

A Modified BGK Collision Operator for Exact Conservation in Numerical Solutions of Boltzmann-BGK

Vienna B. Rossmannith[†]

Project advisor: Preeti Sar[‡]

Abstract. Ideal gases can be modeled by the Boltzmann equation from statistical physics. Instead of trying to track the position and velocity of a large number of gas molecules, it is possible to describe the particles with a particle distribution function. The Boltzmann equation provides the rule for evolving the distribution function over time, allowing one to simulate the gas dynamics. In this work, we develop a novel numerical method for solving the 1D1V Boltzmann-BGK equation. Several important ingredients are combined to create an accurate, efficient, robust numerical method valid in all flow regimes. First, we make use of operator splitting to create separate transport and collision sub-steps, each of which is easier to discretize than the whole system; second, we introduce a third-order accurate Lax-Wendroff-type scheme for the transport sub-step; third, we make use of the second-order L-stable TR-BDF method for the collision sub-step. The final component we introduce is a novel treatment of the collision sub-step to guarantee that the total mass, momentum, and energy are conserved even in the presence of a truncated velocity range and quadrature errors in computing moments. The key idea is to multiply the Maxwell-Boltzmann distribution in the collision sub-step by a quadratic Hermite polynomial, where the coefficients in the polynomial are chosen to ensure exact conservation. The resulting scheme is verified on piecewise constant initial data with periodic boundary conditions; exact conservation up to machine precision is demonstrated for mass, momentum, and energy. The resulting method is implemented in a freely available Java code with Python plotting routines.

1. Introduction. A gas is a collection of unbound molecules; in the absence of external forces, each molecule moves with some velocity until it collides with another molecule. The collisions act to drive the gas towards a thermodynamic equilibrium. Mathematically, one could consider a range of models, including (1) molecular dynamics, which is valid at very small scales, where each molecule in some small portion of the gas is tracked; (2) kinetic Boltzmann equations, which is valid at intermediate scales, where a particle distribution function replaces individual molecules; and (3) fluid models such the Navier-Stokes equation, which are valid at large scales, where macroscopic quantities such as mass, momentum, and energy are tracked (e.g., see [14]).

The focus of this work is the kinetic Boltzmann equation; in particular, we consider the 1D1V Boltzmann-BGK equation [2], in which the full Boltzmann collision operator is replaced by a far simpler collision operator that nonetheless captures key dynamics. While there exist several approaches to solving the Boltzmann-BGK equation, we consider in this work so-called *direct kinetic* methods, rather than stochastic methods such as Direct Simulation Monte-Carlo (DSMC) (e.g., see [12]). In a direct kinetic approach, various numerical discretizations can be applied on a computational mesh that covers a space/velocity domain that is truncated in velocity.

The goal of this work is to develop a finite difference method for the one-dimensional

[†]Ames High School, Ames, IA, USA (887432ros@ames.k12.ia.us)

[‡]Fusion Energy Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA

Boltzmann-BGK model of gas dynamics that is similar in spirit to existing approaches (e.g., Pieraccini and Puppo [13], Li et al. [10], and Sar [16]), but with the additional feature that mass, momentum, and energy are exactly conserved. Several important ingredients are combined to create an accurate, efficient, robust numerical method valid in all flow regimes. First, we make use of operator splitting to create separate transport and collision sub-steps, each of which is easier to discretize than the whole system; second, we introduce a third-order accurate Lax-Wendroff-type scheme for the transport sub-step; third, we make use of the second-order L-stable TR-BDF method for the collision sub-step. The final component we introduce is a novel treatment of the collision sub-step to guarantee that the total mass, momentum, and energy are conserved even in a truncated velocity range and quadrature errors in computing moments. The key idea is to multiply the Maxwell-Boltzmann distribution in the collision sub-step by a quadratic Hermite polynomial, where the coefficients in the polynomial are chosen to ensure exact conservation. The proposed approach is similar to existing schemes in the literature that also ensure exact conservation, such as Gradoboev and Rykov [5], Mieussens [11], and Titarev [18]; the main advantage of the new approach is that we are able to write down a simple closed form solution for the coefficients.

The remainder of this paper is organized as follows. In §2 we briefly describe the 1D1V Boltzmann-BGK equation. In §3 we develop the main components of the proposed numerical method using operator splitting and finite differences. The scheme described in §3 does not exactly conserve mass, momentum, and energy. Therefore, in §4 we develop a modified collision operator that is able to recover exact conservation; in this same section, we present the final proposed numerical method in two short algorithms. A numerical example illustrating the benefits of the proposed modification is provided in §5. We conclude in §6.

2. Boltzmann-BGK. A particle distribution function (PDF) of time, position, and velocity can be used to describe the movement of molecules in a gas. In a one-dimensional system, the PDF is a function of the following form:

$$(2.1) \quad f(t, x, v) : \mathbb{R}_{\geq 0} \times \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}_{\geq 0},$$

where $t \in \mathbb{R}_{\geq 0}$ is time, $x \in \mathbb{R}$ is the space coordinate, and $v \in \mathbb{R}$ is the velocity coordinate. The total number of particles that can be found in some spatial region $x \in [x_1, x_2]$ with some range of velocities $v \in [v_1, v_2]$ can be computed by integrating the PDF over these values:

$$(2.2) \quad \text{total number of particles in } [x_1, x_2] \times [v_1, v_2] = \int_{x_1}^{x_2} \int_{v_1}^{v_2} f(t, x, v) \, dv \, dx.$$

According to gas kinetic theory, the PDF satisfies the Boltzmann equation, which in one dimension can be written as follows:

$$(2.3) \quad \frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} = \frac{1}{\varepsilon} C(f),$$

which consists of two key pieces: (1) the motion of particles (i.e., gas molecules) according to their velocity, v , and the collision of particles as modeled by the collision operator, $C(f)$. The parameter $\varepsilon > 0$ is called the Knudsen number and is defined as the ratio of the mean-free path (i.e., how far a particle travels on average before undergoing a collision) to an important

characteristic length scale in the problem (i.e., size of the domain). Small $\varepsilon > 0$ means high collisionality, while large $\varepsilon > 0$ means low collisionality.

