upstream and downstream slices of the distribution are plotted for reference.

The computed profiles closely match the DSMC solution, but the DVM profiles are slightly thicker. This trend is seen in both the density and temperature profiles. The Rankine–Hugoniot conditions are satisfied across the shock wave and the shift between the temperature and density computed with the discrete velocity model matches DSMC. The shock thickness measured by the maximum density gradient for DSMC is measured as 9.2 mean free paths, where the shock thickness measured from the DVM is 10.1 mean free paths.

Fig. 10 shows the results of a weak shock (Mach 1.20) calculation. The statistical fluctuations are more noticeable than the higher Mach number cases because the jump in properties across a weak shock are very small. To obtain a smooth DSMC profile, 10 million simulated particles were used along with 200 cells in physical space and averaged over several iterations. A variable depleting fraction ranging from 10% to 80% was used in the discrete velocity model. Scaled time and space steps of 0.05 and 0.5, respectively were used and the physical domain extended over 200 spatial grid points. The shock was allowed to propagate for 2000 time steps and the profile was captured at $\hat{t} = 100$. Using the above conditions, the DVM solution required 208 min. At a low Mach number of 1.2 the Rankine–Hugoniot conditions are satisfied, but there are slight differences between the DSMC solution and the DVM solution. The DSMC shock wave is slightly thinner than the discrete velocity shock wave. As the Mach number is decreased below 1.25, the shock wave thickness increases rapidly [1]. The shock thickness is measured as 21.5 mean free paths from the DSMC solution and 23.0 mean free paths from the DVM solution. If the Mach numbers for the DSMC and DVM cases do not match very closely because of discretization errors in computation of the moments of the distribution, the shock thicknesses may be noticeably different.

To compute a Mach 10 shock wave, Fig. 11, the velocity domain must be very large because the temperature increases by a factor of 32 and the corresponding downstream distribution function is relatively wide and flat. In addition to needing a large velocity domain, the resolution has to be sufficient to resolve the cold and narrow upstream distribution function. In order to capture the physical moments of the distribution function, the scaled velocity domain must extend to no less than $\pm 2.5\widehat{T}^{1/2}$ downstream of the shock wave while the scaled velocity spacing, β, needs to be sufficiently small to resolve the upstream distribution. For a Mach 10 shock computation we used a velocity domain extending to ±15 with a scaled velocity spacing of 1.5. The resulting velocity space contains approximately 10,000 points and the corresponding computational costs are very large. A very low depleting fraction ranging from 0.01 to 0.05 was used along with 2 replenishing pairs per collision.

Fig. 11 shows that the computed density and temperature profiles match the DSMC solution closely. The shock thickness measured from DSMC is 19.1 mean free paths and the DVM shock profile is 17.2 mean free paths thick. In order to compute a Mach 10 shock wave with reasonable efficiency on a uniform velocity grid, coarse velocity discretizations must be used because the velocity space is extremely large. There is significant noise on the upstream side of the shock wave because the velocity spacing is coarse and the upstream distribution is represented by relatively few points. Downstream, the coarse velocity grid is not an issue because as the temperature increases and the distribution function widens.

6.3. Cost comparison

To evaluate the performance, the time required to generate a solution with a certain level of noise is compared for the discrete velocity and DSMC solutions. In all the cases presented above, the DSMC solutions required less time than the dis-

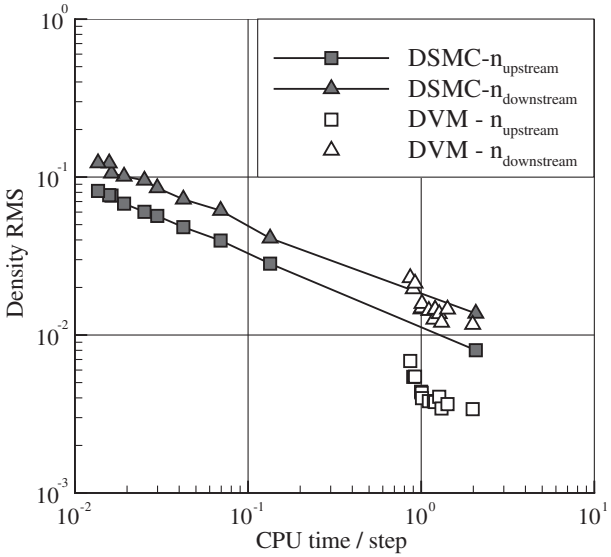

Fig. 12. Noise in the density upstream (squares) and downstream (triangles) of the shock wave plotted against the computational time for discrete velocity and DSMC solutions.

