Efficient numerical methods for solving the Boltzmann equation for small scale flows

by

Lowell L. Baker

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2007

Author .....
Department of Mechanical Engineering
May 21, 2007

Certified by .....
Nicolas G. Hadjiconstantinou
Associate Professor
Thesis Supervisor

Accepted by .....
Lallit Anand
Department Committee on Graduate Students

Efficient numerical methods for solving the Boltzmann equation for small scale flows

by

Lowell L. Baker

Submitted to the Department of Mechanical Engineering
on May 21, 2007, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Mechanical Engineering

Abstract

The Navier-Stokes equations of continuum fluid mechanics fail to accurately describe dilute gas flows when the characteristic lengthscale of the system is on the order of (or smaller than) the molecular mean free path. At these lengthscales, gaseous hydrodynamics may be described by a kinetic description, namely the Boltzmann equation. Currently, the prevalent method for solving the Boltzmann equation is a particle simulation method known as direct simulation Monte Carlo (DSMC). DSMC is very efficient for high-speed (more generally, high signal) flows; unfortunately, due to the statistical sampling used to obtain hydrodynamic fields, the computational cost of DSMC (for a given signal to noise ratio) increases rapidly with decreasing signal. For example, the computational cost for calculating the flow velocity with a fixed signal to noise ratio scales with Ma^{-2} as $\mathrm{Ma} \to 0$ (Ma is the Mach number). As a result, simulation of many low-signal flows of practical interest (for example, in micro- and nano-scale devices) is currently not feasible using DSMC.

This thesis describes how the above limitation can be alleviated through the use of variance reduction techniques. In particular, we show that by simulating only the deviation from equilibrium, one can devise a variety of numerical methods that have a computational cost that is both small and independent of the magnitude of this deviation. For low-speed flows, this leads to methods that are significantly more efficient than DSMC.

Two implementations of this variance reduction concept are presented. The first is a particle method akin to DSMC, differing only in ways necessary to simulate the deviation from equilibrium. This particle formulation retains the most important strengths of DSMC – specifically, importance sampling (providing computational efficiency) and the ability to capture discontinuities in the solution – while offering a significant computational advantage compared to DSMC for low-signal flows. The second approach considered is a PDE-based method using a discontinuous Galerkin formulation, which is able to treat travelling discontinuities. This PDE-based approach has the potential for high-order accuracy, as well as implicit steady-state formulations which can be significantly more efficient when transient phenomena are not of interest.

Thesis Supervisor: Nicolas G. Hadjiconstantinou
Title: Associate Professor

Acknowledgments

First, I would like to thank my advisor, Nicolas Hadjiconstantinou for his patience, support and encouragement over the course of my graduate studies.

The members of my thesis committee, Prof. Anthony Patera and Prof. Gang Chen, have provided numerous helpful comments, ideas and suggestions.

I would like to thank Dr. Michail Gallis and Dr. Steve Kempka for helpful discussions and support, and for hosting me at Sandia.

I am indebted to Prof. Jaime Peraire for his assistance with the discontinuous Galerkin method.

Much thanks to Deborah Alibrandi, for her help with applications, paperwork and reservations.

Finally, to my officemates Sanith, Markus, Husain, Anastassia, Kristin, Yunqing, Joe, Saeed, Ghassan, Thomas, Ho-Man, Zhengyi - thanks for making the office such a lively place.

This work was supported in part by Sandia National Laboratories / National Science Foundation.

Contents

Chapter 1

Introduction

Efficient numerical techniques for modeling small scale, dilute gas flows are expected to be of increasing importance as micro- and nano-scale engineering becomes more prevalent. This thesis discusses the development of efficient numerical methods that can be used for the design and optimization of MEMS and NEMS (micro/nano electromechanical systems); additionally these methods can be used as a tool for gaining a better understanding of the physics of dilute gas flow.

At present, the most prevalent method for simulating dilute gas flow is the direct simulation Monte Carlo (DSMC) method [12, 2] (discussed further in Chapter 3). This method has been extremely successful for simulating high-speed dilute gas flow (for example in aerospace applications), unfortunately its computational efficiency is very poor for low-speed (or more generally low-signal[1]) applications. For example, in DSMC, the computational cost associated with estimating the flow velocity $\mathbf{u}$ with a relative uncertainty E_u scales as $(\mathrm{Ma}E_u)^{-2}$ where Ma is the flow Mach number[22]; as a result, the simulation of low-speed flows with DSMC is essentially intractable; e.g. to obtain 1% statistical uncertainty in a 1m/s flow at room temperature, one would need on the order of 5×10^8 *independent* samples per cell (and per timestep, for transient calculations)[22].

This thesis describes the development and application of a general variance re-

[1]In the present context, low-signal means that the quantity of interest is small relative to the appropriate normalizing factor. For example, both low Mach number flows and flows with small temperature gradients would fall under this heading.

duction technique for efficiently evaluating the collision integral of the Boltzmann equation, which is the governing equation for dilute gas flow. The central idea of this variance reduction technique is to simulate the *deviation* from equilibrium in a manner that yields a highly efficient method when this deviation is small (as will typically be the case for low-speed flows), while remaining accurate even if the deviation is large. This variance reduction technique enables simulation of gaseous flows at low speeds with a computational cost that is *independent* of the flow speed.

This variance reduction technique is sufficiently general to allow a variety of implementations; this thesis describes the use of this technique in both a particle simulation method (similar to DSMC) and a PDE-based approach (based on the discontinuous Galerkin method). As will be shown, both of these methods are able to provide essentially noise-free solutions of the Boltzmann equation under arbitrary flow conditions. In particular, accurate solutions of low-speed flows are readily obtained, without sacrificing applicability to the general case.

1.1 Overview

The present chapter introduces the kinetic description for dilute gas flow and the Boltzmann equation. The set of dimensionless units and the process for obtaining hydrodynamic fields from the kinetic description are given. Additionally, the motivation for solving the full Boltzmann equation, instead of a simplified equation, is discussed.

In Chapter 2, the variance reduction concepts used in this thesis are discussed in a general context. The application of variance reduction to the collision integral as well as a brief discussion of the advantages and interpretations of these methods is presented.

As mentioned previously, variance reduction can be implemented in a variety of ways for the Boltzmann equation. In this thesis, two *independent* approaches sharing the same core ideas are presented. The first implementation discussed is a particle-based formulation drawing from DSMC.

To motivate this particle-based formulation, we first briefly discuss DSMC in Chapter 3. We illustrate how DSMC uses one variance reduction method, specifically importance sampling, to obtain its formidable computational efficiency (for high-signal flows).

We then extend DSMC algorithm by incorporating an additional variance reduction technique. In Chapter 4 we show how one can develop a particle-based formulation that enables the simulation of only the *deviation* from equilibrium. The resulting method is similar to DSMC, and retains many of DSMC's traditional strengths – in fact DSMC is retained as a special case. However, in contrast to DSMC, the resulting method is extremely efficient for low-signal flows.

In Chapter 5, we show how the variance reduction ideas of Chapter 2 can be extended to a direct numerical formulation. In this chapter, we use the Runge Kutta discontinuous Galerkin (RKDG) method[17], which is a finite element formulation applicable to hyperbolic equations. The resulting method combines the strengths of the discontinuous Galerkin (DG) formulation – namely high order accuracy in all dimensions and the ability to capture discontinuities – with high efficiency for low-signal flows resulting from the variance reduction techniques used.

Finally, in Chapter 6 an iterative method for obtaining steady-state solutions to the Boltzmann equation is developed using the variance-reduced discontinuous Galerkin (DG) formulation from Chapter 5.

1.2 Kinetic description of gas flow

1.2.1 Flow regimes

At macroscopic length scales, gas flows are typically well described by the Navier-Stokes equations. This continuum description is appropriate when the characteristic length scale of the physical domain is much larger than the molecular mean free path (the average distance molecules travel between collisions) and in the absence of steep gradients in fluid properties (such as in the interior of a shock wave). However, as the

length scale of the flow decreases relative to the mean free path, the Navier-Stokes equations cease to be valid and more general approaches must be used. It is these regimes, in which the continuum description fails, that are of primary interest in this work.

The breakdown of the Navier-Stokes equations can be quantified by introducing the Knudsen number, $\mathrm{Kn} \equiv \lambda^*/\ell^*$, the dimensionless ratio of the mean free path to the characteristic length scale of the flow (in this thesis, dimensional quantities are indicated by a $*$ superscript). When $\mathrm{Kn} \lesssim 10^{-3}$ or, in other words when the relevent length scale is very large compared to the mean free path, the Navier-Stokes equations (supplemented by the usual no-slip boundary conditions) hold. As the Knudsen number increases, slip begins to become important at the domain boundaries. It can be shown that by replacing the no-slip boundary conditions by the slip boundary conditions, one can still obtain good approximations to the flow[14].

When $\mathrm{Kn} \gtrsim 0.1$, the Navier-Stokes equations are not valid, even in the interior of the domain, and one must solve the Boltzmann equation, which governs the hydrodynamics of dilute gases at all Knudsen numbers and under general flow conditions[2]. For $\mathrm{Kn} \gtrsim 10$ (known as the free molecular flow regime) inter-molecular collisions are so infrequent that they can be neglected. In this case, flow solutions can be obtained by solving the collisionless Boltzmann equation, which is significantly more amenable to analysis.

This thesis will focus primarily on the transition regime, $10^{-1} \lesssim \mathrm{Kn} \lesssim 10$, in which neither the continuum flow approximations nor the collisionless Boltzmann equation are applicable, and the full Boltzmann equation must be solved.

1.2.2 The Boltzmann equation

In the framework of the kinetic theory of gasses [14, 37, 12], the state of a dilute gas is specified by the distribution function $f^* = f^*(\mathbf{x}^*, \mathbf{c}^*, t^*)$, defined such that $f^*\, d^3\mathbf{x}^*\, d^3\mathbf{c}^*$ is the expected number of molecules with a position in the range $d^3\mathbf{x}^*$ about $\mathbf{x}^*$ and a velocity in the range $d^3\mathbf{c}^*$ about $\mathbf{c}^*$ at time t^* [37]. (Recall that the

[2]See [37, 12, 14, 33] for a discussion of the assumptions inherent in the Boltzmann equation.

starred quantities are dimensional.)

The evolution of the distribution function in time is governed by the Boltzmann equation [37, 12, 14]

$$\frac{\partial f^*}{\partial t^*} + \mathbf{c}^* \cdot \frac{\partial f^*}{\partial \mathbf{x}^*} + \mathbf{a}^* \cdot \frac{\partial f^*}{\partial \mathbf{c}^*} = \left[\frac{df}{dt}\right]^*_{\text{coll}} \tag{1.1}$$

Here, $\mathbf{a}^* \equiv \mathbf{F}^*/m^*$ is the acceleration resulting from the body force $\mathbf{F}^*$ acting on a molecule of mass m^*. The Boltzmann equation is a conservation law for the distribution function in the six dimensional phase space (three physical space dimensions and three velocity space dimensions).

The term

$$\frac{\partial f^*}{\partial t^*} \tag{1.2}$$

describes the change in the number of molecules with a position $\mathbf{x}^*$ and a velocity $\mathbf{c}^*$. There are three things that cause this change: the first is the molecules changing position do to their velocity, accounted for by

$$\mathbf{c}^* \cdot \frac{\partial f^*}{\partial \mathbf{x}^*} \tag{1.3}$$

and the second is due to the velocity of particles changing because of the acceleration due to body forces

$$\mathbf{a}^* \cdot \frac{\partial f^*}{\partial \mathbf{c}^*} \tag{1.4}$$

Finally, the third is encompassed in the right side of the Boltzmann equation, known as the collision integral. The collision integral represents a source term due to intermolecular collisions impulsively changing the velocities of molecules. The collision integral, $\left[\frac{df}{dt}\right]^*_{\text{coll}} = \left[\frac{df}{dt}\right]^*_{\text{coll}}(\mathbf{x},^* \mathbf{c}^*, \mathbf{t}^*)$ can be written in the form

$$\left[\frac{df}{dt}\right]^*_{\text{coll}} = \iint \left(f_1^{*\prime} f^{*\prime} - f_1^* f^*\right) \mathrm{g}^* \sigma^* \, d^2\Theta \, d^3\mathbf{c}_1^* \tag{1.5}$$

Here, $\mathbf{c}_1^*$ is a the molecular velocity of the "bullet" molecule, which collides with a molecule of velocity $\mathbf{c}$; together $\mathbf{c}^*$ and $\mathbf{c}_1^*$ are referred to as the pre-collision velocities.

The post-collision velocities, $\mathbf{c}^{*\prime}$ and $\mathbf{c}_1^{*\prime}$, are related to the pre-collision velocities through the scattering angle Θ, which is a solid angle on the unit sphere. In this thesis, integration over Θ extends over the unit sphere and integration over other coordinates (here, $\mathbf{c}_1^*$) extends over the entire space, unless otherwise noted. We define $\mathrm{g}^* \equiv \|\mathbf{c}_1^* - \mathbf{c}^*\| = \|\mathbf{c}^{*\prime}{}_1 - \mathbf{c}^{*\prime}\|$ to be the relative speed, equal before and after the collision due to conservation of energy. The parameter σ^* is known as the (differential) collision cross section; for a hard sphere gas of diameter d^*, $\sigma^*_{\text{hard sphere}} = d^{*2}/4$; expressions for σ^* for other interaction potentials are also available [37, 12]. To simplify the notation, we have defined

$$f^* \equiv f^*(\mathbf{x}^*, \mathbf{c}^*, t^*) \tag{1.6a}$$

$$f_1^* \equiv f^*(\mathbf{x}^*, \mathbf{c}_1^*, t^*) \tag{1.6b}$$

$$f^{*\prime} \equiv f^*(\mathbf{x}^*, \mathbf{c}^{*\prime}, t^*) \tag{1.6c}$$

$$f_1^{*\prime} \equiv f^*(\mathbf{x}^*, \mathbf{c}_1^{*\prime}, t^*) \tag{1.6d}$$

The post collision velocities are related to the scattering angle and the pre-collision velocities by [2]

$$\mathbf{c}^{*\prime} = \frac{\mathbf{c}^* + \mathbf{c}_1^*}{2} + \frac{1}{2}\mathbf{c}_r^{*\prime} \tag{1.7a}$$

$$\mathbf{c}_1^{*\prime} = \frac{\mathbf{c}^* + \mathbf{c}_1^*}{2} - \frac{1}{2}\mathbf{c}_r^{*\prime} \tag{1.7b}$$

where the post-collision relative velocity vector is given by [4, 2]

$$\mathbf{c}_r^{*\prime} = \mathrm{g}^* \left[\sin\vartheta\cos\varphi, \sin\vartheta\sin\varphi, \cos\vartheta\right] \tag{1.8}$$

where φ is the azimuthal component and ϑ is polar angle of the solid angle Θ on the unit sphere.

For an detailed interpretation of the collision integral, we refer to reader to discussions in [37, 12]. For our purposes, it will be enough to recognize that the collision rate between molecules of velocity $\mathbf{c}^*$ and molecules with a velocity $\mathbf{c}_1^*$ with a scatter-

ing angle Θ is given by $f_1^* f^* g^* \sigma^*$ (where, in general, σ^* depends on the pre-collision velocities and the scattering angle). The expression

$$-\iint f_1^* f^* \mathrm{g}^* \sigma^* \, d^2\Theta \, d^3\mathbf{c}_1^* \tag{1.9}$$

is thus the total rate at which molecules of velocity $\mathbf{c}^*$ are scattered by all collisions. The negative sign indicates that these collisions reduces the number of molecules with a velocity $\mathbf{c}^*$. Similarly it can be shown [37] that

$$\iint f_1^{*\prime} f^{*\prime} \mathrm{g}^* \sigma^* \, d^2\Theta \, d^3\mathbf{c}_1^* \tag{1.10}$$

is the rate at which molecules of velocity $\mathbf{c}^*$ are created by collisions; that is the rate of collisions that have $\mathbf{c}^*$ as a post-collision velocity.