The net effect of the collision operator on the dynamics is that it drives the gas towards a *thermodynamic equilibrium*, which in one dimension is given by the following distribution:

$$(2.4) \quad \mathcal{M}(t, x, v) = \frac{\rho(t, x)}{\sqrt{2\pi T(t, x)}} \exp \left[-\frac{(v - u(t, x))^2}{2T(t, x)} \right];$$

this distribution is referred to as the *Maxwell-Boltzmann* distribution and depends on three macroscopic (or fluid) variables:

$$(2.5) \quad \text{Mass density: } \rho(t, x) = \int_{-\infty}^{\infty} f(t, x, v) dv = \int_{-\infty}^{\infty} \mathcal{M}(t, x, v) dv,$$

$$(2.6) \quad \text{Fluid velocity: } u(t, x) = \frac{1}{\rho(t, x)} \int_{-\infty}^{\infty} v f(t, x, v) dv = \frac{1}{\rho(t, x)} \int_{-\infty}^{\infty} v \mathcal{M}(t, x, v) dv,$$

$$(2.7) \quad \begin{aligned} \text{Temperature: } T(t, x) &= \frac{1}{\rho(t, x)} \int_{-\infty}^{\infty} (v - u(t, x))^2 f(t, x, v) dv \\ &= \frac{1}{\rho(t, x)} \int_{-\infty}^{\infty} (v - u(t, x))^2 \mathcal{M}(t, x, v) dv. \end{aligned}$$

These variables are also related to two other important quantities:

$$(2.8) \quad \text{Momentum: } m(t, x) = \int_{-\infty}^{\infty} v f(t, x, v) dv = \int_{-\infty}^{\infty} v \mathcal{M}(t, x, v) dv,$$

$$(2.9) \quad \text{Energy: } \mathcal{E}(t, x) = \int_{-\infty}^{\infty} v^2 f(t, x, v) dv = \int_{-\infty}^{\infty} v^2 \mathcal{M}(t, x, v) dv,$$

where

$$(2.10) \quad m = \rho u \quad \text{and} \quad \mathcal{E} = \rho (u^2 + T).$$

One difficulty in solving equation (2.3) is that the collision operator, $C(f)$, can be quite complicated and involve computationally expensive integration over phase space. To reduce this complexity, a standard approach is to replace the full Boltzmann operator by something simpler; a commonly used model is the Bhatnagar, Gross, and Krook (BGK) [2] approximation, which mimics the net effect of the Boltzmann collision operator, namely driving the distribution function, f , towards its thermodynamic equilibrium: (2.4). The Boltzmann-BGK equation can be written as follows:

$$(2.11) \quad f_{,t} + v f_{,x} = \frac{1}{\varepsilon} C(f) = \frac{1}{\varepsilon} (\mathcal{M} - f).$$

By construction, the mass (2.5), momentum (2.8), and energy (2.9) are invariants of the BGK

collision operator:

$$(2.12) \quad \text{Mass: } \frac{1}{\varepsilon} \int_{-\infty}^{\infty} C(f) dv = \frac{1}{\varepsilon} \int_{-\infty}^{\infty} (\mathcal{M} - f) dv = \frac{1}{\varepsilon} (\rho - \rho) = 0,$$

$$(2.13) \quad \text{Momentum: } \frac{1}{\varepsilon} \int_{-\infty}^{\infty} v C(f) dv = \frac{1}{\varepsilon} \int_{-\infty}^{\infty} v (\mathcal{M} - f) dv = \frac{1}{\varepsilon} (\rho u - \rho u) = 0,$$

$$(2.14) \quad \text{Energy: } \frac{1}{\varepsilon} \int_{-\infty}^{\infty} v^2 C(f) dv = \frac{1}{\varepsilon} \int_{-\infty}^{\infty} v^2 (\mathcal{M} - f) dv = \frac{1}{\varepsilon} (\mathcal{E} - \mathcal{E}) = 0.$$

Note that the fluid velocity (2.6) and temperature (2.7) are also invariants of the collision operator.

3. Numerical Method. To approximate the particle distribution function, we consider a domain with finite extent in both x and v :

$$(3.1) \quad (x, v) \in [a, b] \times [v_{\min}, v_{\max}].$$

The finite extent in v may lead to errors in f (i.e., the Maxwell-Boltzmann distribution (2.4) only becomes zero as $|v| \rightarrow \infty$), but is necessary for a mesh-based approach. We then introduce a mesh with uniform rectangles of size Δx by Δv , centered at (x_i, v_j) :

$$(3.2) \quad x_i = a + \left(i - \frac{1}{2}\right) \Delta x, \quad v_j = v_{\min} + \left(j - \frac{1}{2}\right) \Delta v,$$

for $i = 1, \dots, N_x$ and $j = 1, \dots, N_v$, with grid spacings given by

$$(3.3) \quad \Delta x = \frac{b - a}{N_x}, \quad \Delta v = \frac{v_{\max} - v_{\min}}{N_v}.$$

Note that $N_x \times N_v$ is the total number of grid cells.

On this mesh, we begin with the following initial condition at $t = 0$:

$$(3.4) \quad f_{ij}^0 := f(t = 0, x_i, v_j) \quad \text{for } i = 1, \dots, N_x \quad \text{and} \quad j = 1, \dots, N_v.$$

We aim to advance the initial distribution from $t = 0$ to some final time $t = T_{\text{final}}$. We will advance the initial f^0 at $t = 0$ to f^1 at $t = \Delta t$, then advance f^1 at $t = \Delta t$ to f^2 at $t = 2\Delta t$, and so on, until some final value f^N at time $t = N\Delta t$, where N is the total number of time steps with a time step of $\Delta t = T_{\text{final}}/N$.

Recall that the Boltzmann-BGK equation can be written as (2.11). To simplify the system, we first apply Strang operator splitting [17] to split a single time step of length Δt of the numerical method into transport and collision parts:

$$(3.5) \quad \begin{cases} \frac{\Delta t}{2} : & \frac{\partial f}{\partial t} = \frac{1}{\varepsilon} (\mathcal{M} - f), & // \text{ Half time-step of collisions only} \\ \Delta t : & \frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} = 0, & // \text{ Full time-step of transport only} \\ \frac{\Delta t}{2} : & \frac{\partial f}{\partial t} = \frac{1}{\varepsilon} (\mathcal{M} - f). & // \text{ Half time-step of collisions only} \end{cases}$$

We describe the numerical discretizations used for both the transport and collision sub-steps below.