crete velocity model. However, this is not a direct comparison because the DSMC solution was for a steady shock wave and the discrete velocity model was applied to a traveling shock wave. Since the DSMC code generates a steady shock wave, time averaging is used to obtain smooth profiles rapidly, while averaging was not used with the discrete velocity model. For a direct comparison of the computational costs, the discrete velocity model was adapted to a steady shock wave and the averaging procedure in the DSMC code was turned off. Ensemble averaging was not used when characterizing the performance in either the DSMC or the discrete velocity calculations because ensemble averaging is not possible for general unsteady problems.

To generate a steady shock wave the right half of the physical domain was initialized to the upstream flow conditions and the left half of the domain was set to the Rankine–Hugoniot conditions appropriate to the flow downstream of a shock. Constant inflow and outflow boundary conditions were imposed at both the upstream and downstream boundaries. A steady Mach 2 shock wave was used to compare the computational performance of the discrete velocity and DSMC models.

All simulations were performed on a desktop computer with an Intel 2.4 GHz processor, 4 GB of RAM, and a 2 MB cache. The statistical noise in the solution was evaluated by computing the RMS density fluctuations, defined in Eq. (23), in sections downstream and upstream of the shock wave that were at least 20 physical cells away from the boundaries and sufficiently far from the shock.

$$RMS = \sqrt{\frac{1}{N_s}\sum_{i=1}^{N_s}(n_i - \bar{n})^2} \tag{23}$$

In both the DSMC and discrete velocity simulations, 200 physical cells were used with a scaled physical space step $\alpha = 0.5$ and a scaled time step $\Delta\hat{t} = 0.05$. After a steady shock developed, the simulation continued to run for an additional 1500 time steps and the corresponding computational time was recorded. To evaluate the rate of convergence, the DSMC simulation was performed 10 times with various numbers of computational molecules. In each simulation, the noise levels were measured upstream and downstream of the shock wave and the computational time required to solve the entire flow field was recorded. The discrete velocity simulation was executed 14 times with different depleting fractions and two replenishing pairs per collision. A linearly varying depleting fraction was used and this was adjusted by keeping the lower bound fixed at 1% and changing only the upper bound from 5% to 70%. Fig. 12 shows the noise in the density upstream and downstream of the shock wave plotted against the corresponding computational time per time step. The computational time is the time required to solve for the entire flow field, while the corresponding statistical errors are plotted upstream and downstream of the shock wave.

In both the DSMC and DVM computations, the noise is less on the upstream side of the shock than the downstream side. This is because the flow upstream of the shock is supersonic and the noise does not propagate upstream significantly. Downstream of the shock wave the flow is subsonic and fluctuations can propagate upstream. The discrete velocity model performs appreciably better than DSMC upstream of the shock wave. For a smooth upstream solution with noise less than 0.004, the DVM calculation is approximately an order of magnitude faster than DSMC for the same noise level in the solution. Downstream of the shock wave, the differences between DSMC and the DVM are subtle, with only marginal improvements in the performance of the DVM. DSMC performs relatively better downstream of the shock wave than upstream of the shock

wave because the downstream density is higher and consequently more computational molecules are present in the downstream region. This results in more statistical samples for computing the macroscopic properties. However, upstream of the shock wave the density is lower and there are fewer computational molecules to sample from and the statistical uncertainty increases. There are no particles in the discrete velocity model and consequently the model is not affected by a region of lower density upstream of the shock wave.