In this thesis, we shall also make significant use of an alternative formulation for the collision integral. We begin with the weak form of the collision integral [37]

$$\int v_1 \left[\frac{df}{dt}\right]^*_{\mathrm{coll},1} d^3\mathbf{c}_1^* = \frac{1}{2} \iiint (v_1' + v_2' - v_1 - v_2) f_1^* f_2^* g^* \sigma^* \, d^2\Theta \, d^3\mathbf{c}_1^* \, d^3\mathbf{c}_2^* \tag{1.11}$$

in which $v_1 = v_1(\mathbf{x}^*, \mathbf{c}_1^*)$ is a test function[3] and $g^* = \|\mathbf{c}_1^* - \mathbf{c}_2^*\| = \|\mathbf{c}_1^{*\prime} - \mathbf{c}_2^{*\prime}\|$ is the relative speed and $\left[\frac{df}{dt}\right]^*_{\mathrm{coll},1} = \left[\frac{df}{dt}\right]^*_{\mathrm{coll},1}(\mathbf{x}^*, \mathbf{c}_1^*, t)$. To simplify the notation, we have defined

$$v_1 \equiv v(\mathbf{x}^*, \mathbf{c}_1^*, t^*) \tag{1.12a}$$

$$v_2 \equiv v(\mathbf{x}^*, \mathbf{c}_2^*, t^*) \tag{1.12b}$$

$$v_1' \equiv v(\mathbf{x}^*, \mathbf{c}_1^{*\prime}, t^*) \tag{1.12c}$$

$$v_2' \equiv v(\mathbf{x}^*, \mathbf{c}_2^{*\prime}, t^*) \tag{1.12d}$$

in a similar manner as in (1.6a) – (1.6d).

[3]The function v could of course be replaced with a dimensional function v^*.

If we take $v_1 = \delta(\mathbf{c}_1^* - \mathbf{c}^*)$, we obtain the result

$$\left[\frac{df}{dt}\right]^*_{\text{coll}} = \frac{1}{2}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2) f_1^* f_2^* g^* \sigma^* \, d^2\Theta \, d^3\mathbf{c}_1^* \, d^3\mathbf{c}_2^* \tag{1.13}$$

We note that this expression for the collision integral has a simple interpretation. The term $f_1^* f_2^* g^* \sigma^*$ is the collision rate between molecules with velocity $\mathbf{c}_1^*$ and molecules with velocity $\mathbf{c}_2^*$ and a scattering angle Θ. The expression (1.18) can thus be interpreted as integrating the collision rate over all possible collisions, with the delta functions selecting for those collisions involving (either as a pre- or post-collision velocity) the velocity of interest $\mathbf{c}^*$. The delta functions at the pre-collision velocities have a negative sign; each collision for which a molecule of class $\mathbf{c}^*$ is involved *reduces* the number of molecules of class $\mathbf{c}^*$. Similarly, the positive sign for the delta functions at the post-collision velocities indicates that collisions for which $\mathbf{c}^*$ is the post-collision velocity increase the number of molecules with velocity $\mathbf{c}^*$. We also note that there is a factor of $1/2$ to account for double-counting – interchanging $\mathbf{c}_1^*$ and $\mathbf{c}_2^*$ yields an indistinguishable collision, but these are considered separately in the above expression.

1.2.3 Non-dimensional units

It is convenient to introduce a set of dimensionless variables. We will use a characteristic molecular mean free path λ^* as our characteristic lengthscale, the most probable molecular speed $\bar{c}^* = \sqrt{\frac{2k^*\bar{T}^*}{m^*}}$ as our characteristic velocity, and as a characteristic timescale we will use $\bar{t}^* \equiv \frac{\sqrt{\pi}}{2}\frac{\lambda^*}{\bar{c}^*}$. Here, k^* is Boltzmann's constant, $\bar{T}^*$ is a reference temperature, and m^* is the molecular mass.

In the examples presented in this thesis, we will typically use the hard sphere collision cross section[4]; in this case, the molecular mean free path is given by

$$\lambda^* = \frac{1}{\sqrt{2}\pi {d^*}^2 \bar{n}^*} \tag{1.14}$$

[4]None of the methods developed in this thesis are *limited* to hard spheres.

where $\bar{n}^*$ is the reference number density. Additionally, for the hard-sphere case $\bar{t}^*$ will be the mean time between collisions.

We will then define our set of dimensionless variables as

$$t = \frac{1}{\bar{t}^*} t^* \tag{1.15a}$$

$$\mathbf{x} = \frac{1}{\lambda^*} \mathbf{x}^* \tag{1.15b}$$

$$\mathbf{c} = \frac{1}{\bar{c}^*} \mathbf{c}^* \tag{1.15c}$$

$$f = \frac{\bar{c}^{*3}}{\bar{n}^*} f^* \tag{1.15d}$$

$$\mathbf{a} = \frac{\bar{t}^*}{\bar{c}^*} \mathbf{a}^* \tag{1.15e}$$

$$\sigma = \bar{n}^* \lambda^* \sigma^* \tag{1.15f}$$

Using this nondimensionalization, the Boltzmann equation can be written

$$\frac{\partial f}{\partial t} + \frac{\sqrt{\pi}}{2} \mathbf{c} \cdot \frac{\partial f}{\partial \mathbf{x}} + \mathbf{a} \cdot \frac{\partial f}{\partial \mathbf{c}} = \left[\frac{df}{dt}\right]_{\text{coll}} \tag{1.16}$$

with the dimensionless form of the collision integral given by

$$\left[\frac{df}{dt}\right]_{\text{coll}} = \frac{\sqrt{\pi}}{2} \iint (f_1' f' - f_1 f)\, \mathrm{g}\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \tag{1.17}$$

The alternative form of the collision integral can be written

$$\left[\frac{df}{dt}\right]_{\text{coll}} = \frac{\sqrt{\pi}}{4} \iint (\delta_1' + \delta_2' - \delta_1 - \delta_2) f_1 f_2 g \sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \tag{1.18}$$

and the weak form of the collision integral is given by

$$\int v_1 \left[\frac{df}{dt}\right]_{\text{coll},1} d^3\mathbf{c}_1 = \frac{\sqrt{\pi}}{4} \iiint (v_1' + v_2' - v_1 - v_2) f_1 f_2 g \sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \tag{1.19}$$

1.2.4 Maxwell-Boltzmann distribution

The equilibrium distribution for the Boltzmann equation is known as a Maxwell-Boltzmann distribution, given by [37]

$$f^{MB}(\mathbf{c}) \equiv n\pi^{-3/2}T^{-3/2}\exp\left(-\frac{(\mathbf{c}-\mathbf{u})^2}{T}\right) \tag{1.20}$$

The nondimensional parameters describing a Maxwell-Boltzmann distribution are the number density $n \equiv n^*/\bar{n}^*$, the temperature $T \equiv T^*/\bar{T}^*$ and the mean velocity $\mathbf{u} = \mathbf{u}^*/\bar{c}^*$.

The Maxwell-Boltzmann distribution satisfies

$$\left[\frac{df}{dt}\right]_{\text{coll, MB}} = 0 \tag{1.21}$$

for any choice of parameters $\{n, T, \mathbf{u}\}$.

1.2.5 Hydrodynamic fields

The distribution function provides a complete description of a dilute gas flow. On the other hand, in typical applications one will be interested in macroscopic properties, such as the fluid velocity, shear stress, temperature and heat flux. These quantities can be obtained as moments of the distribution function.

$$n = \int f\, d^3\mathbf{c} \tag{1.22a}$$

$$u_i = \frac{1}{n}\int c_i f\, d^3\mathbf{c} \tag{1.22b}$$

$$P_{i,j} = \int (c_i - u_i)(c_j - u_j) f\, d^3\mathbf{c} \tag{1.22c}$$

$$T = \frac{2}{3n}\int \|\mathbf{c}-\mathbf{u}\| f\, d^3\mathbf{c} \tag{1.22d}$$

$$q_i = \frac{1}{2}\int (c_i - u_i)\|\mathbf{c}-\mathbf{u}\| f\, d^3\mathbf{c} \tag{1.22e}$$

Here, n is the number density, $\mathbf{u}$ is the fluid velocity, $\mathbf{P}$ is the stress tensor, T is the temperature, $\mathbf{q}$ is the heat flux and i and j index the vector components.

The dimensional values for these quantities can be obtained as follows

$$n^* = \bar{n}^* n \tag{1.23a}$$

$$\mathbf{u}^* = \bar{c}^* \mathbf{u} \tag{1.23b}$$

$$\mathbf{P}^* = \bar{n}^* m^* \bar{c}^{*2} \mathbf{P} \tag{1.23c}$$

$$T^* = \bar{T}^* T \tag{1.23d}$$

$$\mathbf{q}^* = m^* n^* \bar{c}^{*3} \mathbf{q} \tag{1.23e}$$

1.3 Previous work on numerical solutions of the Boltzmann equation

Experience has shown that one of the most difficult aspects of solving the Boltzmann equation is the evaluation of the collision integral. Thus, much of this thesis will focus on evaluating this term efficiently.

The collision integral is a five dimensional integral with an integrand that is, in general, discontinuous. Moreover, in order to solve the Boltzmann equation, the collision integral must be evaluated at a large number of points in phase space (physical and velocity space) and time.

To cope with the difficulty in evaluating the collision integral, a variety of numerical solution techniques have been developed. Here, we will discuss two representative approaches, both of which are typically implemented using deterministic approaches, which do not suffer from statistical uncertainty limitations. The first is based on the relaxation time (or BGK) model [37], in which the collision integral is approximated as

$$\left[\frac{df}{dt}\right]_{\text{coll, relaxation}} = -\frac{f - f^0}{\tau} \tag{1.24}$$

where τ is an empirical relaxation time and f^0 is the assumed equilibrium distribution. While this approach results in a simplified governing equation, the tradeoff is a lack

of fidelity resulting from this rather crude model. For example, this approximation predicts the Prandtl number for an ideal gas to be 1 [37].

Another notable approach consists of obtaining direct numerical solutions of the linearized Boltzmann equation, which can be written (in the absence of body forces)

$$\frac{\partial \eta f^{MB}}{\partial t} + \frac{\sqrt{\pi}}{2}\mathbf{c} \cdot \frac{\partial \eta f^{MB}}{\partial \mathbf{x}} = \frac{\sqrt{\pi}}{2} \iint \left(\eta_1' + \eta' - \eta_1 - \eta\right) f^{MB} f_1^{MB} \mathrm{g}\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \quad (1.25)$$

where η is a small perturbation from equilibrium, defined by the relationship $f = (1+\eta)f^{MB}$.

This approach is sufficient for many of the low-speed cases of interest in this work. However, as will be shown later, evaluation of the right side of (1.25) by direct quadrature has proven sufficiently costly that similarity solutions have been used to reduce the dimensionality of velocity space from three to two [35]. Additionally, for time-independent problems, the discontinuities in the distribution function are stationary and can be aligned with mesh elements. Extentions of this method to two spatial dimensions has been done with the collision integral replaced by the BGK model, and velocity space again reduced to two dimensions [3]. While the performance of computers has advanced immensely since these papers were published (1989 and 2001, respectively), this author is unaware of any implementation that is able to practically and accurately simulate flows of interest in two or three physical dimensions, for the general case, using these methods.

Using the methods developed in this thesis, it is possible to retain the non-linear terms of the Boltzmann equation in a way that leads to negligible additional computational expense in the case where these terms are small, while maintaining the applicability of the method to cases where these terms are not small. This allows the user to apply the method without concern as to whether the non-linear terms are relevant for a particular problem.

Additionally, in this thesis, all work will be done using the full three dimensional velocity space. While the work presented here will use only zero or one dimensions in physical space, the methods can be directly extended to higher dimensional problems.

Preliminary estimates indicate that problems in two physical dimensions should be tractable on a single (circa 2007) workstation, while three dimensional problems are feasible on a small cluster.

Chapter 2

Variance reduction techniques for evaluating the Boltzmann collision integral

In this chapter, we will discuss Monte Carlo evaluation of the collision integral for the nonlinear Boltzmann equation. We focus on the use of variance reduction techniques, specifically importance sampling and control variate integration, to improve the efficiency of the Monte Carlo integration. These variance reduction techniques yield a highly efficient means by which to evaluate the collision integral, and form the central theme of this thesis. In later chapters, we will illustrate how these variance reduction techniques can be incorporated into a variety of solution methods for the Boltzmann equation; however in this chapter we will focus on the variance reduction techniques in isolation.

2.1 Standard Monte Carlo integration

The collision integral is a high-dimensional integral with an integrand that is, in general, discontinuous. Thus, Monte Carlo techniques are a natural choice for its evaluation [32, 27]. As a starting point, we consider direct application of Monte

Carlo integration to the problem at hand[1]

Monte Carlo integration [32] of a function $y(\mathbf{r})$ over a region $\mathcal{R}$ in any number of dimensions can be performed by approximating

$$\int_{\mathcal{R}} y(\mathbf{r})\, d\mathbf{r} = \mathcal{V} \times \langle y \rangle \tag{2.1}$$

$$\approx \frac{\mathcal{V}}{N} \sum_{i=1}^{N} y(\mathbf{r}_i) \tag{2.2}$$

where $\mathcal{V}$ is the volume of the region $\mathcal{R}$ and $\mathbf{r}_i \in \mathcal{R}$ is a point chosen at random with a uniform probability distribution over $\mathcal{R}$. Here $\langle \cdot \rangle$ denotes the expected value.

The statistical uncertainty of this method scales with [32]

$$\mathcal{V} \sqrt{\frac{\langle y^2 \rangle - \langle y \rangle^2}{N}} \tag{2.3}$$

2.1.1 Standard Monte Carlo evaluation of the collision integral

To apply this Monte Carlo integration approach to the Boltzmann collision integral, we restrict integration to a finite region in velocity space[4], instead of to infinity. In practice, this truncation of velocity space introduces a negligible error if the maximum considered speeds are sufficiently large. Typical implementations will include (dimensionless) speeds up to the order of 3 to 5.

Applying Monte Carlo integration to the form of the collision integral (1.17)

$$\left[\frac{df}{dt}\right]_{\text{coll}} = \frac{\sqrt{\pi}}{2} \iint (f_1' f' - f_1 f)\, g\sigma\, d^2\Theta\, d^3\mathbf{c}_1 \qquad [1.17]$$

we obtain

$$\left[\frac{df}{dt}\right]_{\text{coll}} \approx \frac{\sqrt{\pi}}{2} 4\pi \mathcal{V} \frac{1}{N} \sum_{i=1}^{N} \left(f_{1,i}' f_i' - f_{1,i} f\right) g_i \sigma_i \tag{2.4}$$

Here, i indexes the Monte Carlo sample, $\mathcal{V}$ is the volume from which $\mathbf{c}_{1,i}$ is randomly

[1] See [30] for an early implementation of standard Monte Carlo evaluation of the collision integral, or [39] for a later review.

(and uniformly) chosen and 4π is the area of the unit sphere, from which Θ_i is chosen.