3.1. Transport Sub-Step. We discretize the transport sub-step,

$$(3.6) \quad \frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} = 0,$$

by a third-order variant of the Lax-Wendroff method [9]. Consider the following Taylor series in time, comparing the PDF at some time $t = t^n$ to the PDF at the time $t = t^n + \Delta t$:

$$(3.7) \quad f(t^n + \Delta t, x, v) = \left(f + \Delta t \frac{\partial f}{\partial t} + \frac{\Delta t^2}{2} \frac{\partial^2 f}{\partial t^2} + \frac{\Delta t^3}{6} \frac{\partial^3 f}{\partial t^3} \right) \Big|_{(t^n, x, v)} + \mathcal{O}(\Delta t^4).$$

Each time derivative can then be replaced by a spatial derivative via the transport equation:

$$(3.8) \quad \frac{\partial f}{\partial t} = -v \frac{\partial f}{\partial x}, \quad \frac{\partial^2 f}{\partial t^2} = -v \frac{\partial}{\partial x} \left(\frac{\partial f}{\partial t} \right) = v^2 \frac{\partial^2 f}{\partial x^2}, \quad \frac{\partial^3 f}{\partial t^3} = v^2 \frac{\partial^2}{\partial x^2} \left(\frac{\partial f}{\partial t} \right) = -v^3 \frac{\partial^3 f}{\partial x^3}.$$

Plugging this into (3.7) and evaluating at the cell center (x_i, v_j) yields:

$$(3.9) \quad f(t^n + \Delta t, x_i, v_j) \approx \left(f - v \Delta t \frac{\partial f}{\partial x} + \frac{v^2 \Delta t^2}{2} \frac{\partial^2 f}{\partial x^2} - \frac{v^3 \Delta t^3}{6} \frac{\partial^3 f}{\partial x^3} \right) \Big|_{(t^n, x_i, v_j)} + \mathcal{O}(\Delta t^4).$$

Finally, all spatial derivatives are replaced by finite differences. For all negative velocity values ($v_j < 0$), we compute the first, second, and third spatial derivatives at x_i using the highest-order approximation possible using the points $(x_{i-1}, x_i, x_{i+1}, x_{i+2})$. For all positive velocity values ($v_j > 0$), we compute the first, second, and third spatial derivatives at x_i using the highest-order approximation possible using the points $(x_{i-2}, x_{i-1}, x_i, x_{i+1})$. This yields the following:

$$(3.10) \quad \frac{\partial f}{\partial x} \Big|_{(x_i, v_j)} = D_x^{(1)} f_{ij} + \mathcal{O}(\Delta x^3), \quad D_x^{(1)} f_{ij} := \begin{cases} \frac{-2f_{i-1j} - 3f_{ij} + 6f_{i+1j} - f_{i+2j}}{6\Delta x} & v_j < 0, \\ \frac{f_{i-2j} - 6f_{i-1j} + 3f_{ij} + 2f_{i+1j}}{6\Delta x} & v_j > 0, \end{cases}$$

$$(3.11) \quad \frac{\partial^2 f}{\partial x^2} \Big|_{(x_i, v_j)} = D_x^{(2)} f_{ij} + \mathcal{O}(\Delta x^2), \quad D_x^{(2)} f_{ij} := \frac{f_{i-1j} - 2f_{ij} + f_{i+1j}}{\Delta x^2},$$

$$(3.12) \quad \frac{\partial^3 f}{\partial x^3} \Big|_{(x_i, v_j)} = D_x^{(3)} f_{ij} + \mathcal{O}(\Delta x), \quad D_x^{(3)} f_{ij} := \begin{cases} \frac{-f_{i-1j} + 3f_{ij} - 3f_{i+1j} + f_{i+2j}}{\Delta x^3} & v_j < 0, \\ \frac{-f_{i-2j} + 3f_{i-1j} - 3f_{ij} + f_{i+1j}}{\Delta x^3} & v_j > 0. \end{cases}$$

Plugging these approximations into (3.9) results in the following:

$$f(t^n + \Delta t, x_i, v_j) \approx \left(f - v \Delta t D_x^{(1)} f + \frac{v^2 \Delta t^2}{2} D_x^{(2)} f - \frac{v^3 \Delta t^3}{6} D_x^{(3)} f \right) \Big|_{ij}^n + \mathcal{O}(\Delta t \Delta x^3 + \Delta t^2 \Delta x^2 + \Delta t^3 \Delta x + \Delta t^4).$$

For stability and accuracy reasons, we take $\Delta t = \mathcal{O}(\Delta x)$, which means that the so-called 1-step error is $\mathcal{O}(\Delta x^4)$, meaning that the truncation error over a fixed time interval $[0, T]$ is $\mathcal{O}(\Delta x^3)$.

The final update formula can be summarized as follows:

$$(3.13) \quad \begin{aligned} \nu_j < 0 : \quad f_{ij} = f_{ij}^* - \frac{\nu_j}{6} \left(-2f_{i-1j}^* - 3f_{ij}^* + 6f_{i+1j}^* - f_{i+2j}^* \right) \\ + \frac{\nu_j^2}{2} \left(f_{i-1j}^* - 2f_{ij}^* + f_{i+1j}^* \right) - \frac{\nu_j^3}{6} \left(-f_{i-1j}^* + 3f_{ij}^* - 3f_{i+1j}^* + f_{i+2j}^* \right), \end{aligned}$$

$$(3.14) \quad \begin{aligned} \nu_j > 0 : \quad f_{ij} = f_{ij}^* - \frac{\nu_j}{6} \left(f_{i-2j}^* - 6f_{i-1j}^* + 3f_{ij}^* + 2f_{i+1j}^* \right) \\ + \frac{\nu_j^2}{2} \left(f_{i-1j}^* - 2f_{ij}^* + f_{i+1j}^* \right) - \frac{\nu_j^3}{6} \left(-f_{i-2j}^* + 3f_{i-1j}^* - 3f_{ij}^* + f_{i+1j}^* \right), \end{aligned}$$

where $\nu_j = v_j \Delta t / \Delta x$, and in the above formulas, f^* represents the old value of the distribution (i.e., at time $t = t^n$), while f represents the new value (i.e., at time $t = t^n + \Delta t$).

3.2. Collision Sub-Step. The collision sub-step requires us to approximately solve

$$(3.15) \quad \frac{\partial f}{\partial t} = \frac{1}{\varepsilon} (\mathcal{M} - f).$$

We first need to be able to compute the discrete Maxwell-Boltzmann distribution (2.4), which first requires the computation of the mass density (2.5), fluid velocity (2.6), and temperature (2.7). At the discrete level, we accomplish this via simple Riemann sums:

$$(3.16) \quad \begin{pmatrix} \rho_i \\ m_i \\ \mathcal{E}_i \end{pmatrix} = \Delta v \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} f_{ij}, \quad u_i = \frac{m_i}{\rho_i}, \quad T_i = \frac{\mathcal{E}_i}{\rho_i} - (u_i)^2,$$

and then evaluate the Maxwell-Boltzmann distribution at each cell center:

$$(3.17) \quad \mathcal{M}_{ij} = \frac{\rho_i}{\sqrt{2\pi T_i}} \exp \left[-\frac{\mu_{ij}^2}{2} \right], \quad \text{where} \quad \mu_{ij} = \frac{v_j - u_i}{\sqrt{T_i}}.$$