The cost comparisons above only address the effect of varying the depleting fraction. There are other ways to modify the accuracy of the computation of the collision integral. The number of replenishing pairs, the time step, the physical space discretization, velocity grid interval, and the extent of the velocity space also affect computational performance. The dependence becomes more complicated when a varying depleting fraction is used instead of a fixed depleting fraction. The effects of the many parameters on the computational efficiency have not yet been examined in detail. Our preliminary studies showed that varying the depleting fraction had a relatively large effect on accuracy. In addition, the ability to adjust the level of fidelity when solving the collision integral is a unique capability this method relative to more traditional methods. For these reasons, we chose to focus on the effect of using different depleting fractions in this first study.

7. Conclusions

A discrete velocity model for the Boltzmann equation solver has been developed that can be applied to unsteady flows that bridge the rarefied and continuum regimes. In this regime the flows are non-equilibrium and highly collisional and consequently it is computationally expensive to obtain smooth transient data. The key features of the proposed discrete velocity method are:

- The interpolation algorithm that was developed to allow for arbitrary post-collision velocities.
- The computational cost was reduced by evaluating the collision integral statistically while preserving conservation of mass, momentum, and energy.
- 1D results for traveling shock waves agree well with DSMC solutions over wide ranges of Mach number.
- The performance of the new scheme is comparable to DSMC in the high density region downstream of the shock wave. Quiet solutions in the low density upstream regions can be obtained much more efficiently than with DSMC. The ability to obtain low noise results independent of local gas density is an advantage of this method over DSMC when computing flow fields with large density variations.
- Complete knowledge of the distribution function allows for simple implementation of variance reduction procedures to this method [25].

The interpolation routine allows the discrete velocity scheme to permit post-collision velocities that lie between grid points or entirely outside of the velocity domain. It also allows for the possibility of mapping to different velocity grids. This unique ability addresses a major obstacle for discrete velocity methods which have invariably used a constant step size in velocity space. For example, one could use a coarse velocity grid downstream of a high Mach number shock wave and a fine velocity grid upstream. The mapping scheme also allows the scheme to be applied to polyatomic gas mixtures, because it can readily treat collisions that yield post-collision velocities that do not lie on the local velocity grid.

Appendix A

A.1. Mean free path definition

In continuum dynamics a shock is generally regarded as an abrupt change in macroscopic properties and the ratios between the upstream and downstream properties are described by the Rankine–Hugoniot conditions. In reality shock waves have a finite thickness and the properties inside of the shock wave depend on the nature of the molecular interactions. The thickness of a shock wave is commonly defined as the distance required to span the density change divided by the maximum density gradient. Commonly the shock thickness is expressed as a multiple of the upstream mean free path. The most general definition of the mean free path, λ, is the mean thermal speed divided by the collision rate:

$$\lambda = \frac{\langle c' \rangle}{n \langle \sigma g \rangle} \tag{A1}$$

The mean thermal speed for a gas in equilibrium is

$$\langle c' \rangle = \sqrt{\frac{8 k_b T}{\pi m}} \tag{A2}$$

The VHS collision cross-section is:

$$\sigma = \sigma_{ref} \left(\frac{g^2}{c_{ref}^2} \right)^{-\omega} \tag{A3}$$

The parameter ω describes the variation of the collision cross section with the relative translational energy. For hard sphere molecules, ω is 0 and for pseudo-Maxwell molecules ω is 0.5. The mean collision rate is computed for a gas in equilibrium and is

$$n\langle\sigma g\rangle = 2n\sigma_{ref}c_{ref}^{2\omega}\left(\frac{2k_bT}{\pi m^*}\right)^{1/2}\left(\frac{m^*}{2k_bT}\right)^{\omega}\Gamma(2-\omega) \tag{A4}$$