Evaluating the collision integral using equation (2.4) is straightforward, however one must typically evaluate this sum for *every* point in phase space at every timestep or iteration. This results in a method that is far too slow to be competitive with DSMC, however we shall see that incorporating variance reduction techniques can significantly improve the effectiveness of Monte Carlo integration.

To motivate the first variance reduction technique, let us examine equation (2.4) more closely. In evaluating the collision integral for all points in velocity space, $\|\mathbf{c}\|$ and $\|\mathbf{c}_1\|$ will often be large (that is, at the extremes of our finite region in velocity space), and the corresponding value $ff_1g\sigma$ will be small[2]. One would also expect the values for $f_1'f'g\sigma$ to be small in these cases. Thus, these terms will not contribute significantly to the collision integral. Physically, this represents the fact that collisions involving molecules with large speeds are very rare, because these molecules are themselves rare. Similarly, collisions resulting in molecules with large speeds are also rare.

In this Monte Carlo scheme, however, these rare collision events will dominate the computational cost. Additionally, each of these rare collision events only contributes a small amount to (2.4); however to obtain an accurate method these collisions must be considered. One way to include these rare collisions, while maintaining a reasonable computational cost, is to utilize importance sampling.

2.2 Importance sampling

In importance sampling, the sample points are distributed nonuniformly, with the aim of focusing the samples on the regions where the integrand is most significant. Assuming $p(\mathbf{r})$ is a (normalized, by definition) probability distribution defined on $\mathcal{R}$,

[2]The magnitude of the distribution function for large molecular velocities is expected to decay roughly as $\exp(-\|\mathbf{c}\|^2)$, much faster than the relative speed g increases

we can write

$$\int_{\mathcal{R}} y(\mathbf{r})\, d\mathbf{r} \approx \frac{1}{N} \sum_{i=1}^{N} \frac{y(\mathbf{r}_i)}{p(\mathbf{r}_i)} \tag{2.5}$$

where $\mathbf{r}_i$ is chosen with a probability p. The statistical uncertainty of this integration method scales with [32]

$$\sqrt{\frac{\left\langle \left(\frac{y}{p}\right)^2 \right\rangle - \left\langle \frac{y}{p} \right\rangle^2}{N}} \tag{2.6}$$

It is obvious that (2.6) corresponds to (2.4) when one takes $p = 1/\mathcal{V}$. Furthermore, if p is a "good" approximation (in the sense that $\langle (y/p)^2 \rangle - \langle y/p \rangle^2 < \langle y^2 \rangle - \langle y \rangle^2$, or in other words if the variance of y/p is less than the variance of y), we will obtain better accuracy through the use of importance sampling. (Of course, it is necessary that samples from the distribution p be efficient to generate.)

2.2.1 Importance sampling for the Boltzmann collision integral

The use of importance sampling for the Boltzmann equation (in a discrete velocity context) was presented in [36]. As was noted in that paper (and will be discussed in section 3.2), this method has much in common with particle methods, such as DSMC.

We can use importance sampling to evaluate the form (1.18) of the collision integral [36]. We write

$$\left[\frac{df}{dt}\right]_{\text{coll}} = \chi \frac{\sqrt{\pi}}{4} \iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2) \frac{f_1 f_2 g \sigma}{\chi}\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \tag{2.7}$$

where we have multiplied and divided by the normalizing constant

$$\chi \equiv \iiint f_1 f_2 g \sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \tag{2.8}$$

Noting that $\frac{f_1 f_2 g \sigma}{\chi}$ is a normalized probability distribution, we can perform importance

sampling and obtain

$$\left[\frac{df}{dt}\right]_{\text{coll}} \approx \chi \frac{\sqrt{\pi}}{4} \frac{1}{N} \sum_{i=1}^{N} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) \tag{2.9}$$

where the collision parameters $\{\mathbf{c}_{1,i}, \mathbf{c}_{2,i}, \Theta_i\}$ are chosen with a probability $\frac{f_{1,i} f_{2,i} g_i \sigma_i}{\chi}$.

We can think of this sum as considering collision events with a probability proportional to the physical collision rate ($f_1 f_2 g \sigma$). That means that the computational cost associated with accounting for rare collision events (which do not significantly contribute to the collision integral) will be small. This greatly improves the efficiency of evaluating the collision integral, and is a key aspect of particle-based simulation techniques such as DSMC.

2.3 Control variate integration

We can further improve the efficiency of evaluating the collision integral by using control variate integration [18, 27]. We evaluate $\int_{\mathcal{R}} y(\mathbf{r})\, d\mathbf{r}$ by writing

$$\int_{\mathcal{R}} y(\mathbf{r})\, d\mathbf{r} = \int_{\mathcal{R}} z(\mathbf{r})\, d\mathbf{r} + \int_{\mathcal{R}} \left[y(\mathbf{r}) - z(\mathbf{r})\right]\, d\mathbf{r} \tag{2.10}$$

$$\approx \int_{\mathcal{R}} z(\mathbf{r})\, d\mathbf{r} + \frac{1}{N} \sum_{i=1}^{N} \frac{y(\mathbf{r}_i) - z(\mathbf{r}_i)}{p(\mathbf{r}_i)} \tag{2.11}$$

where we assume that z can be integrated analytically (or its value can otherwise be determined efficiently and accurately). We have then performed Monte Carlo integration (using importance sampling) on the remainder $(y - z)$. The uncertainty of this method can thus be expected to scale as

$$\sqrt{\frac{\left\langle \left(\frac{y-z}{p}\right)^2 \right\rangle - \left\langle \frac{y-z}{p} \right\rangle^2}{N}} \tag{2.12}$$

It is clear that this will be preferable if we can find an appropriate z that approximates y well, or such that the difference $(y - z)$ can be approximated[3] well by a function (proportional to) p.

2.3.1 Control variate integration of the Boltzmann collision integral

We observe that low-speed flows are typically well-approximated by an equilibrium, or Maxwell-Boltzmann distribution (1.20). We thus separate the distribution function into an equilibrium and a deviational term

$$f = f^{MB} + f^d \tag{2.13}$$

We note that, in all work presented in this thesis, we do not rely on knowing the "correct" f^{MB}; all analyses hold for an *arbitrary* Maxwell-Boltzmann distribution (though the efficiency of the resulting methods will be affected by the Maxwell-Boltzmann distribution chosen).

If we substitute (2.13) into the expression for the collision integral (1.18), we obtain

$$\left[\frac{df}{dt}\right]_{\text{coll}} = \frac{\sqrt{\pi}}{4} \iiint \left(\delta_1' + \delta_2' - \delta_1 - \delta_2\right) \left(f_1^{MB} f_2^{MB} + f_1^{MB} f_2^d + f_1^d f_2^{MB} + f_1^d f_2^d\right) g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \tag{2.14}$$

We note that the integral involving $f_1^{MB} f_2^{MB}$ is identically zero, as this is the collision integral for an equilibrium distribution. We also note that the integrals involving $f_1^{MB} f_2^d$ and $f_1^d f_2^{MB}$ are equal (formally interchanging $\mathbf{c}_1$ and $\mathbf{c}_2$ yields an equivalent

[3]Note that this method requires us to analytically know both the integral of z and p, where the latter must have an integral of unity.

integral). Thus, we can write

$$\left[\frac{df}{dt}\right]_{\text{coll}} = \frac{\sqrt{\pi}}{4}\iiint(\delta_1' + \delta_2' - \delta_1 - \delta_2)\left(2f_1^d f_2^{MB} + f_1^d f_2^d\right) g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \quad (2.15)$$

We can separate this into two terms, and evaluate each using importance sampling.

$$\begin{aligned}\left[\frac{df}{dt}\right]_{\text{coll}} &= \chi_{MB,d}\frac{\sqrt{\pi}}{2}\iiint(\delta_1' + \delta_2' - \delta_1 - \delta_2)\frac{f_1^d f_2^{MB} g\sigma}{\chi_{MB,d}}\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \\ &\quad + \chi_{d,d}\frac{\sqrt{\pi}}{4}\iiint(\delta_1' + \delta_2' - \delta_1 - \delta_2)\frac{f_1^d f_2^d g\sigma}{\chi_{d,d}}\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \quad (2.16) \\ &= \frac{\chi_{MB,d}}{N_{MB,d}}\frac{\sqrt{\pi}}{2}\sum_{i=1}^{N_{MB,d}}(\delta_{1,i}' + \delta_{2,i}' - \delta_{1,i} - \delta_{2,i})\text{sgn}(f_1^d) \\ &\quad + \frac{\chi_{d,d}}{N_{d,d}}\frac{\sqrt{\pi}}{4}\sum_{i=1}^{N_{d,d}}(\delta_{1,i}' + \delta_{2,i}' - \delta_{1,i} - \delta_{2,i})\text{sgn}(f_1^d)\text{sgn}(f_2^d) \quad (2.17)\end{aligned}$$

Here the collision parameters are chosen from the normalized probability distribution $\frac{|f_1^d| f_2^{MB} g\sigma}{\chi_{MB,d}}$ in the first sum and $\frac{|f_1^d||f_2^d| g\sigma}{\chi_{d,d}}$ in the latter. The constants $\chi_{MB,d}$ and $\chi_{d,d}$ are defined as

$$\chi_{MB,d} \equiv \iiint |f_1^d| f_2^{MB} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \quad (2.18a)$$

$$\chi_{d,d} \equiv \iiint |f_1^d||f_2^d| g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \quad (2.18b)$$

In the limit of small f^d, we expect $\chi_{MB,d}$ to scale with $\|f^d\|$ and $\chi_{MB,d}$ to scale with $\|f^d\|^2$; thus we expect $\chi_{MB,d}$ to be the larger term. From equations (2.16), we can then see that the collision integral is expected scale with $\|f^d\|$. More importantly, from our perspective, is the fact the the statistical uncertainty in evaluating (2.17) (with a fixed number of Monte Carlo samples) will also scale with $\|f^d\|$; in other words we expect a *constant* signal to noise ratio. The physical interpretation of the effectiveness of control variate integration is discussed further in section 4.1.2. In brief, control variate integration neglects a large number of collisions "within" the Maxwell-Boltzmann distribution that have zero *net* effect.

This constant signal to noise ratio was illustrated in [7]; in figure 2.1, we plot

the statistical uncertainty in evaluating the fluid velocity $\mathbf{u}$, normalized by the wall velocity, for Couette flow at a variety of wall speeds. We see that, when using the variance reduction techniques described in this paper, we obtain a constant signal to noise ratio in the output of interest, $\mathbf{u}$, regardless of the flow velocity. The DSMC trend illustrates the rapid growth in the signal to noise ratio as the wall velocity decreases. The DSMC trend was scaled to approximately illustrate the crossover point, at which the current method is more efficient than DSMC.

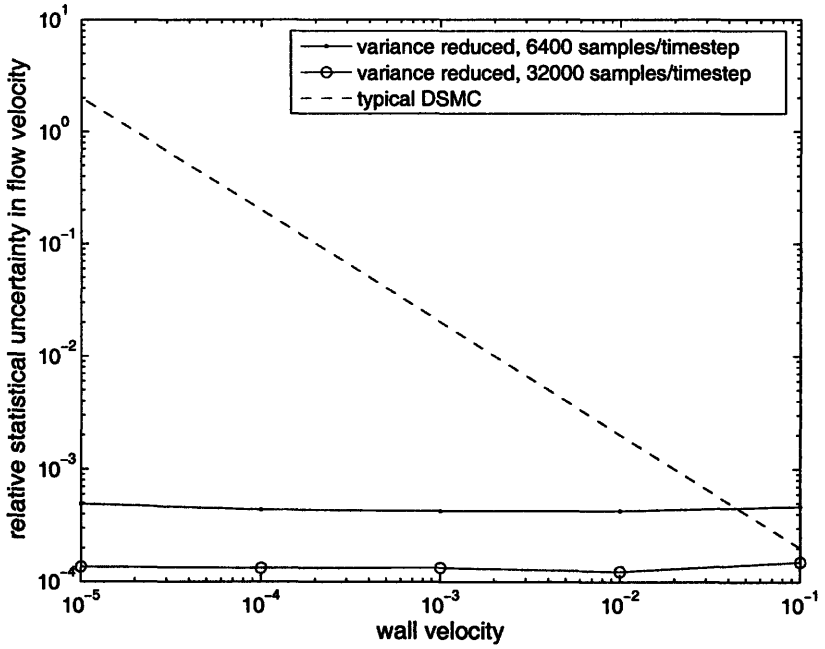

Figure 2.1: Statistical uncertainty in flow velocity (normalized by wall velocity) as a function of dimensionless wall speed using control variate integration and DSMC

Chapter 3

Direct simulation Monte Carlo

In this chapter, we briefly describe the DSMC algorithm, and show how one can interpret it as an implementation of the variance reduction ideas discussed in section 2.2. The aim of this chapter is to illustrate how DSMC utilizes importance sampling to obtain high computational efficiency, as well as to motivate the improvements to DSMC that will be discussed in the next chapter.

3.1 DSMC algorithm

This section gives an short overview of the DSMC algorithm, with an emphasis on the collision process. For a more complete discussion of DSMC, see [12, 2].

DSMC is a particle simulation technique; the state of the computational system is defined by the positions and velocities of the simulated particles, with each particle representing a number ($\mathcal{N}_{\text{eff}}$) of physical molecules. DSMC utilizes a time splitting scheme to simulate the Boltzmann equation; in other words each timestep is split into two parts: an advection step, in which the positions of all particles are updated without modifying the velocities[1], and a collision step, in which the velocities of the particles are modified via the action of simulated collisions.

[1] Assuming the absence of body forces.

The collisionless advection step integrates

$$\frac{\partial f^*}{\partial t^*} + \mathbf{c}^* \cdot \frac{\partial f^*}{\partial \mathbf{x}^*} = 0 \tag{3.1}$$

while the collision step integrates

$$\frac{\partial f^*}{\partial t^*} = \left[\frac{df}{dt}\right]^*_{\text{coll}} \tag{3.2}$$

During the collision step, binary collisions are processed between collision partners chosen at random within the same computational cell. (We note that, with particle methods, it is often more convenient to use dimensional units.)

For a timestep Δt^* with each simulation particle representing $\mathcal{N}_{\text{eff}}$ molecules, the DSMC algorithm can be outlined as follows

1. Collisionless advection step

   - Update the position of all particles: $\mathbf{x}^*(t^* + \Delta t^*) = \mathbf{x}^*(t^*) + \mathbf{c}^*(t^*) \times \Delta t^*$
   - Reflect any particles that collided with the boundaries back into the domain. For boundary conditions such as diffuse walls, this will entail selecting a new velocity from an appropriate probability distribution. See [12, 2] or section 4.1.4 for further details.

2. Sort particles into cells (of volume $\mathcal{V}^*$). Denote the number of particles in a cell by $\mathcal{N}$.

3. Process collisions within each cell

   - Choose $\frac{2\pi}{\mathcal{V}^*}\mathcal{N}^2\mathcal{N}_{\text{eff}}\Delta t^* \left(g^*\sigma^*\right)_{\max}$ pairs of molecules (collision candidates) from each cell. Index each pair by i. Here, and $\left(g^*\sigma^*\right)_{\max}$ is a number chosen to be larger than $\left(g^*\sigma^*\right)_i$ in (almost) all cases.
   - With a probability $\left(g^*\sigma^*\right)_i / \left(g^*\sigma^*\right)_{\max}$, accept the collision by updating each of their velocities to the appropriate post-collision velocity. If the collision is rejected, do nothing and move on to the next pair of candidates.