To get stable and accurate solutions from the collision sub-step, we make use of the L-stable TR-BDF2 scheme [1]. The L-stable property allows for the resulting numerical method to be stable for fixed Δt , Δx , and Δv for any Knudsen number, $\varepsilon > 0$, including in the singular limit: $\varepsilon \rightarrow 0^+$. This makes the overall scheme *asymptotic-preserving* (AP) [3, 7, 8]. For a generic ordinary differential equation of the form: $u' = g(u)$, the TR-BDF2 scheme is a 2-stage implicit Runge-Kutta method that can be written as follows:

$$(3.18) \quad U^* = U^n + \frac{\Delta t}{4} (g(U^n) + g(U^*)), \quad 3U^{n+1} - 4U^* + U^n = \Delta t g(U^{n+1}).$$

The collision sub-system over a single time-step from $t = t^n$ to $t = t^n + \Delta t$ for the Boltzmann-BGK equation will have the following form:

$$(3.19) \quad \frac{\partial f}{\partial t} = \frac{1}{\varepsilon} (\mathcal{M}^n - f),$$

where we note that over the time step, $t \in [t^n, t^n + \Delta t]$, the Maxwell-Boltzmann distribution is constant in time: $\mathcal{M}^n$. Why is it constant in time? Because, as already shown above, the mass (2.12), momentum (2.13), and energy (2.14), are all collision invariants, which also means the Maxwell-Boltzmann distribution is a collision invariant. With a fixed $\mathcal{M}^n$, it is quite simple to apply the TR-BDF2 scheme (3.18) to (3.19):

$$(3.20) \quad f_{ij}^* = f_{ij}^n + \frac{\Delta t}{4\varepsilon} (2\mathcal{M}_{ij}^n - f_{ij}^n - f_{ij}^*), \quad 3f_{ij}^{n+1} - 4f_{ij}^* + f_{ij}^n = \frac{\Delta t}{\varepsilon} (\mathcal{M}_{ij}^n - f_{ij}^{n+1}).$$

After some algebra (i.e., first solve for f^* in terms of f^n and $\mathcal{M}^n$ in the first stage, then use this result to solve for f^{n+1} in terms of f^n and $\mathcal{M}^n$ in the second stage), this can be reduced to the following update:

$$(3.21) \quad f_{ij}^{n+1} = \theta \mathcal{M}_{ij}^n + (1 - \theta) f_{ij}^n, \quad \text{where} \quad \theta = \frac{\Delta t (\Delta t + 12\varepsilon)}{(\Delta t + 3\varepsilon)(\Delta t + 4\varepsilon)}.$$

Note that in the current discussion, we are using a time step of size Δt , while in the splitting paradigm of (3.5), we take half-time steps of the collision sub-step. To take a half-time step, we just need to replace Δt in (3.21) with $\Delta t/2$.

4. Modified Collision Operator. As described above in (2.12), (2.13), and (2.14), the mass, momentum, and energy are invariants of the collision operator. At the discrete level, this is only approximately true [11, 13]. Consider the following collision sub-step as described above:

$$(4.1) \quad f_{ij}^{n+1} = \theta \mathcal{M}_{ij}^n + (1 - \theta) f_{ij}^n,$$

and compute the discrete mass, momentum, and energy via (3.16):

$$(4.2) \quad \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} f_{ij}^{n+1} = \theta \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} \mathcal{M}_{ij}^n + (1 - \theta) \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} f_{ij}^n.$$

In general, mass, momentum, and energy will not be conserved:

$$(4.3) \quad \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} f_{ij}^{n+1} \neq \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} f_{ij}^n,$$

because the distribution, f , and the Maxwell-Boltzmann distribution, $\mathcal{M}$, do not have exactly matching discrete moments:

$$(4.4) \quad \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} f_{ij}^n \neq \sum_{j=1}^{N_v} \begin{pmatrix} 1 \\ v_j \\ v_j^2 \end{pmatrix} \mathcal{M}_{ij}^n.$$

There are two reasons that the moments in (4.4) do not match:

1. The velocity extent is limited to the domain $v \in [v_{\min}, v_{\max}]$, but the Maxwell-Boltzmann distribution satisfies $\mathcal{M} > 0$ for all $v \in (-\infty, \infty)$, which means that any portion of the distribution that contributes to the total mass, momentum, or energy for $v > v_{\max}$ or $v < v_{\min}$ is lost to the numerical method; and
2. The numerical quadrature is only approximate and cannot exactly integrate the Maxwell-Boltzmann distribution on a finite mesh.

We propose to overcome limitation (4.4) by replacing the discrete Maxwell-Boltzmann distribution, $\mathcal{M}_{ij}$, by a modified version, $\widetilde{\mathcal{M}}_{ij}$, that will allow us to recover exact conservation. The idea is similar in spirit to existing work in the literature: Gradoboev and Rykov [5], Mieussens [11], and Titarev [18]. The proposed approach is similar to Gradoboev and Rykov [5] in that we multiply the Maxwell-Boltzmann equation by a polynomial that closely approximates the number one, but with unknown coefficients that are used to get exact conservation. However, the details of the underlying numerical scheme and how we compute our coefficients are quite different compared to [5]; a key advantage of our approach is that we obtain a simplified closed form solution for the unknown coefficients. Mieussens [11] puts the unknown coefficients in the exponent of an exponential, which results in a nonlinear system for the coefficients. Titarev [18] modifies the macroscopic variables through a projection that guarantees conservation; this again requires the solution of a nonlinear system. The main advantage of the new approach is that it only requires the solution of a linear system, and the solution can be written down in closed form.

We consider a modified Maxwell-Boltzmann distribution of the form:

$$(4.5) \quad \widetilde{\mathcal{M}}_{ij} = \frac{\rho_i}{\sqrt{2\pi T_i}} \exp \left[-\frac{\mu_{ij}^2}{2} \right] \left(a_{1i} + a_{2i} \mu_{ij} + a_{3i} (\mu_{ij}^2 - 1) \right) \quad \text{where} \quad \mu_{ij} = \frac{v_j - u_i}{\sqrt{T_i}}.$$

This form also borrows from the classical Grad moment-closure [4] in that

$$(4.6) \quad \{1, \mu, \mu^2 - 1\}$$

are the first three Hermite polynomial basis functions (i.e., orthogonal polynomials on $\mu \in (-\infty, \infty)$ with weight $\exp(-\mu^2/2)$). If there were no discrete errors in the numerical integration, the coefficients would be $a_{1i} = 1$, $a_{2i} = 0$, and $a_{3i} = 0$, which would recover the original Maxwell-Boltzmann distribution, $\mathcal{M}_{ij}$.