Here m^* is the reduced mass in a binary collision, ($m^* = m/2$ for a single species gas). Substituting the expressions for the mean thermal speed and collision rate into Eq. (A1) the upstream mean free path is:

$$\lambda_{upstream} = \frac{\widehat{T}_{upstream}^{\omega}}{2^{\left(\frac{1}{2}-\omega\right)}\Gamma(2-\omega)\hat{n}_{upstream}n_r\sigma_{ref}\left(\frac{c_{ref}}{\eta_r}\right)^{2\omega}} \tag{A5}$$

In our scheme the reference thermal speed c_{ref} is the same reference speed η_r. Using that scaling, the mean free path in our scheme is

$$\lambda_{upstream} = \frac{\widehat{T}_{upstream}^{\omega}}{2^{\left(\frac{1}{2}-\omega\right)}\Gamma(2-\omega)\hat{n}_{upstream}n_r\sigma_r} \tag{A6}$$

In Eq. (A6), the cross section σ_r is a reference collision cross section when η_r is used as the reference speed, c_{ref}. This reference cross section changes if a different reference speed is used. An alternative definition for the mean free path for a variable hard sphere gas is presented by Bird in [19,20] and is given by:

$$\lambda_{upstream} = \frac{\widehat{T}_{upstream}^{\omega}}{\sqrt{2}(2-\omega)^{\omega}\Gamma(2-\omega)\hat{n}_{upstream}n_r\sigma_{ref}} \tag{A7}$$

Bird's mean free path definition, Eq. (A7), appears different than that of Eq. (A6) because Bird defines a different reference thermal speed via

$$c_{ref}^2 = 2(2-\omega)k_bT_r/m^* \tag{A8}$$

As a consequence of using difference reference speeds, σ_r in Eq. (A6) is not the same as σ_{ref} in Eq. (A7), but one can show that Eq. (A6) and (A7) are identical if the same reference cross section is used. From Eq. (A3), one can relate σ_r to σ_{ref} via

$$\sigma_r = [2(2-\omega)]^{\omega}\sigma_{ref} \tag{A9}$$

By substituting Eq. (A9) into Eq. (A6) it can easily be shown that the two mean free path definitions are equivalent.

Appendix B

The process of selecting random collision partners can be made more efficient than simple acceptance–rejection by creating a cumulative distribution function Ψ.

$$\Psi(l) = \sum_{0}^{l}\left|\hat{\varphi}_{IJK}\right| \tag{B1}$$

The function $\Psi(s)$ increases monotonically from $l = 0$ to $l = (N_V - 1)$. A uniformly distributed random number R between 0 and $\Psi(N_V - 1)$ is drawn. A binary search is then used to find the value l such that R lies between $\Psi(l)$ and $\Psi(l+1)$. Since the

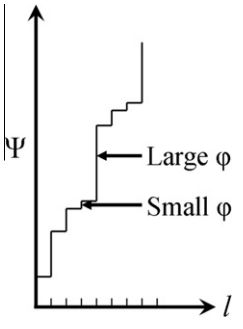

Fig. B1. A sample plot of the cumulative distribution function Ψ. Large steps correspond to large values of φ and smaller steps correspond to smaller values of φ.

increment between $\Psi(l+1)$ and $\Psi(l)$ is $|\varphi_l|$, points in velocity space with higher density are more likely to be chosen than points with lower density, as shown in Fig. B1.