4. Sample hydrodynamic properties.

We observe that the (expected) total number of collisions accepted in step 3 of the above algorithm is $\frac{2\pi}{\mathcal{V}^*}\mathcal{N}^2\mathcal{N}_{\text{eff}}\,\Delta t^*\,\langle g^*\sigma^*\rangle$. Recalling that each particle represents $\mathcal{N}_{\text{eff}}$ molecules, we see that the effective number of physical collisions is $\frac{2\pi}{\mathcal{V}^*}\left(\mathcal{N}\mathcal{N}_{\text{eff}}\right)^2\,\Delta t^*\,\langle g^*\sigma^*\rangle$ – this matches the number of collisions that would be expected to occur in the corresponding set of physical molecules if there were $\mathcal{N}\mathcal{N}_{\text{eff}}$ molecules in that cell[12].

3.2 Importance sampling in DSMC

The above summary shows that in DSMC collisions occur between particles with a probability proportional to $f_1^* f_2^* g^* \sigma^*$; picking a particle at random from the cell is equivalent to picking with a probability proportional to the distribution function, and collision candidates are accepted with a probability proportional to $g^*\sigma^*$. In other words the collision samples are chosen with a probability proportional to the collision rate for physical molecules. This means that DSMC does not spend excessive time processing rare collision events; this is analogous to the importance sampling techniques described in section 2.2.

Let us take a closer look at equation (2.9), which in dimensional form, can be written

$$\left[\frac{df}{dt}\right]^*_{\text{coll}} \approx \chi^* \frac{1}{2}\frac{1}{N}\sum_{i=1}^{N} \delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i} \tag{3.3}$$

where the set of collision parameters $\left\{\mathbf{c}^*_{1,i}, \mathbf{c}^*_{2,i}, \Theta_i\right\}$ is chosen with a probability $\frac{f^*_{1,i} f^*_{2,i} g^*_i \sigma^*_i}{\chi^*}$ and we have defined

$$\chi^* \equiv \iiint f_1^* f_2^* g^* \sigma^*\, d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \tag{3.4}$$

We recall that the collision integral is (proportional to) the rate of change in the number of particles at a given position in phase space due to the action of collisions. We can see that equation (3.3) evaluates the collision integral this by processing a

set of 'collision events'. For each collision event, the set of collision parameters are chosen with a probability proportional to the physical collision rate (i.e. $f_1^* f_2^* g^* \sigma^*$). Collision events that have one of the pre-collision velocities ($\mathbf{c}_{1,i}$ or $\mathbf{c}_{2,i}$) equal to $\mathbf{c}$ will decrease the calculated value of the collision integral; collision events for which one of the post-collision velocities equals $\mathbf{c}$ will increase the calculated value of the collision integral. It is clear that this collision procedure is analogous to the DSMC algorithm described above.

We will reconsider this interpretation, more carefully and in more detail, in later chapters; the aim of this section was to illustrate how the DSMC collision process utilizes importance sampling. In Chapter 4 we will show how one can incorporate both importance sampling and control variate integration into a particle-based simulation method.

3.3 Limitations of DSMC

DSMC has proven to be an extremely successful method for simulating the dilute gas flows arising in aerospace and other high-speed applications. The strengths of DSMC are numerous: it is highly efficient for high-speed flows, its formulation is straightforward and physically motivated, and it does not require elaborate meshing techniques to simulate complex boundaries. However, DSMC also has significant weaknesses when dealing with the gaseous flows of interest in the present thesis.

First, in DSMC, quantities of interest are obtained by an averaging process; for example the bulk flow velocity is estimated using the mean velocity of all of the particles in a cell. This averaging process leads to a degree of statistical error which is proportional to $N^{-1/2}$, where N is the number of (independent) samples used. However, more critically for low-signal flows, the statistical uncertainty is independent of the magnitude of the signal. This leads to a signal to noise ratio that is inversely proportional to the signal[2]. As a concrete example, when calculating the mean ve-

[2]One can see this by considering Monte Carlo evaluation of the integral (2.14); as $f^d \to 0$, the statistical uncertainty is dominated by the $f_1^{MB} f_2^{MB}$ term, which does not depend on f^d. Thus the level of statistical uncertainty remains constant and the *relative* level of statistical uncertainty

locity of a low-speed gas, for a fixed number of samples the statistical uncertainty in the velocity will be a constant. This is what leads to the *relative* level of statistical uncertainty scaling with $\|\mathbf{u}\|^{-1}$ or, equivalently, the number of samples required to obtain a fixed degree of statistical uncertainty scaling with $\|\mathbf{u}\|^{-2}$ [22].

A second disadvantage of DSMC is that boundary conditions are imposed by re-emitting particles that impact on the boundaries, or by creating particle reservoirs. Both of these require the generation of samples from a (potentially complex) distribution. For simple distributions, this is easily done, however this becomes more difficult when more elaborate boundary conditions need to be imposed.

Finally, DSMC does not directly lend its self to iterative methods for steady state solutions (though other methods, such as the equation-free-framework [1] can be used).

In the next chapter we will discuss how the first limitation, high statistical uncertainty, can be alleviated by incorporating control variate integration into a particle formulation. The second two limitations are inherent to particle approaches; the PDE-based discontinuous Galerkin approach of Chapter 5 does not suffer from these issues.

increases.

Chapter 4

Variance reduced particle method

In this chapter, we develop a particle based simulation method[1] that is analogous to DSMC; however, in contrast to DSMC, we will simulate the *deviation* from equilibrium using a set of particles. Simulating only the deviation from equilibrium, instead of the full distribution function, will lead to a significant computational advantage over DSMC for low-signal (i.e. low-speed) applications.

4.1 Formulation

Our starting point is the (dimensional) form of the collision integral analogous to (2.15)

$$\left[\frac{df}{dt}\right]^*_{\text{coll}} = \frac{1}{2}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\left(2f_1^{d*}f_2^{MB*} + f_1^{d*}f_2^{d*}\right) g^*\sigma^*\, d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \quad (4.1)$$

Recall that we have split the distribution function according to (2.13)

$$f^* = f^{MB*} + f^{d*} \quad (4.2)$$

[1]The work described in this chapter will appear in [11]. We note that [16] describes an independently developed particle scheme that, while significantly different from the present method, shares similar goals.

where f^{MB*} is an *arbitrary* equilibrium distribution and f^{d*} is the deviation from (this particular) equilibrium.

This "variance-reduced" form of the collision integral, (4.1), exhibits reduced statistical uncertainty when evaluated using a Monte Carlo procedure because the integrand, and thus the statistical error resulting from evaluating it via an appropriate Monte Carlo method [32], scale with f^{d*} as $f^{d*} \to 0$; consequently, in this limit, the statistical error decreases linearly with the signal, leading to a constant signal to noise ratio [8]. This result is independently verified for the present particle method in section 4.2.

This variance-reduced form of the collision integral (4.1) is sufficiently general to allow use both in numerical solution methods using standard numerical approaches (as will be discussed in Chapter 5) and particle simulation methods [9]. The latter is the focus of the present chapter: Starting from equation (4.1) we show how one can develop a particle simulation scheme akin to DSMC. In addition to the overall algorithm structure, the method presented here retains a number of DSMC features; in fact, as explained later, DSMC is retained as a special case. The principal difference is that, as suggested by (4.1), we represent the distribution function using the combination of an (arbitrary) underlying equilibrium (Maxwell-Boltzmann) distribution (that can, in general, vary as a function of space and time) and a set of particles representing the deviation of the true distribution function from this equilibrium distribution. This is in contrast to DSMC, in which the entire distribution function is represented using particles.

4.1.1 Simplifying assumptions

In this chapter we will focus on the case where there are no body forces acting on the molecules; extension to case where $\mathbf{F}^* \neq 0$ is straightforward. In the interest of simplicity, for the remainder of this chapter we will assume that the underlying Maxwell-Boltzmann distribution is identical in all spatial cells. In this case the advection step is identical to that of DSMC – the positions of all particles are updated according to their velocities, while the velocities remain constant. The present method

differs from DSMC in processing the collisions and the boundary conditions; these are discussed in more detail below.

4.1.2 Collision algorithm

As in DSMC, collisions are processed in physical cells of volume $\mathcal{V}^*$ with collision partners chosen within the same cell. The collision process, however, differs from DSMC as required by the new form of the collision integral. To derive the collision process we write equation (4.1) in the following form

$$\left[\frac{df}{dt}\right]^*_{\text{coll}} = \iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\, f_1^{d*} f_2^{MB*} g^* \sigma^* d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \tag{4.3}$$

$$+ \frac{1}{2} \iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\, f_1^{d*} f_2^{d*} d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \tag{4.4}$$

and recall that according to the splitting method used here (and in Chapter 3), the collision part of the algorithm integrates equation (3.2), via collisions between simulation particles, by effecting a change equal to

$$\left[\frac{df}{dt}\right]^*_{\text{coll}} \Delta t^* \tag{4.5}$$

onto the (deviational) distribution function (represented by the simulation particles). In what follows we will discuss how equation (4.4) can be interpreted and implemented in terms of inter-particle collisions. A more in-depth discussion is given in section 4.1.3.

DSMC, as discussed in the previous chapter, corresponds to the case $f^{MB*} = 0$, ($f^{d*}_1 = f^*_1 > 0$, $f^{d*}_2 = f^*_2 > 0$). In the more general case, equation (4.4) suggests that there are two distinct contributions to the collision integral: those involving collisions between two deviational particles (corresponding to the term involving $f_1^{d*} f_2^{d*}$) and those involving collisions between a deviational particle and the underlying Maxwell-Boltzmann distribution (corresponding to the term involving $f_1^{d*} f_2^{MB*}$). As in the previous chapter, we will interpret a positive delta function as adding a particle to the distribution function at that point, and a negative delta function as

removing a particle. However, the fact that f^{d*} can be negative in a given region of phase space can affect the sign of the generated particles (this is discussed in significantly more detail below).

In the limit $\mathrm{Ma} \to 0$, the quadratic term (involving $f_1^{d*} f_2^{d*}$) will be negligible (if f^{MB*} is chosen appropriately). However, we retain this term so that the method will remain applicable for all flow conditions[2] and to facilitate comparison with DSMC at $\mathrm{Ma} \sim 0.1$ (the cost of DSMC calculations at the resolution required for the comparisons of the present chapter becomes prohibitive for $\mathrm{Ma} \lesssim 0.1$).

Before we proceed with the implementation, let us repeat that, in general, $f^{d*}(\mathbf{x}^*, \mathbf{c}^*, t)$ may be either positive or negative at any point in phase space. This is a natural consequence of the fact that $f^* = f^{MB*} + f^{d*}$ for an arbitrary f^{MB*}. A negative f^{d*} means that the number of *physical* molecules in the differential phase-space volume element in question is less than that given by the underlying Maxwell-Boltzmann distribution f^{MB*}. In order to allow our particle formulation to capture a negative deviation from equilibrium, we must allow for the possibility of negative deviational particles.

Implementation

In the simulation, each computational particle represents $\pm\mathcal{N}_{\text{eff}}$ ($\mathcal{N}_{\text{eff}} > 0$) physical molecules. We will refer to those simulation particles representing $+\mathcal{N}_{\text{eff}}$ molecules as positive (deviational) particles and those representing $-\mathcal{N}_{\text{eff}}$ molecules as negative (deviational) particles. These particles can be thought of as samples from the *deviational* distribution function, that is,

$$f^{d*} = \frac{\mathcal{N}_{\text{eff}} \left(\langle \mathcal{N}^+ \rangle - \langle \mathcal{N}^- \rangle \right)}{\mathcal{V}^*_{\text{cell}} \, d^3\mathbf{c}^*} \tag{4.6}$$

where $\langle \mathcal{N}^+ \rangle$ and $\langle \mathcal{N}^+ \rangle$ are respectively the expected number of positive and negative particles with a velocity in the range $d^3\mathbf{c}^*$ about $\mathbf{c}^*$ and $\mathcal{V}^*_{\text{cell}}$ is the volume of the cell

[2] As will be seen in section (4.1.2), the computational cost associated with retaining the $(f_1^{d*} f_2^{d*})$ term will be small when this term is small, provided effective cancellation between deviational particles takes place – see below.

in physical space.

From equation (4.6) we can see that a change in the distribution function is equivalent to a change in the (expected) number of particles with a given velocity. Our collision algorithm will update the set of particles (number and distribution) in a manner consistent with the action of the collision integral by using an acceptance-rejection scheme.

Several additional quantities that will be useful in describing the collision algorithm are defined below: let $\mathcal{N}$ be the total number of particles (both positive and negative) in the cell of interest, $n^{MB*} = \int f^{MB*} d^3\mathbf{c}^*$, $\mathcal{R}$ a uniform random number on $[0, 1)$, and $(g^*\sigma^*)_{\max}$ a parameter used as an 'effective ceiling' of $(g^*\sigma^*)$ in the acceptance-rejection scheme (chosen such that the probability that $(g^*\sigma^*) > (g^*\sigma^*)_{\max}$ is negligible).

For a timestep Δt^*, collisions in a physical space cell are performed by the following algorithm:

1. Perform collisions between a deviational particle and a "particle" from the Maxwell-Boltzmann distribution. This step updates the (deviational) distribution function[3] by adding

$$\Delta t^* \iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\, f_1^{d*} f_2^{MB*} g^*\sigma^* d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \quad (4.7)$$

   to its value. This can be achieved by the following acceptance-rejection scheme:

   (a) Select $4\pi\mathcal{N}n^{MB*}\Delta t^* (g^*\sigma^*)_{\max}$ pairs of pre-collision velocities and scattering angles. The velocity $\mathbf{c}_2^*$ is chosen with a (normalized) probability f^{MB*}_2/n^{MB*} and $\mathbf{c}_1^*$ is the velocity of a deviational particle chosen randomly from the cell. The scattering angle Θ is chosen with uniform probability on the unit sphere.

   (b) For each of these potential collision partners, or collision candidates (enumerated by i), if $(g^*\sigma^*)_i / (g^*\sigma^*)_{\max} > \mathcal{R}$ accept the collision by:

[3] See [24] for an extension to the present method that updates f^{MB} in addition to f^d.

i. Setting the velocity of the particle to ${\mathbf{c}'_{1,i}}^*$ where, as before, a prime indicates a post-collision velocity.

ii. *Creating* a particle with velocity $\mathbf{c}^*_{2,i}$ and a sign *opposite* to that of the deviational particle involved in the collision

iii. *Creating* a particle with velocity ${\mathbf{c}'_{2,i}}^*$ and a sign *equal* to that of the deviational particle involved in the collision

2. Perform collisions between two deviational particles. This step updates the distribution function by adding

$$\Delta t^* \frac{1}{2} \iiint \left(\delta'_1 + \delta'_2 - \delta_1 - \delta_2\right) f_1^{d^*} f_2^{d^*} g^* \sigma^* d^2\Theta \, d^3\mathbf{c}_1^* \, d^3\mathbf{c}_2^* \tag{4.8}$$

to its value. This update is performed by the following acceptance-rejection scheme:

(a) Select $\frac{2\pi}{\mathcal{V}^*_{\text{cell}}} \mathcal{N}^2 \mathcal{N}_{\text{eff}} \Delta t^* \left(g^* \sigma^*\right)_{\max}$ pre-collision velocities and scattering angles with $\mathbf{c}_1^*$ and $\mathbf{c}_2^*$ being the velocities of distinct particles chosen at random from the cell. The scattering angle Θ is chosen with uniform probability on the unit sphere.