To compute the unknown coefficients, a_{1i} , a_{2i} , and a_{3i} , we take the modified collision sub-step update:

$$(4.7) \quad f_{ij} \leftarrow \theta \widetilde{\mathcal{M}}_{ij} + (1 - \theta) f_{ij},$$

and require that $\widetilde{\mathcal{M}}$ and f have the same *discrete* mass, momentum, and energy moments:

$$(4.8) \quad \frac{\Delta v}{\sqrt{2\pi T_i}} \sum_{j=1}^{N_v} \left(\frac{1}{\mu_{ij}^2} \right) \exp \left[-\frac{\mu_{ij}^2}{2} \right] \left(a_{1i} + a_{2i} \mu_{ij} + a_{3i} (\mu_{ij}^2 - 1) \right) = \frac{\Delta v}{\rho_i} \sum_{j=1}^{N_v} \left(\frac{1}{\mu_{ij}^2} \right) f_{ij},$$

Algorithm 4.1 Initialize computation

INPUT: $N_x, N_v, x_{\text{low}}, x_{\text{high}}, v_{\text{low}}, v_{\text{high}}, f_0(x, v)$ (PDF initial condition function)

OUTPUT: f_{ij} (PDF at initial time), $x_i, v_j, \Delta x, \Delta v, v_{\text{max}}$

```
 $\Delta x = (x_{\text{high}} - x_{\text{low}}) / N_x$            // Grid spacing
 $\Delta v = (v_{\text{high}} - v_{\text{low}}) / N_v$        // Velocity grid spacing
 $v_{\text{max}} = \max \{ |v_{\text{low}}|, |v_{\text{high}}| \}$  // Maximum particle velocity

for  $i = 1, \dots, N_x$  do
     $x_i = x_{\text{low}} + (i - 0.5) \Delta x$  // x cell center
end for

for  $j = 1, \dots, N_v$  do
     $v_j = v_{\text{low}} + (j - 0.5) \Delta v$  // v cell center
end for

for  $i = 1, \dots, N_x$  do
    for  $j = 1, \dots, N_v$  do
         $f_{ij} = f_0(x_i, v_j)$  // initial PDF at cell centers
    end for
end for
```

which we can write as a 3×3 linear system:

$$(4.9) \quad \begin{bmatrix} A_{0i} & A_{1i} & A_{2i} - A_{0i} \\ A_{1i} & A_{2i} & A_{3i} - A_{1i} \\ A_{2i} & A_{3i} & A_{4i} - A_{2i} \end{bmatrix} \begin{bmatrix} a_{1i} \\ a_{2i} \\ a_{3i} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix},$$

where

$$(4.10) \quad A_{ki} = \frac{\Delta v}{\sqrt{2\pi T_i}} \sum_{j=1}^{N_v} \mu_{ij}^k \exp \left[-\frac{\mu_{ij}^2}{2} \right] \quad \text{for } k = 0, 1, 2, 3, 4.$$

This 3×3 linear system can be easily solved exactly:

$$(4.11) \quad \begin{bmatrix} a_{1i} \\ a_{2i} \\ a_{3i} \end{bmatrix} = \frac{1}{d_i} \begin{bmatrix} A_{1i}^2 + A_{2i}(2A_{2i} - A_{0i} - A_{4i}) - A_{3i}(2A_{1i} - A_{3i}) \\ A_{1i}(A_{4i} - A_{2i}) + A_{3i}(A_{0i} - A_{2i}) \\ A_{1i}(A_{1i} - A_{3i}) + A_{2i}(A_{2i} - A_{0i}) \end{bmatrix},$$

where

$$(4.12) \quad d_i = A_{2i}^3 - 2A_{1i}A_{2i}A_{3i} + A_{0i}A_{3i}^2 + A_{1i}^2A_{4i} - A_{0i}A_{2i}A_{4i}.$$

We now have a suitable modified Maxwell-Boltzmann distribution, $\widetilde{\mathcal{M}}_{ij}$, as defined in (4.5).

Algorithm 4.2 Advance PDF and moments from initial condition to final time

INPUT: f_{ij} (initial PDF), v_j , Δx , Δv , N_x , N_v , CFL, $v_{\max}$, ε , finalTime

OUTPUT: f_{ij} (PDF at final time)

```
 $\Delta t = \Delta x \text{CFL} / v_{\max}$            // Set time step
nSteps =  $\lceil \text{finalTime} / \Delta t \rceil$    // Compute number of time steps
 $\Delta t = \text{finalTime} / \text{nSteps}$       // Reset time step
 $\text{CFL} = v_{\max} \Delta t / \Delta x$       // Reset CFL number
 $\theta = \frac{\Delta t(\Delta t + 48\varepsilon)}{(\Delta t + 6\varepsilon)(\Delta t + 8\varepsilon)}$  // Set collision parameter

for  $n = 1, \dots, \text{nSteps}$  do
  Step 1: From  $f_{ij}$  compute moments via (3.16) and  $a_{1i}, a_{2i}, a_{3i}$  via (4.10)–(4.12)
  Step 2:  $\Delta t/2$  collision step:  $f_{ij} \leftarrow \theta \widetilde{\mathcal{M}}_{ij} + (1 - \theta) f_{ij}$  using (4.5)
  Step 3: Copy  $f_{ij}$  into a temporary array:  $f_{ij}^* \leftarrow f_{ij}$ 
  Step 4:  $\Delta t$  transport step: compute new  $f_{ij}$  from old  $f_{ij}^*$  using (3.13) and (3.14)
  Step 5: From  $f_{ij}$  compute moments via (3.16) and  $a_{1i}, a_{2i}, a_{3i}$  via (4.10)–(4.12)
  Step 6:  $\Delta t/2$  collision step:  $f_{ij} \leftarrow \theta \widetilde{\mathcal{M}}_{ij} + (1 - \theta) f_{ij}$  using (4.5)
end for
```

We note that the coefficient matrix in (4.9) is an approximation of the matrix:

$$(4.13) \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix},$$

which is fully invertible and has a very small condition number (≈ 2.618). The error in the coefficient matrix in (4.9) relative to the above exact matrix is $\mathcal{O}(\Delta v^2)$ plus a small error due to the truncation of the velocity range. Therefore, in practice, as long as Δv is sufficiently small and velocity range truncation is not too severe, the matrix in (4.9) should be invertible.

To finalize the description of the proposed numerical method for the 1D1V Boltzmann-BGK equation (2.11), we summarize the scheme in two short chunks of pseudocode. In Algorithm 4.1, we show how to initialize the mesh and the distribution function. In Algorithm 4.2, we summarize the complete time-stepping procedure, including how the operator splitting can be implemented. We implement the above-described algorithm in a freely available code [15]; the central portion of the code is written in Java, and the plotting routines are written in Python using NumPy and Matplotlib [6].