References

[1] G.A. Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, Clarendon Press, Oxford, 1994.
[2] A. Nordsieck, B.L. Hicks, Monte Carlo evaluation of the Boltzmann collision integral, in: C.L. Brundin (Ed.), Proceedings of 5th International Symposium on Rarefied Gas Dynamics, Academic Press, New York, 1967, pp. 695–710.
[3] S.M. Yen, Numerical solution of the nonlinear Boltzmann equation for non-equilibrium gas flow problems, Annu. Rev. Fluid Mech 16 (1984) 67.
[4] F.G. Tcheremissine, A method for direct numerical integration of Boltzmann equation, in: V.P. Shidlovskii (Ed.), Numerical Method in the Theory of Gases, Comput Cent., USSR Acad. Sci., Moscow, 1969, pp. 43–63. Transl. in NASA TT F-638.
[5] Z. Tan, P.L. Varghese, The $\Delta - \varepsilon$ method for the Boltzmann equation, J. Comput. Phys. 110 (1994) 327.
[6] T. Inamuro, B. Sturtevant, Numerical study of discrete-velocity gases, Phys. Fluids A 2 (1990) 2196–2203.
[7] Z. Tan, Y.K. Chen, P.L. Varghese, J.L. Howell, new numerical strategy to evaluate the collision integral of the Boltzmann equation, in: E.P. Muntz (Ed.), Rarefied Gas Dynamics: Theoretical and Computational Techniques, AIAA, Washington, DC, 1989, p. 359.
[8] P.L. Varghese, Arbitrary post-collision velocities in a discrete velocity scheme for the Boltzmann equation, in: M.S. Ivanov, A.K. Rebrov (Eds.), Rarefied Gas Dynamics: Proceedings of the 25th International Symposium, Novosibirsk, Russia, 2007, pp. 227–232.
[9] A.B. Morris, P.L. Varghese, D.G. Goldstein, Improvement of a discrete velocity Boltzmann equation solver that allows for arbitrary post-collision velocities, in: Rarefied Gas Dynamics: Proceedings of the 26th International Symposium, Kyoto, Japan, 2008.
[10] F.G. Tcheremissine, Direct numerical solution of the Boltzmann equation, in: M. Capitelli (Ed.), Rarefied Gas Dynamics, AIP, Conference Proceedings, NY, 762, 2005, pp. 677–685.
[11] S.P. Popov, F.G. Tcheremissine, A conservative method for solving the Boltzmann equation with centrally symmetric interaction potentials, Comput. Math. Math. Phys. 39 (1999) 156.
[12] F.G. Tcheremissine, Solution to the Boltzmann kinetic equation for high-speed flows, Comput. Math. Math. Phys. 46 (2006) 315.
[13] V.I. Kolobov, R.R. Arslanbekov, V.V. Aristov, A.A. Frolova, S.A. Zabelokm, Unified solver for rarefied and continuum flows with adaptive mesh algorithm refinement, J. Comput. Phys. 223 (2007) 589–608.
[14] V.V. Aristov, F.G. Tcheremissine, Shock wave structure in monatomic gas at power potentials, Fluid Dyn. 2 (1982) 179–183.
[15] T. Ohwada, Structure of normal shock waves on the basis of the Boltzmann equation for hard-sphere molecules, Phys. Fluids A5 (1993) 217–234.
[16] J.E. Lennard-Jones, Cohesion, Proc. Phys. Soc. 43 (1931) 461–482.
[17] G.A. Bird, Monte-Carlo simulation in an engineering context, Prog. Astro. Aero. 74 (1981) 239–255.
[18] K. Koura, H. Matsumoto, Variable soft sphere molecular model for inverse-power-law or Lennard-Jones potential, Phys. Fluids A 3 (1991) 2459–2465.
[19] G.A. Bird, Definition of mean free path for real gases, Phys. Fluids 26 (11) (1983).
[20] G.A. Bird, Monte-Carlo simulation, in: An Engineering Context, Rarefied Gas Dynamics: Proceedings of the 12th International Symposium, July 7th–11th, Charlottesville, Va, 1980.
[21] A. Morris, Investigation of a Discrete Velocity Monte Carlo Boltzmann Equation Solver, The University of Texas at Austin, Austin, TX, 2009.
[22] D. Goldstein, Investigations of a Discrete Gas, California Institute of Technology, Pasadena, CA, 1990.
[23] W. Wagner, A convergence proof for bird's direct simulation Monte Carlo method for the Boltzmann equation, J. Stat. Phys. 66 (1992) 1011.
[24] I. Boyd, Temperature dependence of rotational relaxation in shock waves of nitrogen, J. Fluid Mech. 243 (1993) 343.
[25] A. Morris, P. Varghese, D. Goldstein, Variance reduction for a discrete velocity gas, in: Rarefied Gas Dynamics: Proceedings of the 27th International Symposium, Pacific Grove, USA, 2010.