(b) For each of these potential collision partners (again enumerated by i), if $\left(g^* \sigma^*\right)_i / \left(g^* \sigma^*\right)_{\max} > \mathcal{R}$ accept the collision by:

i. If both particles are positive, updating the velocity of the particles to ${\mathbf{c}'_{1,i}}^*$ and ${\mathbf{c}'_{2,i}}^*$ respectively

ii. If both particles are negative, *creating* a total of four particles: a negative particle at each of $\mathbf{c}^*_{1,i}$ and $\mathbf{c}^*_{2,i}$ and a positive particle at each of ${\mathbf{c}'_{1,i}}^*$ and ${\mathbf{c}'_{2,i}}^*$

iii. If the particle with pre-collision velocity $\mathbf{c}^*_{1,i}$ is negative and that with pre-collision velocity $\mathbf{c}^*_{2,i}$ is positive, setting the velocity of the negative particle to ${\mathbf{c}'_{1,i}}^*$ and *creating* a total of two particles: a positive particle at $\mathbf{c}^*_{2,i}$ and a negative particle at ${\mathbf{c}'_{2,i}}^*$

iv. If the particle with pre-collision velocity $\mathbf{c}^*_{2,i}$ is negative and that with pre-collision velocity $\mathbf{c}^*_{1,i}$ is positive, setting the velocity of the negative particle to ${\mathbf{c}'_{2,i}}^*$ and *creating* a total of two particles: a positive particle at $\mathbf{c}^*_{1,i}$ and a negative particle at ${\mathbf{c}'_{1,i}}^*$

In the present implementation, the (physical space) position of a newly created deviational particle is the same as that of the corresponding deviational particle involved in the collision.

Also, in the case where the number of collision candidates is not an integer, we randomly choose (using the case of particle-Maxwell Boltzmann collisions as an example) either $\lfloor 4\pi\mathcal{N}n^{MB*}\Delta t^* \left(g^*\sigma^*\right)_{\max}\rfloor$ or $\lceil 4\pi\mathcal{N}n^{MB*}\Delta t^* \left(g^*\sigma^*\right)_{\max}\rceil$ collisions such that the expected number of collisions is correct ($\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$ are the floor and ceiling operators, respectively).

In summary, this collision algorithm processes two types of collisions: those between two particles, and those between one particle and the equilibrium distribution. The efficiency of this method comes in part from the collisions we do *not* process, specifically those between a pair of molecules that are both in the equilibrium distribution. Collisions of this type have zero net effect, yet explicitly performing them (as is done with DSMC) could dominate the computation time.

We would like to emphasize that within the framework just described, *any* Maxwell-Boltzmann distribution can be used for f^{MB*}; this method does not depend on knowledge of the "correct" equilibrium distribution, although the choice of the Maxwell-Boltzmann distribution does affect efficiency.

4.1.3 Consistency of the collision algorithm

In this section, we show that (under appropriate conditions) the change in the distribution function as a result of the collision process satisfies $f^*(t^* + \Delta t^*) = f(t) + \Delta t^* \left[\frac{df}{dt}\right]^*_{\text{coll}}$, or equivalently (for time-independent f^{MB*}), $f^{d*}(\mathbf{x}^*, \mathbf{c}^*, t^* + \Delta t^*) = f^{d*}(\mathbf{x}^*, \mathbf{c}^*, t^*) + \Delta t^* \left[\frac{df}{dt}\right]^*_{\text{coll}}$. The purpose of this discussion is to *sketch* a consistency argument between equation (4.4) and the collision algorithm described above,

as well as to provide insight into the present algorithm.

For convenience, we will expand $f^{d*} = f^{d+*} - f^{d-*}$, where positive particles will represent f^{d+*} and negative particles f^{d-*} (we require the functions f^{d+*} and f^{d-*} to be nonnegative). Note that, for a given f^* and f^{MB*}, there exists a unique f^{d*}, but the choice of f^{d+*} and f^{d-*} is not unique; adding the same function to both f^{d+*} and f^{d-*} yields an identical f^{d*}. We also note that in our particle interpretation

$$f^{d+*} = \frac{\mathcal{N}_{\text{eff}} \langle \mathcal{N}^+ \rangle}{\mathcal{V}^*_{\text{cell}} d^3\mathbf{c}^*} \tag{4.9}$$

where $\langle \mathcal{N}^+ \rangle$ is the expected number of molecules in the cell with a velocity in the range $d^3\mathbf{c}^*$. Of course, a similar equation holds for f^{d-*}.

If we substitute this expression for f^* into the collision integral, we obtain

$$\left[\frac{df}{dt}\right]^*_{\text{coll}} = \frac{1}{2}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\left(f_1^{MB*} + f_1^{d+*} - f_1^{d-*}\right) \times \left(f_2^{MB*} + f_2^{d+*} - f_2^{d-*}\right) g^*\sigma^* d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \tag{4.10}$$

This expression can be rearranged to obtain

$$\begin{aligned}\left[\frac{df}{dt}\right]^*_{\text{coll}} = {} & \chi^*_{MB,d+}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\frac{f_1^{d+*} f_2^{MB*} g^*\sigma^*}{\chi^*_{MB,d+}} d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \\ & - \chi^*_{MB,d-}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\frac{f_1^{d-*} f_2^{MB*} g^*\sigma^*}{\chi^*_{MB,d-}} d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \\ & - \chi^*_{d+,d-}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\frac{f_1^{d+*} f_2^{d-*} g^*\sigma^*}{\chi^*_{d+,d-}} d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \\ & + \frac{\chi^*_{d+,d+}}{2}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\frac{f_1^{d+*} f_2^{d+*} g^*\sigma^*}{\chi^*_{d+,d+}} d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \\ & + \frac{\chi^*_{d-,d-}}{2}\iiint (\delta_1' + \delta_2' - \delta_1 - \delta_2)\frac{f_1^{d-*} f_2^{d-*} g^*\sigma^*}{\chi^*_{d-,d-}} d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^*\end{aligned} \tag{4.11}$$

Here, we have defined the number $\chi^*_{\alpha,\beta}$ as

$$\chi^*_{\alpha,\beta} \equiv \iiint f_1^{\alpha*} f_2^{\beta*} g^*\sigma^* d^2\Theta\, d^3\mathbf{c}_1^*\, d^3\mathbf{c}_2^* \tag{4.12}$$

where α and β can each be any of the set $\{d^+, d^-, MB\}$.

We note that in the above integrals,

$$\frac{f_1^{\alpha *} f_2^{\beta *} g^* \sigma^*}{\chi^*_{\alpha,\beta}} \tag{4.13}$$

is a *normalized* distribution function, so the above integrals can be evaluated by importance sampling [32].

Thus, equation (4.11) can be approximately evaluated as

$$\begin{aligned}
\left[\frac{df}{dt}\right]_{\text{coll}} \approx & \frac{\chi^*_{MB,d+}}{\mathrm{N}_{MB,d+}} \sum_{i=1}^{\mathrm{N}_{MB,d+}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) \\
& - \frac{\chi^*_{MB,d-}}{\mathrm{N}_{MB,d-}} \sum_{i=1}^{\mathrm{N}_{MB,d-}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) \\
& - \frac{\chi^*_{d+,d-}}{\mathrm{N}_{d+,d-}} \sum_{i=1}^{\mathrm{N}_{d+,d-}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) \\
& + \frac{\chi^*_{d+,d+}}{2\mathrm{N}_{d+,d+}} \sum_{i=1}^{\mathrm{N}_{d+,d+}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) \\
& + \frac{\chi^*_{d-,d-}}{2\mathrm{N}_{d-,d-}} \sum_{i=1}^{\mathrm{N}_{d-,d-}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right)
\end{aligned} \tag{4.14}$$

Each of the above sums are computed using importance sampling [32], that is, $\mathrm{N}_{\alpha,\beta}$ sets of collision parameters (pre-collision velocities and scattering angle) are chosen with set i having (normalized) probability equal to $\left(f_{1,i}^{\alpha *} f_{2,i}^{\beta *} g_i^* \sigma_i^*\right) / \chi^*_{\alpha,\beta}$. Of course, for the Monte Carlo sums to be accurate, $\mathrm{N}_{\alpha,\beta}$ must be large.

Each of the terms in (4.14) can be seen to fall within one of the two types of collisions described in section 4.1.2 – namely those between two deviational particles and those between a deviational particle and the underlying equilibrium distribution. However, note that collisions involving positive and negative particles need to be treated separately. In the interest of brevity, we will discuss one such term, namely the one representing collisions between the Maxwell Boltzmann distribution and a positive particle. Other cases can be considered using a similar approach.

Maxwell-Boltzmann – particle collisions

In this section we show that collisions between positive particles and the Maxwell Boltzmann distribution correspond to the first sum on the right side of equation (4.14).

In the collision algorithm described in section 4.1.2, we choose $4\pi\mathcal{N}n^{MB*}\Delta t^* \left(g^*\sigma^*\right)_{\max}$ collision candidates of the Maxwell-Boltzmann – particle type. The probability density that a chosen pair will belong to the Maxwell-Boltzmann-positive particle group $(MB, d+)$ and will be characterized by precollision velocities $\mathbf{c}_1^*$ and $\mathbf{c}_2^*$ and scattering angle Θ is

$$\frac{\mathcal{N}^+}{\mathcal{N}} \frac{f_1^{d+*} f_2^{MB*}}{\iiint f_1^{d+*} f_2^{MB*} d^2\Theta d^3\mathbf{c}_1^* d^3\mathbf{c}_2^*} \tag{4.15}$$

The probability that this particular collision will be accepted (assuming $\left(g^*\sigma^*\right)_{\max} \geq g^*\sigma^*$) is

$$\frac{g^*\sigma^*}{\left(g^*\sigma^*\right)_{\max}} \tag{4.16}$$

Thus, the number of $(MB, d+)$ collisions (per unit volume in phase space) with collision parameters $\mathbf{c}_1^*, \mathbf{c}_2^*, \Theta$ that are accepted per unit time (assuming N is large) is

$$N_{MB,d+} = \frac{\mathcal{N}^+}{\mathcal{N}} \frac{f_1^{d+*} f_1^{MB*}}{\iiint f_1^{d+*} f_2^{MB*} d^2\Theta d^3\mathbf{c}_1^* d^3\mathbf{c}_2^*} \times \frac{g^*\sigma^*}{\left(g^*\sigma^*\right)_{\max}} \times 4\pi\mathcal{N}n^{MB*}\left(g^*\sigma^*\right)_{\max} \tag{4.17}$$

Using the fact that

$$\iiint f_1^{d+*} f_2^{MB*} d^2\Theta d^3\mathbf{c}_1^* d^3\mathbf{c}_2^* = n^{MB*} \frac{\mathcal{N}^+ \mathcal{N}_{\text{eff}}}{\mathcal{V}_{\text{cell}}^*} 4\pi \tag{4.18}$$

we obtain

$$N_{MB,d+} = \frac{\mathcal{V}_{\text{cell}}^*}{\mathcal{N}_{\text{eff}}} f_1^{d+*} f_2^{MB*} g^*\sigma^* \tag{4.19}$$

The *total* number of $(MB, d+)$ collisions, $N_{MB,d+}$, accepted per unit time is

$$N_{MB,d+} = \iiint N_{MB,d+} d^2\Theta d^3\mathbf{c}_1^* d^3\mathbf{c}_2^* = \frac{\mathcal{V}_{\text{cell}}^*}{\mathcal{N}_{\text{eff}}} \chi_{MB,d+}^* \tag{4.20}$$

Thus, the probability that an accepted $(MB, d+)$ collision has parameters $\mathbf{c}_1^*$,$\mathbf{c}_2^*$ and Θ is

$$\frac{N_{MB,d+}}{N_{MB,d+}} = \frac{f_1^{d+*} f_2^{MB*} g^* \sigma^*}{\chi^*_{MB,d+}} \tag{4.21}$$

In other words, we are sampling with the probabilities indicated in equation (4.14).

We see that each accepted $(MB, d+)$ collision leads to the addition of a particle at each of the post-collision velocities and the subtraction of a particle at each of the pre-collision velocities. Thus, the collisions of type $(MB, d+)$ will lead to a net change

$$d^3\mathbf{c}^* \sum_{i=1}^{N_{MB,d+}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) \tag{4.22}$$

in the number of *particles* (both positive and negative) with velocity in the range $d^3\mathbf{c}^*$ about $\mathbf{c}^*$ per unit time.

From (4.6), (4.20), and (4.22), we can see that this change in the number of particles leads to a change

$$\frac{\mathcal{N}_{\text{eff}}}{\mathcal{V}^*_{\text{cell}}} \sum_{i=1}^{N_{MB,d+}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) = \frac{\chi^*_{MB,d+}}{N_{MB,d+}} \sum_{i=1}^{N_{MB,d+}} \left(\delta'_{1,i} + \delta'_{2,i} - \delta_{1,i} - \delta_{2,i}\right) \tag{4.23}$$

in the *distribution function* at $\mathbf{c}^*$ per unit time. As the collision parameters are chosen with a probability $f_1^{d+*} f_2^{MB*} g^* \sigma^* / \chi^*_{MB,d+}$, we can see that this is equal to the first sum on the right side of equation (4.14) (in the limit where $\mathcal{N}$, and thus $N_{MB,d+}$, is large).

We make a few observations about this result. First, the number of collisions is always such that in equation (4.14), the weight of each delta function is $\mathcal{N}_{\text{eff}}/\mathcal{V}^*_{\text{cell}}$, which is the effect that a single particle has on the distribution function in equation (4.9). This is of course necessary for a particle interpretation. Second, the collision rate for negative particles, which were introduced for the purpose of computation, is the same as for 'real' particles. Finally, the number of collision partners chosen for particle-particle collisions matches that of DSMC.

Perhaps a more intuitive explanation of the collision process is as follows: Consider

a point in phase space where $f^{d*} > 0$. This means that there are more molecules with that velocity than are accounted for by the Maxwell-Boltzmann distribution. Therefore, there should be more collisions involving these molecules. To correct for this, we subtract from the distribution function at this point in phase space, as well as for the other pre-collision velocity; we also add to the distribution function at the corresponding post-collision velocities. In the opposite case ($f^{d*} < 0$), there are fewer molecules than the Maxwell-Boltzmann distribution indicates; thus, there should be fewer collisions involving molecules at this point in phase space. To correct for this, we must add to the distribution function at the pre-collision velocities and subtract from the distribution function at the post-collision velocities.

4.1.4 Boundary conditions

In this work, we will focus on the diffuse wall boundary condition. This boundary condition requires that the net mass flux at the wall equal zero and the distribution function for particles leaving the wall be at equilibrium with the wall. In our case, this is equivalent to

$$\int_{\mathbf{c}^* \cdot \tilde{n} < 0} (\mathbf{c}^* \cdot \tilde{n}) \left(f^{MB*} + f^{d*} \right) d^3\mathbf{c}^* = \int_{\mathbf{c}^* \cdot \tilde{n} > 0} (\mathbf{c}^* \cdot \tilde{n}) \left(f^{MB*} + f^{d*} \right) d^3\mathbf{c}^* \tag{4.24}$$

$$f^{MB*} + f^{d*} \propto f^{\text{wall}*} \text{ for } \mathbf{c}^* \cdot \tilde{n} > 0 \tag{4.25}$$

where $f^{\text{wall}*}$ is a distribution (with an arbitrary number density) at equilibrium with the wall and $\tilde{n}$ is a unit normal at the boundary pointing into the gas.