5. Numerical Results. To demonstrate the effectiveness of the modified collision operator, we consider a test case with the following initial data:

$$(5.1) \quad f(t=0, x, v) = \mathcal{M}_0(x, v) = \frac{\rho_0(x)}{\sqrt{2\pi T_0(x)}} \exp \left[-\frac{(v - u_0(x))^2}{2T_0(x)} \right],$$

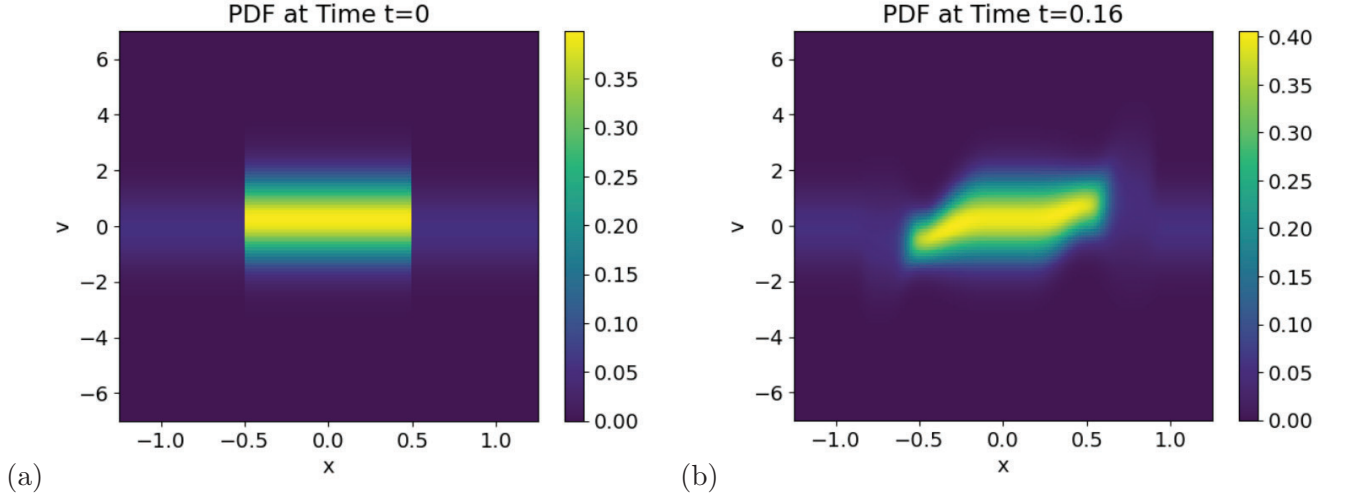


Figure 1. Particle distribution function in the (x, v) phase space at times (a) $t = 0$ and (b) $t = 0.16$. We used the mesh and parameter values described in §5.

where

$$(5.2) \quad (\rho_0, u_0, T_0)(x) = \begin{cases} (1.000, 0.250, 1.000) & \text{if } |x| < 0.5, \\ (0.125, -0.10, 0.800) & \text{otherwise,} \end{cases}$$

on the domain $x \in [-1.25, 1.25]$ with *periodic* boundary conditions:

$$(5.3) \quad f(t, x = -1.25, v) = f(t, x = +1.25, v).$$

We choose the truncated velocity domain $v \in [-7, 7]$ and the following parameters:

$$(5.4) \quad N_x = 256, \quad N_v = 128, \quad \varepsilon = 0.01, \quad \text{CFL} = 1.95, \quad T_{\text{final}} = 0.16.$$

The periodic boundary conditions are enforced during the transport sub-step via (3.13) and (3.14) by replacing all indices that stray out of the range $i = 1, 2, \dots, N_x$ by the following:

$$(5.5) \quad (-1) \rightarrow (N_x - 1), \quad (0) \rightarrow (N_x), \quad (N_x + 1) \rightarrow (1), \quad (N_x + 2) \rightarrow (2).$$

We implemented the algorithms described in the previous section, Algorithms 4.1 and 4.2, in freely available Java code, with corresponding plotting routines in Python (via NumPy and Matplotlib [6]). The particle density function (PDF) in the (x, v) phase space is shown at the initial ($t = 0$) and final ($t = 0.16$) times in Figure 1. Corresponding plots of the various moments of the distribution function are shown in Figure 2; these panels display the (a) density (ρ) at time $t = 0$, (b) density (ρ) at time $t = 0.16$, (c) fluid velocity (u) at time $t = 0$, (d) fluid velocity (u) at time $t = 0.16$, (e) temperature (T) at time $t = 0$, and (f) temperature (T) at time $t = 0.16$. These plots show that the initial discontinuous block in the center of the domain breaks up into shocks and rarefactions that propagate to the left and right. Because the Knudsen number is set to $\varepsilon = 0.01$, we also see the effect of viscosity on