The mass flux incident upon the wall is due to two sources: deviational particles colliding with the wall and the flux of particles due to the underlying Maxwell-Boltzmann distribution. Our boundary condition algorithm will consider these sources separately. In the case of particles colliding with the wall, the algorithm is much like that of DSMC, except pairs consisting of a positive and negative particle can be cancelled because it is only the *net* mass flux that is of interest. The effect of the underlying Maxwell-Boltzmann distribution is more subtle; the presence of this dis-

tribution implies both a molecular flux incident upon the wall, and a molecular flux leaving the wall. Thus, deviational particles need to be *created* according to the distribution given by the difference in molecular fluxes between $f^{\text{wall}*}$ and f^{MB*}, that is

$$(\mathbf{c}^* \cdot \tilde{n}) \left(\beta f^{\text{wall}*} - f^{MB*} \right) \tag{4.26}$$

Here,

$$\beta \equiv \frac{\int_{\mathbf{c}^* \cdot \tilde{n} < 0} (\mathbf{c}^* \cdot \tilde{n}) \, f^{MB*} d^3\mathbf{c}^*}{\int_{\mathbf{c}^* \cdot \tilde{n} > 0} (\mathbf{c}^* \cdot \tilde{n}) \, f^{\text{wall}*} d^3\mathbf{c}^*} \left(= \frac{n^{MB*}\sqrt{T^{MB*}}}{n^{\text{wall}*}\sqrt{T^{\text{wall}*}}} \right) \tag{4.27}$$

where n^* is the number density and T^* the temperature. The second equality holds if the wall distribution is a Maxwell-Boltzmann and both $f^{\text{wall}*}$ and f^{MB*} have zero velocity in the normal direction. The parameter β is necessary to ensure conservation of mass; it is a consequence of allowing $f^{\text{wall}*}$ to have an arbitrary number density. Generating particles according to equation (4.26) is accomplished by using an acceptance-rejection scheme in the present implementation.

Let $\mathcal{N}_{\text{wall}}^+$ and $\mathcal{N}_{\text{wall}}^-$ be the number of positive and negative particles, respectively, that collided with the wall in the timestep of interest. We will also let $\mathcal{V}_{\mathbf{c}}^*$ be a large (but finite) volume in velocity space, A^* be the cross sectional area of the boundary, $\mathcal{R}$ be a uniform variate on $[0, 1)$, D_{max}^* be a parameter used in the acceptance rejection scheme, chosen to be greater than $\left| (\mathbf{c}^* \cdot \tilde{n}) \left(\beta f^{\text{wall}*} - f^{MB*} \right) \right|$ for (almost) all $\mathbf{c}^*$ in $\mathcal{V}_{\mathbf{c}}^*$.

In this case, the boundary condition algorithm for diffuse walls can be implemented as follows

1. Process particle-wall collisions

   (a) Remove $\min\left(\mathcal{N}_{\text{wall}}^+, \mathcal{N}_{\text{wall}}^-\right)$ pairs consisting of a positive and negative particle. Reflect the remaining particles (which will be either all positive or all negative) with a velocity distribution proportional to $(\mathbf{c}^* \cdot \tilde{n}) \, f^{\text{wall}*}$ (as is done in DSMC).

2. Generate particles due to the difference between $f^{\text{wall}*}$ and f^{MB*}

   (a) Generate $\frac{\Delta t^* A^* \mathcal{V}_{\mathbf{c}}^* D_{\text{max}}^*}{\mathcal{N}_{\text{eff}}}$ trial velocities in the volume $\mathcal{V}_{\mathbf{c}}^*$ with $\mathbf{c}^* \cdot \tilde{n} \geq 0$. Each

of these trial velocities is indexed by i.

(b) If $|c^*_{x,i}\left(\beta f_i^{\text{wall}*} - f^{MB^*}_i\right)| > \mathcal{R}_i D^*_{\max}$, generate a particle at the wall with a sign equal to $\text{sgn}\left(c^*_{x,i}\left[\beta f_i^{\text{wall}*} - f^{MB^*}_i\right]\right)$

(c) Finally, the positions of all particles that were created are updated for a random fraction of a timestep.

If we allow the underlying Maxwell-Boltzmann distribution to vary between adjoining cells, a procedure analogous to the above would need to be used in the advection step to ensure molecular flux conservation across cell boundaries.

4.1.5 Particle removal

Both the collision process and boundary conditions described above involve the creation of particles (unless $f^{MB^*} = 0$ and the inital state does not include negative particles, in which case the DSMC algorithm is obtained). For the method to remain practical, one must have a means to remove excess particles and thus avoid rapid growth in their number. To some extent this is done by the boundary conditions; the diffuse wall boundary conditions as described above can reduce the total number of particles in the system due to the cancellation of positive and negative particles. This removal is sufficient in cases where collisions with the boundaries are relatively frequent compared to collisions between molecules, i.e. when the Knudsen number is large. For smaller Knudsen numbers an additional technique is necessary. Similar challenges have been reported in [16].

We observe that a pair consisting of a positive and negative particle at the same position in physical space and with the same velocity has no effect on the system; this pair can be removed with no effect on the distribution function. Of course, in the present method such a pair has essentially zero probability of occurring. Motivated by this, in the present work we cancel pairs consisting of a positive and negative particle that are in the same cell in physical space and for which the relative speed of the molecules is less than a threshold value, termed the cancellation radius. This introduces additional discretization into the present algorithm with its associated

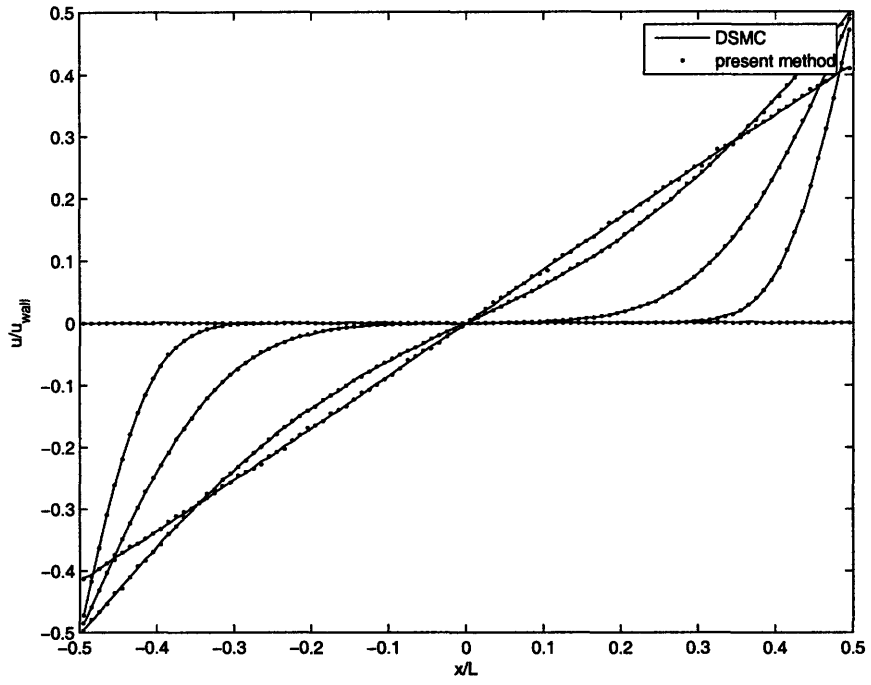

Figure 4.1: Velocity profiles for Kn = 10 (snapshots at $t^* = 0$, 0.011, 0.023, 0.045 and 0.12 $\bar{t}^*$)

potential for numerical error. The effect of this cancellation approach on the accuracy of the overall method will be discussed below. A method that is derived from the present formulation, but does not require a cancellation routine has been developed and is presented in [24]

4.2 Comparison to DSMC simulations

In this section we present a comparison between DSMC solutions and solutions obtained using the proposed particle scheme for the impulsively started shear flow problem. In this problem, the gas is initially at rest and bounded by two parallel, infinite and fully accommodating walls. At time $t = 0$, the two walls are impulsively accelerated to a velocity $u^{\text{wall}*} = \pm 10^{-1}\bar{c}^*$, where $\bar{c}^*$ is the most probable thermal speed. The initial temperature of the gas (and walls) is $T^{\text{wall}*} = 300\text{K}$, and the gas properties used were a density of $\rho^* = 1.78\text{kg/m}^3$, a molecular mass of $m^* = 6.63 \times 10^{-26}\text{kg}$ and a hard-sphere diameter of $d^* = 3.66 \times 10^{-10}\text{m}$.

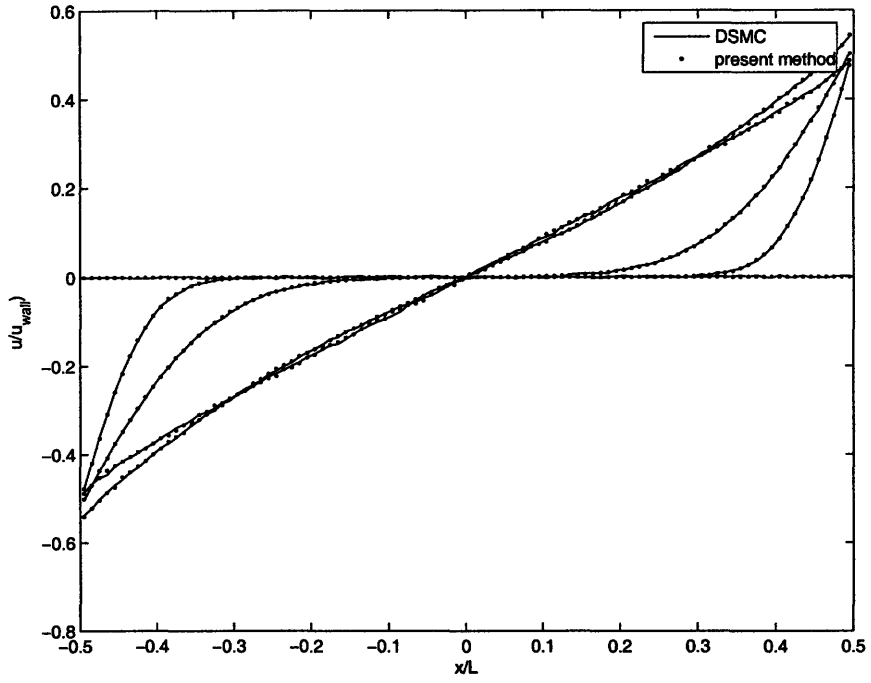

Figure 4.2: Velocity profiles for Kn = 1 (snapshots at $t^* = 0$, 0.11, 0.23 and 0.56 and 2.3 $\bar{t}^*$)

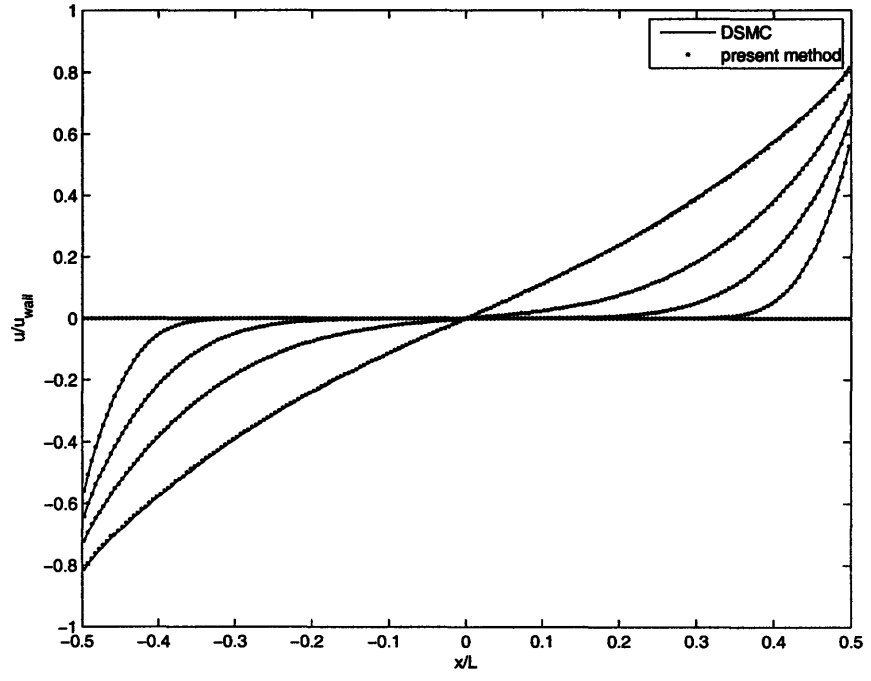

Figure 4.3: Velocity profiles for Kn = 10^{-1} (snapshots at $t^* = 0$, 1.1, 2.5, 5.4 and 14 $\bar{t}^*$)

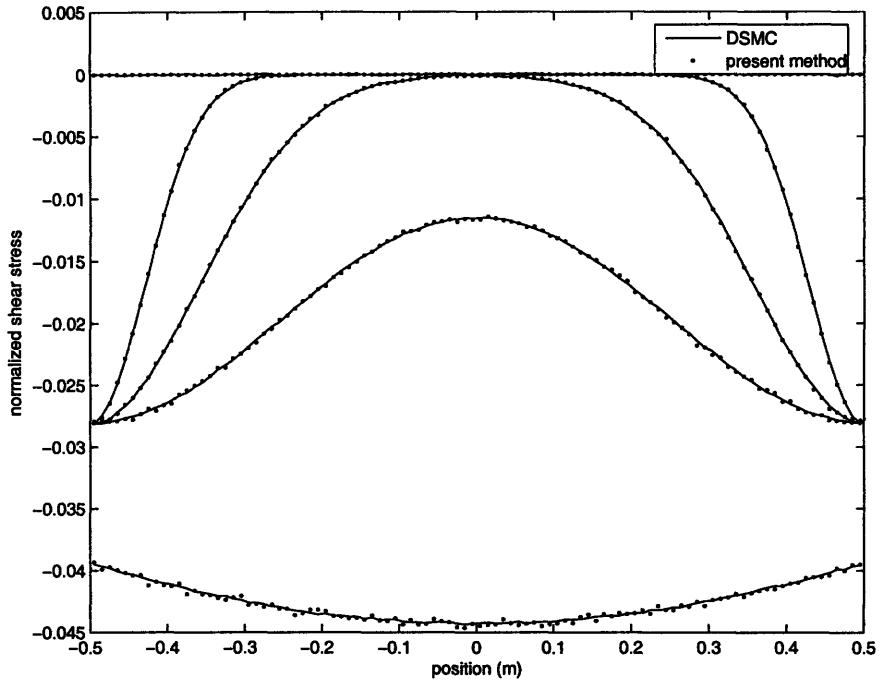

Figure 4.4: Shear stress for Kn = 10 (snapshots at $t^* = 0$, 0.011, 0.023, 0.045 and 0.12 $\bar{t}^*$)

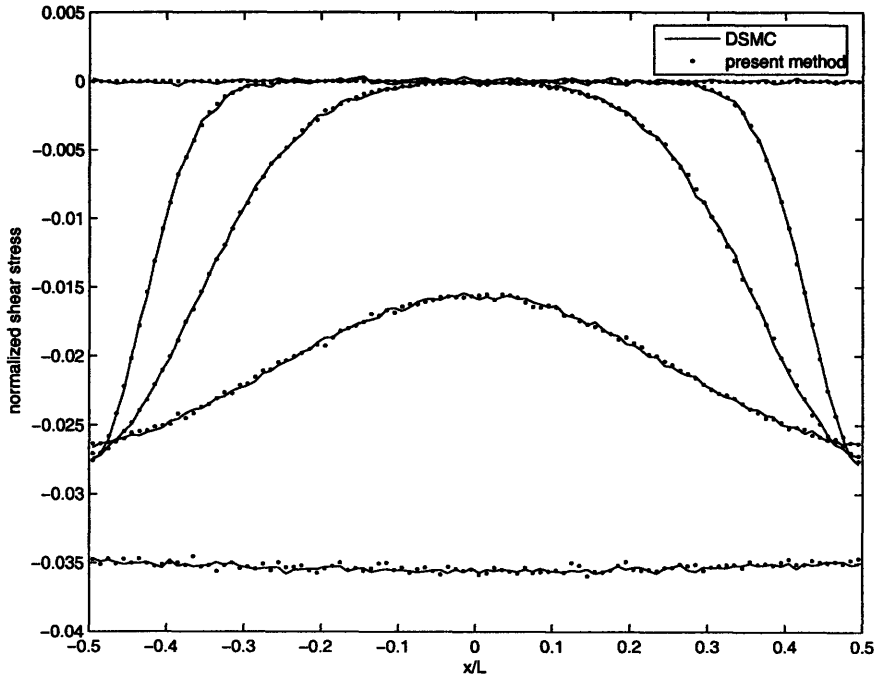

Figure 4.5: Shear stress for Kn = 1 (snapshots at $t^* = 0$, 0.11, 0.23 and 0.56 and 2.3 $\bar{t}^*$)

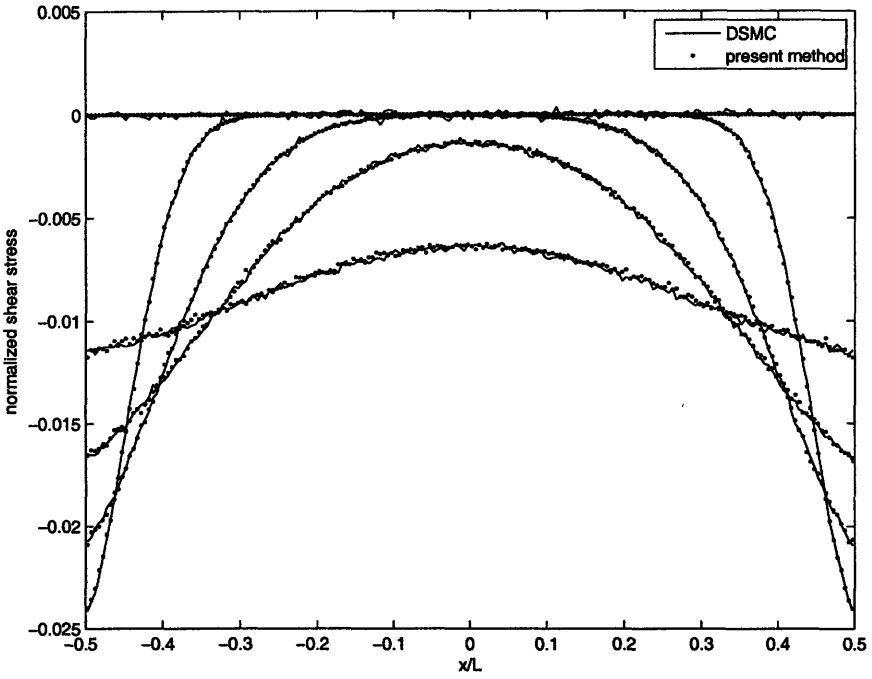

Figure 4.6: Shear stress for Kn = 10^{-1} (snapshots at $t^* = 0$, 1.1, 2.5, 5.4 and 14 $\bar{t}^*$)

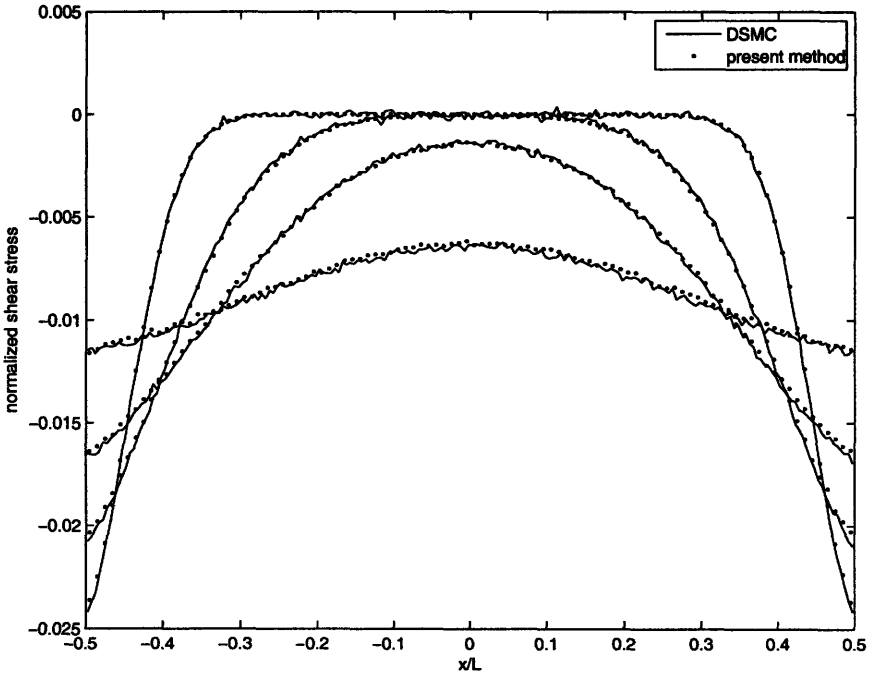

Figure 4.7: Shear stress for Kn = 10^{-1} (snapshots at $t^* = 0$, 1.1, 2.5, 5.4 and 14 $\bar{t}^*$). This figure illustrates the use of a larger cancellation radius than used for figure 4.6

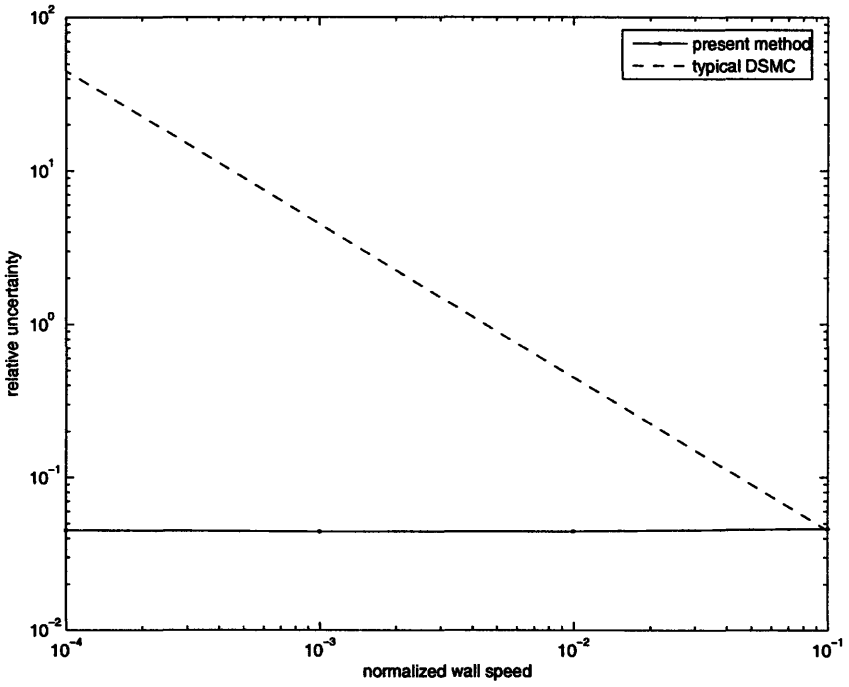

Figure 4.8: Scaling of relative statistical uncertainty in normalized flow speed as a function of normalized wall speed

We note that this wall speed was chosen to make the DSMC calculation feasible; the present method is efficient for arbitrarily low wall speeds. We also note that the present test problem is in fact a stringent test of the accuracy of the proposed method (particularly the cancellation routine) due to its impulsive nature and the associated discontinuities in the distribution function originating at the wall at $t^* = 0$ and propagating into the physical domain for $t^* > 0$. This will be further discussed in the next section.

Calculations were performed at Knudsen numbers of 10^{-1}, 1 and 10. A cancellation radius of about $0.042\bar{c}^*$ (15m/s) was used for the Kn $= 10^{-1}$, and Kn $= 1$ calculations, while no cancellation (other than that due to the boundary conditions) was necessary for the Kn $= 10$ calculation. A total of 100 cells were used in physical space for Kn $= 10$ and Kn $= 1$, while 200 cells were used for Kn $= 0.1$ calculations. Times are reported in units of the collision time, $\bar{t}^*$. Although a large number of calculations have been performed, here we present representative results. Insight gained from other calculations will be discussed in the next section.

In all calculations shown here, we have chosen f^{MB*} to be a spatially homogeneous Maxwell Boltzmann distribution with a mean velocity of zero and a number density equal to the bulk number density. Choosing f^{MB*} to vary in physical space in a manner that reduces the magnitude of f^{d*} would be expected to increase the efficiency of the method; in the interest of simplicity, this is not implemented here.

Figures 4.1, 4.2 and 4.3 compare the normalized velocity profiles computed with the proposed approach to DSMC results for Kn = 10, Kn = 1 and Kn = 10^{-1}, respectively. The agreement is excellent in all cases.

We also compare the shear stress profiles (normalized by $\rho^*\bar{c}^{*2}$) between DSMC and the present method in figures 4.4, 4.5 and 4.6. There is excellent agreement between the current method and DSMC for Kn = 10 and Kn = 1, but a small degree of error is apparent for smaller Knudsen numbers. This is due to the various forms of discretization involved in our solution; finer discretization or a more refined cancellation algorithm reduces the discrepancy. Figure 4.7 illustrates the effect of a larger cancellation radius (about $0.071\bar{c}^*$, or 25m/s) on the shear stress; note that significantly more error is present compared to figure 4.6.

The most important advantage of the proposed method is that its relative statistical uncertainty remains (approximately) constant as flow speeds decrease (for a fixed computational cost), in sharp contrast to DSMC. This is illustrated in figure 4.8, which shows the relative statistical uncertainty (specifically the maximum statistical uncertainty over all cells, normalized by wall velocity) versus the wall velocity (normalized by $\bar{c}^*$) with wall velocities ranging from $10^{-4}\bar{c}^*$ to $10^{-1}\bar{c}^*$ and a constant computational cost. The Knudsen number for the test cases was 10^{-1} and the cancellation radius was about $0.071\bar{c}^*$ (25m/s), and the DSMC trend is from [22].

Figure 4.8 shows that below a certain flow speed, the present method will have a lower computational cost to obtain a given level of statistical uncertainty compared to DSMC, with the relative advantage rising sharply with decreasing flow speeds – specifically as $(u^{\text{wall }*})^{-2}$. Note that figure 4.8 is only intended to illustrate this trend; the actual location of the crossover point has not been determined (and will be implementation dependent), though preliminary work places it at a wall speed on

the order of $10^{-1}\bar{c}^*$ (about 35m/s at room temperature).

4.3 Discussion

We have presented and extensively tested a particle simulation scheme for solving the Boltzmann equation which incorporates the variance reduction approach presented in [8].

The primary challenge associated with this method is the particle cancellation necessary to ensure a manageable number of simulation particles. Our numerical results suggest that the method described here, in its present form, is very effective for $\mathrm{Kn} \gtrsim 1$, whereas for $\mathrm{Kn} \lesssim 1$ the direct numerical discretization of the Boltzmann equation described in [8] is preferable. The primary reason is that, while one of the main advantages of particle formulations such as DSMC is that they do not require a direct velocity space discretization (thus eliminating the associated numerical error and storage requirements), the present particle cancellation scheme effectively introduces a velocity space discretization.

Despite this, as shown in the previous section, the numerical error can be kept to acceptable levels with a small cancellation radius. In fact, in the stringent[4] test-problem of section 4.2, which involves propagating discontinuities, error was discernible only in the higher moments (shear stress) of the distribution function.

Moreover, the approach for removing particles described in this chapter is perhaps the simplest of a number of possible approaches. A more sophisticated particle cancellation scheme may extend the excellent computational efficiency of this scheme for $\mathrm{Kn} > 1$ to the whole range of Knudsen numbers of interest ($0.1 \lesssim \mathrm{Kn} \lesssim 10$). A further development of the current algorithm which removes the need for particle cancellation is described in [24, 23].

We conclude by emphasizing that the new method exhibits a statistical uncertainty that does not depend on flow speed in the low Mach number limit and is thus ideal for

[4]The particle cancellation method used here is expected to smooth-out discontinuities in the distribution function; thus problems involving discontinuities present perhaps the most demanding test of this method.

use in low-speed flows where low-signal-to-noise ratio presents the biggest challenge, and in fact makes DSMC calculations intractable. Although our verification was limited to one-dimensional flows in this chapter, the method is sufficiently general to be applied directly to other flow configurations and to problems in a higher number of physical dimensions.

Chapter 5

Variance reduced discontinuous Galerkin method

In this chapter, we present an alternative to the particle simulation method discussed in Chapter 4. Instead, we will incorporate the variance reduction ideas of Chapter 2 into a discontinuous Galerkin (DG)[17] framework[1]. Within this formulation, the Boltzmann equation is discretized in physical and velocity space with a (discontinuous) finite element approach, while the time integration is performed using a Runge Kutta integration scheme. This combination is known as Runge Kutta discontinuous Galerkin (RKDG)[17]. In the present RKDG implementation, variance reduction techniques are used to provide an efficient means of evaluating the collision integral term.

5.1 Formulation

As before, we separate the distribution function into equilibrium and deviational parts

$$f = f^{MB} + f^{d} \qquad [2.13]$$

[1]The work presented in this chapter has been submitted to *Physics of Fluids* [10].

where f^{MB} is an *arbitrary* equilibrium (Maxwell-Boltzmann) distribution. For simplicity, in this chapter we take f^{MB} to be only a function of velocity space – that is f^{MB} is chosen to be independent of physical space and time[2]. Using this definition and equation (1.16), we can write

$$\frac{\partial f^d}{\partial t} + \frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial f^d}{\partial \mathbf{x}} + \mathbf{a}\cdot\frac{\partial f^d}{\partial \mathbf{c}} = \left[\frac{df}{dt}\right]_{coll} - \mathbf{a}\cdot\frac{\partial f^{MB}}{\partial \mathbf{c}} \tag{5.1}$$

To obtain the RKDG formulation of this equation, we closely follow the approach laid out in [17]. We will suppose our computational domain is divided into elements, denoted by Ω, in physical and velocity space. We require that for *any*[3] test function $v(\mathbf{x},\mathbf{c})$ defined to be zero outside of the element Ω

$$\int_\Omega v\left(\frac{\partial f^d}{\partial t} + \frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial f^d}{\partial \mathbf{x}} + \mathbf{a}\cdot\frac{\partial f^d}{\partial \mathbf{c}}\right) d^6\Omega = \int_\Omega v\left(\left[\frac{df}{dt}\right]_{coll} - \mathbf{a}\cdot\frac{\partial f^{MB}}{\partial \mathbf{c}}\right) d^6\Omega \tag{5.2}$$

where the integrals extend over the element Ω. We note that as v is defined to be zero outside of Ω, a similar equation also holds when the integrals extend over any part of the entire domain.