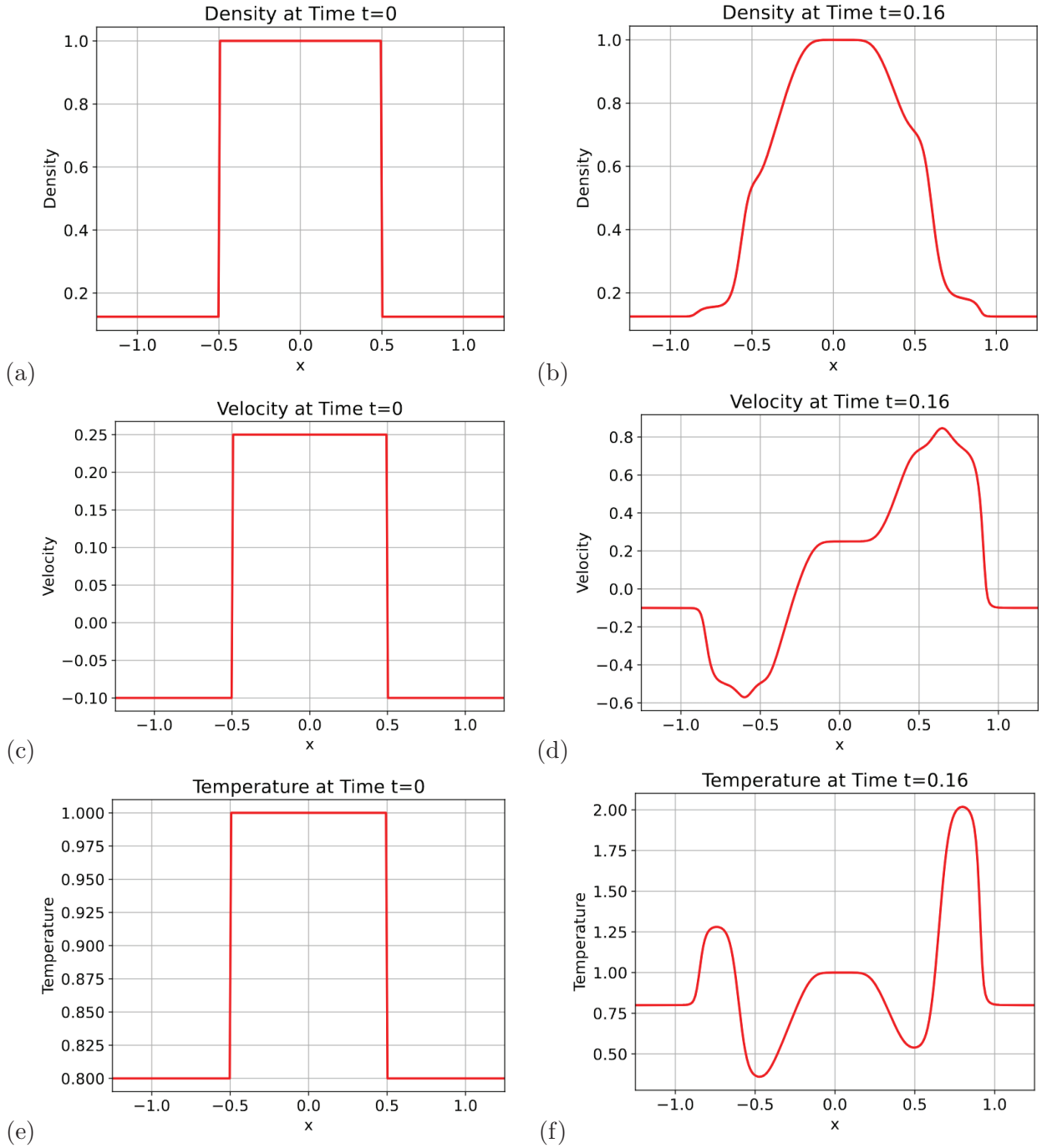


Figure 2. Plots of various moments of the distribution function. Shown in these panels are the (a) density (ρ) at time $t = 0$, (b) density (ρ) at time $t = 0.16$, (c) fluid velocity (u) at time $t = 0$, (d) fluid velocity (u) at time $t = 0.16$, (e) temperature (T) at time $t = 0$, and (f) temperature (T) at time $t = 0.16$. We used the mesh and parameter values described in §5.

smoothing out shocks and rarefactions. These simulation results are consistent with similar problems reported in the literature, including in Sar [16] and Pieraccini and Puppo [13].

The results shown here are with the modified Maxwell-Boltzmann distribution, $\widetilde{\mathcal{M}}$, as defined by (4.5). We have also run this example on the identical method, but with the standard Maxwell-Boltzmann distribution, $\mathcal{M}$, and, at least on the scale of the plots, the results with either $\widetilde{\mathcal{M}}$ or $\mathcal{M}$ are essentially indistinguishable. Upon closer inspection, however, there is a key difference: the modified approach with $\widetilde{\mathcal{M}}$ exactly conserves – up to machine precision – the mass, momentum, and energy, while the same method with $\mathcal{M}$ does not. We quantify this by defining the deviations from the initial total mass, momentum, and energy:

$$(5.6) \quad \text{relative total mass change: } \Delta\rho^n := \left| \sum_{i=1}^{N_x} \sum_{j=1}^{N_v} (f_{ij}^n - f_{ij}^0) \right| \bigg/ \left| \sum_{i=1}^{N_x} \sum_{j=1}^{N_v} f_{ij}^0 \right|,$$

$$(5.7) \quad \text{relative total momentum change: } \Delta m^n := \left| \sum_{i=1}^{N_x} \sum_{j=1}^{N_v} (v_j f_{ij}^n - v_j f_{ij}^0) \right| \bigg/ \left| \sum_{i=1}^{N_x} \sum_{j=1}^{N_v} v_j f_{ij}^0 \right|,$$

$$(5.8) \quad \text{relative total energy change: } \Delta\mathcal{E}^n := \left| \sum_{i=1}^{N_x} \sum_{j=1}^{N_v} (v_j^2 f_{ij}^n - v_j^2 f_{ij}^0) \right| \bigg/ \left| \sum_{i=1}^{N_x} \sum_{j=1}^{N_v} v_j^2 f_{ij}^0 \right|.$$

The relative deviations are shown in Figure 3, where the lack of exact conservation when using $\mathcal{M}$ is shown in 3(a), while exact preservation (at least up to machine precision) with the proposed modified collision operator is shown in 3(b). We also note that the relative deviations accumulate and grow over time when using $\mathcal{M}$.

Finally, we note that the Java/Python code that produced the plots in this section can be freely downloaded and executed [15]; the central portion of the code is written in Java, and the plotting routines are written in Python using NumPy and Matplotlib [6].

6. Conclusion. In this work, we developed a novel numerical method for solving the 1D1V Boltzmann-BGK equation. Several important ingredients were combined to create the resulting numerical method. First, we made use of Strang operator splitting [17] to create separate transport and collision sub-steps, each of which was easier to discretize than the whole system; second, we introduced a third-order accurate Lax-Wendroff-type scheme [9] for the transport sub-step; third, we made use of the second-order L-stable TR-BDF method [1] for the collision sub-step. These ingredients resulted in an accurate, efficient, and robust numerical method across all Knudsen numbers, $\varepsilon > 0$, including in the singular limit: $\varepsilon \rightarrow 0^+$. The final component we introduced is a novel treatment of the collision sub-step to guarantee that the total mass, momentum, and energy are conserved even in the presence of a truncated velocity range and quadrature errors in computing moments. The key idea was to multiply the Maxwell-Boltzmann distribution in the collision sub-step by a quadratic Hermite polynomial, where the coefficients are chosen to ensure exact conservation. The resulting scheme was verified on piecewise constant initial data with periodic boundary conditions; exact conservation up to machine precision was demonstrated. The resulting Java/Python code is freely available [15]; the central portion of the code is written in Java, and the plotting

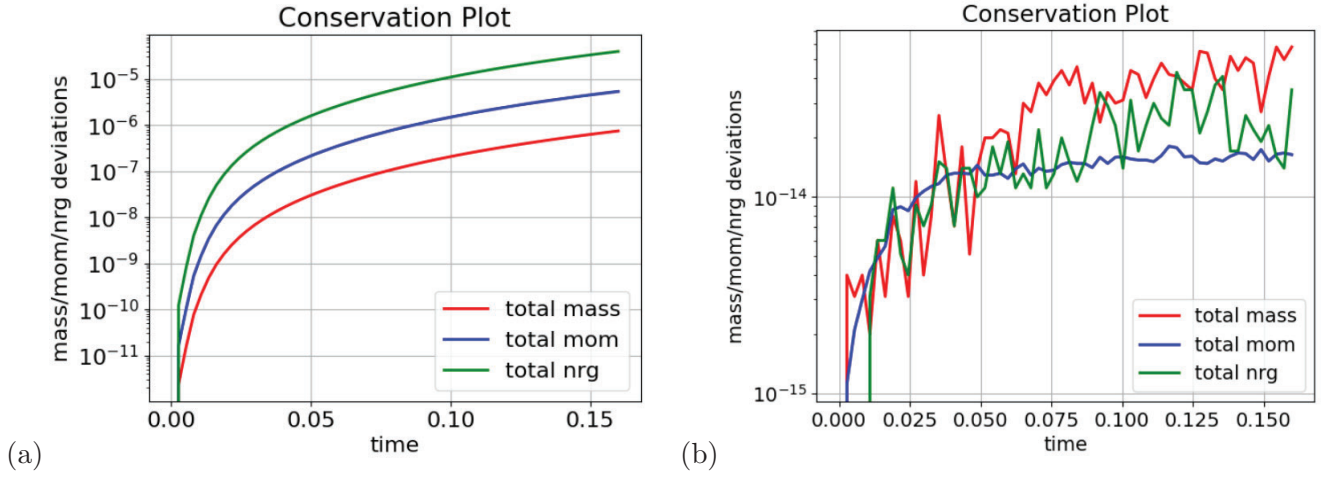


Figure 3. Conservation of mass, momentum, and energy. Shown in Panel (a) is the simulation without the collision operator modification (i.e., $\mathcal{M} \leftarrow \widetilde{\mathcal{M}}$). Shown in Panel (b) is the simulation with the proposed modified collision operator, $\widetilde{\mathcal{M}}$. These plots clearly show the lack of exact conservation when using $\mathcal{M}$; furthermore, the conservation errors tend to increase over time. The plots also clearly show the recovery of exact conservation (at least up to machine precision) with the proposed modified collision operator.

routines are written in Python using NumPy and Matplotlib. Future extensions of this work will focus on the higher-dimensional variants of the Boltzmann-BGK equation.

Acknowledgments. We would like to thank the anonymous reviewer for their thoughtful comments and suggestions that helped to improve this paper.

REFERENCES

- [1] R. BANK, W. COUGHRAN, W. FICHTNER, E. GROSSE, D. ROSE, AND R. SMITH, *Transient simulation of silicon devices and circuits*, IEEE Transactions on Electron Devices, 32 (1985), pp. 1992–2007, <https://doi.org/10.1109/T-ED.1985.22232>.
- [2] 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*, Physical Review, 94 (1954), pp. 511–525, <https://doi.org/10.1103/PhysRev.94.511>.
- [3] R. E. CAFLISCH, S. JIN, AND G. RUSSO, *Uniformly accurate schemes for hyperbolic systems with relaxation*, SIAM J. Numer. Anal., 34 (1997), pp. 246–281, <https://doi.org/10.1137/S0036142994268090>.
- [4] H. GRAD, *On the kinetic theory of rarefied gases*, Communications on Pure and Applied Mathematics, 2 (1949), pp. 331–407, <https://doi.org/10.1002/cpa.3160020403>.
- [5] M. GRADOBOEV AND V. RYKOV, *A conservative method for the numerical solution of the kinetic equation at low Knudsen numbers*, Comput. Math. Math. Phys., 34 (1994), pp. 205–222.
- [6] J. HUNTER, *Matplotlib: A 2D graphics environment*, Computing in Science & Engineering, 9 (2007), pp. 90–95, <https://doi.org/10.1109/MCSE.2007.55>.
- [7] S. JIN, *Runge-Kutta methods for hyperbolic conservation laws with stiff relaxation terms*, J. Comput. Phys., 122 (1995), pp. 51–67, <https://doi.org/10.1006/jcph.1995.1196>.
- [8] S. JIN, *Efficient asymptotic-preserving (AP) schemes for some multiscale kinetic equations*, SIAM J. Sci. Comput., 21 (1999), pp. 441–454, <https://doi.org/10.1137/S1064827598334599>.
- [9] P. LAX AND B. WENDROFF, *Systems of conservation laws*, Communications on Pure and Applied Mathematics, 13 (1960), <https://doi.org/10.1002/cpa.3160130205>.

- [10] W. LI, M. FANG, J. ZHAO, M. TAO, AND J. MEI, *A gas kinetic Lax–Wendroff scheme for low-speed isothermal rarefied gas flows*, *Physics of Fluids*, 34 (2022), p. 123102, <https://doi.org/10.1063/5.0126281>.
- [11] L. MIEUSSENS, *Discrete velocity model and implicit scheme for the BGK equation of rarefied gas dynamics*, *Mathematical Models and Methods in Applied Sciences*, 10 (2000), pp. 1121–1149, <https://doi.org/10.1142/S0218202500000562>.
- [12] L. PARESCHI AND G. RUSSO, *An introduction to Monte Carlo method for the Boltzmann equation*, *ESAIM: Proceedings*, 10 (2001), pp. 35–75, <https://doi.org/10.1051/proc:2001004>.
- [13] S. PIERACCINI AND G. PUPPO, *Implicit-explicit schemes for BGK kinetic equations*, *Journal of Scientific Computing*, 32 (2007), pp. 1–28, <https://doi.org/10.1007/s10915-006-9116-6>.
- [14] E. ROOHI, H. AKHLAGHI, AND S. STEFANOV, *A Brief Review of Kinetic Theory*, Springer Nature Singapore, Singapore, 2025, pp. 37–83, https://doi.org/10.1007/978-981-96-8200-3_2.
- [15] V. B. ROSSMANITH, *Boltzmann-BGK code*. <https://github.com/ViennaRossmanith/Numerical-Gas-Simulation>, September 2025.
- [16] P. SAR, *Asymptotic Preserving Schemes for the Kinetic Boltzmann-ES-BGK Equation*, PhD thesis, Iowa State University, 2024. <https://www.proquest.com/docview/3165604968>.
- [17] G. STRANG, *On the construction and comparison of difference schemes*, *SIAM J. Numer. Anal.*, 5 (1968), pp. 506–517, <https://doi.org/https://doi.org/10.1137/070504>.
- [18] V. TITAREV, *Conservative numerical methods for kinetic equations*, *Computers and Fluids*, 36 (2007), pp. 1446–1459, <https://doi.org/doi:10.1016/j.compfluid.2007.01.009>.