Noting that the Boltzmann equation is a conservation law in 6-dimensional phase space, we proceed with integration by parts to obtain

$$\int_\Omega v\frac{\partial f^d}{\partial t}\, d^6\Omega + \int_\Gamma v f^d\left(\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\mathbf{n}_\mathbf{x} + \mathbf{a}\cdot\mathbf{n}_\mathbf{c}\right) d^5\Gamma - \int_\Omega f^d\left(\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial v}{\partial \mathbf{x}} - \mathbf{a}\cdot\frac{\partial v}{\partial \mathbf{c}}\right) d^6\Omega = \int_\Omega v\left(\left[\frac{df}{dt}\right]_{coll} - \mathbf{a}\cdot\frac{\partial f^{MB}}{\partial \mathbf{c}}\right) d^6\Omega \tag{5.3}$$

Here, $\mathbf{n}_\mathbf{x}$ and $\mathbf{n}_\mathbf{c}$ are, respectively, outward-normal vectors of the element in physical and velocity space and $|\mathbf{n}_\mathbf{x}|^2 + |\mathbf{n}_\mathbf{c}|^2 = 1$; Γ denotes the 5 dimensional surface of the element Ω in phase space.

In the present work, our interest lies in cases where body forces are negligible or

[2] A more involved approach might allow f^{MB} to vary so as to reduce the magnitude of f^d.
[3] Provided integrability requirements are met, see [17].

not present; we will thus set $\mathbf{a} = 0$ to obtain

$$\int_\Omega v\frac{\partial f^d}{\partial t}\,d^6\Omega + \int_\Gamma vh\,d^5\Gamma - \int_\Omega f^d\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial v}{\partial \mathbf{x}}\,d^6\Omega = \int_\Omega v\left[\frac{df}{dt}\right]_{coll} d^6\Omega \tag{5.4}$$

where we have defined h to be a flux function[4] that approximates $f^d \times \left(\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\mathbf{n_x} + \mathbf{a}\cdot\mathbf{n_c}\right)$.

We expand f^d (specifically, f^d within a single element) and v in terms of our (now taken to be finite[5]) set of basis functions $\phi_i(\mathbf{x},\mathbf{c})$.

$$f^d = \sum_i \hat{f}_i^d(t)\phi_i(\mathbf{x},\mathbf{c}) \tag{5.5}$$

$$v = \sum_j \hat{v}_j\phi_j(\mathbf{x},\mathbf{c}) \tag{5.6}$$

The functions ϕ_i are defined to be nonzero only within a single element (see appendix A for a description of the shape functions used in the present work). No continuity requirements exist between elements, allowing the formulation to capture discontinuities in the solution. Substituting these expressions into (5.4) and requiring the equation to hold for any set of coefficients $\hat{v}_j$ (i.e. any $\hat{v}$), we obtain

$$\sum_i \frac{\partial \hat{f}^d_i}{\partial t}\int_\Omega \phi_i\phi_j\,d^6\Omega + \int_\Gamma \phi_j h\,d^5\Gamma - \frac{\sqrt{\pi}}{2}\sum_i \hat{f}_i^d\int_\Omega \phi_i\mathbf{c}\cdot\frac{\partial\phi_j}{\partial\mathbf{x}}\,d^6\Omega = \int_\Omega \phi_j\left[\frac{df}{dt}\right]_{coll} d^6\Omega \tag{5.7}$$

which must hold for all j. Equation (5.7) defines a linear system of equations for $\frac{\partial \hat{f}_i^d}{\partial t}$. We also note that, because the boundary conditions are weakly imposed on an element, we can solve for $\frac{\partial \hat{f}_i^d}{\partial t}$ independently within each element; in other words if there are N_e total elements and N_s shape functions per element, one needs to solve N_e $N_s \times N_s$ systems instead of one $(N_eN_s)\times(N_eN_s)$ system. Integrating $\frac{\partial \hat{f}_i^d}{\partial t}$ in time using a (strong stability preserving [17]) Runge Kutta method gives the time evolution of $\hat{f}_i^d$, and thus $\hat{f}^d$.

All integrals on the left side of (5.7) are evaluated using Gaussian quadrature

[4]This will simply be an upwind flux as our convection term is linear.

[5]To simplify the notation, we do not distinguish between the deviational distribution function and its discretized form.

using standard methods [26]. In particular, this work utilizes tensor product shape functions so the sum-factorization technique can be used to greatly speed evaluation of the relevent integrals.

The shape functions used in this work are tensor products of Legendre polynomials. However, tensor product bases have more components than strictly needed to represent a polynomial function of given order. This fact will impact the evaluation of the collision integral; as mentioned in appendix B, the current method used to evaluate the collision integral required the evaluation of distribution functions (and thus the constituent shape functions) at a set of random points. This step in particular could be sped by using a smaller set of shape functions.

5.1.1 Collision integral

To evaluate the term involving the collision integral in equation (5.7) we will use a Monte Carlo integration technique which adapts the variance reduction ideas proposed in [8] to the present DG formulation. Specifically, the variance reduction arises from considering only the deviational distribution function f^d [equation (2.13)] and the use of importance sampling (see below).

Let us define

$$\mathbb{I}(v) \equiv \int_\Omega v \left[\frac{df}{dt}\right]_{coll} d^6\Omega = \iint v \left[\frac{df}{dt}\right]_{coll} d^3\mathbf{c}\, d^3\mathbf{x} \tag{5.8}$$

The second equality follows by using the fact that v is taken to be zero outside the element in question and thus integration over both $\mathbf{x}$ and $\mathbf{c}$ can extend over all space. Using the properties of the collision operator [37, 14], and the fact that integration extends over all velocity space we can write

$$\mathbb{I}(v) = \frac{\sqrt{\pi}}{4} \iiiint (v_1' + v_2' - v_1 - v_2)\, f_1\, f_2\, g\, \sigma\; d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \tag{5.9}$$

where $v_1 = v(\mathbf{x}, \mathbf{c}_1)$, $v_2 = v(\mathbf{x}, \mathbf{c}_2)$, $v_2' = v(\mathbf{x}, \mathbf{c}_2')$ and $v_1' = v(\mathbf{x}, \mathbf{c}_1')$.

Using expression (2.13), noting that the collision integral for a Maxwell Boltzmann

distribution is identically zero, and taking advantage of the symmetry in the collision operator [14], we can write

$$\mathbb{I}(v) = \frac{\sqrt{\pi}}{2}\chi_{MB,d}\iiiint (v_1' + v_2' - v_1 - v_2)\frac{f_1^d f_2^{MB}}{\chi_{MB,d}} g\sigma \; d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 d^3\mathbf{x}$$
$$+ \frac{\sqrt{\pi}}{4}\chi_{d,d}\iiiint (v_1' + v_2' - v_1 - v_2)\frac{f_1^d f_2^d}{\chi_{d,d}} g\sigma \; d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 d^3\mathbf{x} \qquad (5.10)$$

where we have defined the constants

$$\chi_{MB,d} \equiv \iiiint |f_1^d| f_2^{MB} d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} = 4\pi \left(\int f_2^{MB} d^3\mathbf{c}_2\right) \iint |f_1^d| d^3\mathbf{c}_1 d^3\mathbf{x} \qquad (5.11)$$

$$\chi_{d,d} \equiv \iiiint |f_1^d||f_2^d| d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} = 4\pi \int \left\{\left(\int |f_1^d| d^3\mathbf{c}_1\right)\left(\int |f_2^d| d^3\mathbf{c}_2\right)\right\} d^3\mathbf{x} \qquad (5.12)$$

with 4π being the surface area of the unit sphere.

Noting that $f_1^d f_2^d/\chi_{d,d}$ and $f_1^d f_2^{MB}/\chi_{MB,d}$ in equation (5.10) are *normalized* probability distribution functions, we can perform importance sampling[6][32], obtaining

$$\mathbb{I}(v) \approx \frac{\sqrt{\pi}}{2}\chi_{MB,d}\frac{1}{N_{MB}}\sum_{i=1}^{N_{MB}} \left[v_{1,i}' + v_{2,i}' - v_{1,i} - v_{2,i}\right] g_i\sigma_i \mathrm{sgn}(f_{1,i}^d)$$
$$+ \frac{\sqrt{\pi}}{4}\chi_{d,d}\frac{1}{N_d}\sum_{i=1}^{N_d} \left[v_{1,i}' + v_{2,i}' - v_{1,i} - v_{2,i}\right] g_i\sigma_i \mathrm{sgn}(f_{1,i}^d)\mathrm{sgn}(f_{2,i}^d) \qquad (5.13)$$

where the set of collision parameters $\{\mathbf{x}_i, \mathbf{c}_{1,i}, \mathbf{c}_{2,i}, \Theta_i\}$ are chosen with a probability $|f_{1,i}^d| f_{2,i}^{MB}/\chi_{MB,d}$ in the first sum and $|f_{1,i}^d||f_{2,i}^d|/\chi_{d,d}$ in the second (this implies Θ is chosen with a uniform probability on the unit sphere in each case).

To implement (5.7), we must evaluate (the discretized version of) $\mathbb{I}(\phi_j)$ for all shape functions in all elements. To do this more efficiently, we note that the cost of evaluating (5.13) (and thus the cost of this DG approach) can be reduced significantly because this equation can be evaluated for *all* shape functions in *all* elements using the *same* set of samples, namely the N_{MB} collision events in the first term and N_d

[6]Note that here, as opposed to in Chapters 2 – 4, we have *not* included $g\sigma$ in the importance sampling scheme.

collision events in the second term of equation (5.13). This is achieved by updating the sum only for the cells containing pre- or post-collision velocities for each collision event. This is possible because the shape functions are zero outside their associated element, and thus only the elements containing a pre- or post-collision velocity for a given collision event will be affected by that collision event. When f^d is small, $\chi_{MB,d} \gg \chi_{d,d}$; in such cases, for improved computational efficiency, one could choose N_{MB} to be much larger than N_d.

We also note that the above derivation holds for an arbitrary f^{MB} ("underlying Maxwell-Boltzmann distribution"). The choice of f^{MB} does not affect the accuracy of the method (no approximation has been made), only its efficiency: in general, the smaller the magnitude of the resulting f^d, the greater the degree of variance reduction and the higher the efficiency.

5.1.2 Boundary conditions

In this formulation, boundary conditions are imposed by specifying the (upwind) numerical fluxes at the walls. In this work we used diffuse wall boundary conditions [14, 34]; this model is by far the most widely used [21, 34], primarily because it appears to capture the behavior of engineering surfaces of practical interest quite well.

The diffuse wall boundary condition can be expressed as

$$f^{MB} + f^d = n^{\text{wall}} f^{\text{wall}} \text{ for } \mathbf{c} \cdot \tilde{n} > 0 \tag{5.14}$$

where f^{wall} is a (normalized) distribution at equilibrium with the wall, $\tilde{n}$ is the unit normal pointing into the fluid, and the constant n^{wall} is determined by the mass conservation requirement

$$\int_{\mathbf{c}\cdot\tilde{n}<0} (\mathbf{c}\cdot\tilde{n}) \left(f^{MB} + f^d\right) d^3\mathbf{c} = \int_{\mathbf{c}\cdot\tilde{n}>0} (\mathbf{c}\cdot\tilde{n}) \left(n^{\text{wall}} f^{\text{wall}}\right) d^3\mathbf{c} \tag{5.15}$$

In our implementation, we took advantage of the fact that for the low-speed, isothermal flows presented here, $n^{\text{wall}} = n$, and did not fully implement (5.15). For flows

were $n^{\text{wall}} \neq n$, implementation of (5.15) is straightforward.

Imposition of other typical boundary conditions is also straightforward. For boundary conditions that are more complex than diffuse or specular walls, application of boundary conditions is an area where PDE-based approaches arguably have an advantage over particle-based approaches; in both cases a fluxal quantity is required, however, in particle approaches, random *samples* from this distribution must be generated, while here only the numerical value of the flux at the Gaussian quadrature points is required.

5.1.3 Collision integral implementation details

The implementation of the advection terms follows standard approaches [17]. We will thus focus on the numerical evaluation of the collision integral using equation (5.13).

To simplify the implementation, we will assume that our mesh is a "tensor product" of an (arbitrary) mesh in physical space and an (arbitrary) mesh in velocity space[7] – in other words we assume that every element is part of a set of elements *having an identical extent in physical space* that, as a set, span velocity space. This simplifies the procedure for finding the element containing the post-collision velocities in equation (5.13) as well as for picking the pre-collision velocities (because $\mathbf{x}$ is the same for the two pre- and two post-collision velocities).

We also note that equation (5.10) holds when the integral is taken over an arbitrary region in *physical* space (though the integrals must still extend over all of velocity space). In particular, this means that we can independently evaluate (5.13) for each set of elements sharing a common extent in physical space; this greatly simplifies the implementation of a parallel code[8].

Implementing equation (5.13) is straightforward, provided an efficient method for generating the collision parameters $\{\mathbf{x}_i, \mathbf{c}_{1,i}, \mathbf{c}_{2,i}, \Theta_i\}$ exists. The method used in the present work is outlined in Appendix B.

[7] In the current code, a tensor product grid in all dimensions is used.

[8] The tradeoff in doing this is a smaller degree of variance reduction in cases where the magnitude of f^d varies significantly in physical space

5.2 Numerical results

We have validated this method using spatially homogeneous and one-dimensional model problems. Extension to higher-dimensional cases and other interaction models (e.g. variable hard-sphere gas [12]) is straightforward.

5.2.1 Spatially homogeneous case

The convergence properties of the RKDG method have been studied extensively[17]. For this reason, we will focus on the convergence results for the collision integral formulation of section 5.1.1 as this represents the primary difference between the present work and existing RKDG formulations.

We use the analytical solution [28] of the Boltzmann equation for spatially homogenous relaxation with Maxwellian molecules[9] as a testcase. Figure 5.1, shows the convergence for evaluating $\left[\frac{df}{dt}\right]_{\text{coll}}$ as a function of the mesh size for zeroth-order (p_0) and first-order (p_1) elements. The error reported in this case is the root-mean-square difference between the calculated and exact value of the collision integral at a lattice of points in velocity space (approximating an L_2 error norm). This error is normalized by the correct value of the collision integral at $\mathbf{c} = 0$.

As expected, with p_0 elements we observe a linear convergence rate, while the convergence rate is approximately quadratic with p_1 elements. This suggests that, when higher levels of accuracy are needed, it will be advantageous to use higher order elements, even when taking into account the fact that higher order elements have more degrees of freedom and a higher computational cost per element.

Although calculations using p_2 elements are certainly possible (as will be shown in the next section), we were unable to observe the expected asymptotic convergence rate for p_2 elements. This is primarily because the associated error levels are too low and decrease too quickly to accurately resolve with the Monte Carlo integration method used: for example, with only 11 elements in each dimension we obtain nor-

[9]Maxwellian molecules are defined such that $\sigma \propto 1/g$ (the constant of proportionality is immaterial for the present test case).

malized error levels on the order of 5×10^{-3}; the error levels in the asymptotic regime will be *much* lower. Our inability to observe the asymptotic convergence rate does not imply that there is no advantage in using p_2 or higher order elements; to the contrary, this low error level is precisely what one desires. However, the choice of the optimum polynomial order will also depend on statistical uncertainty considerations; this is discussed further in section 5.2.3. Additionally, note that the difficulty in resolving exceptionally low noise levels does not diminish the effectiveness of our Monte Carlo method for its intended purpose – obtaining low-noise solutions for low-signal flows. As will be shown later, our method is very effective at obtaining relative statistical uncertainties of significantly less than 1% for arbitrarily small deviations from equilibrium.

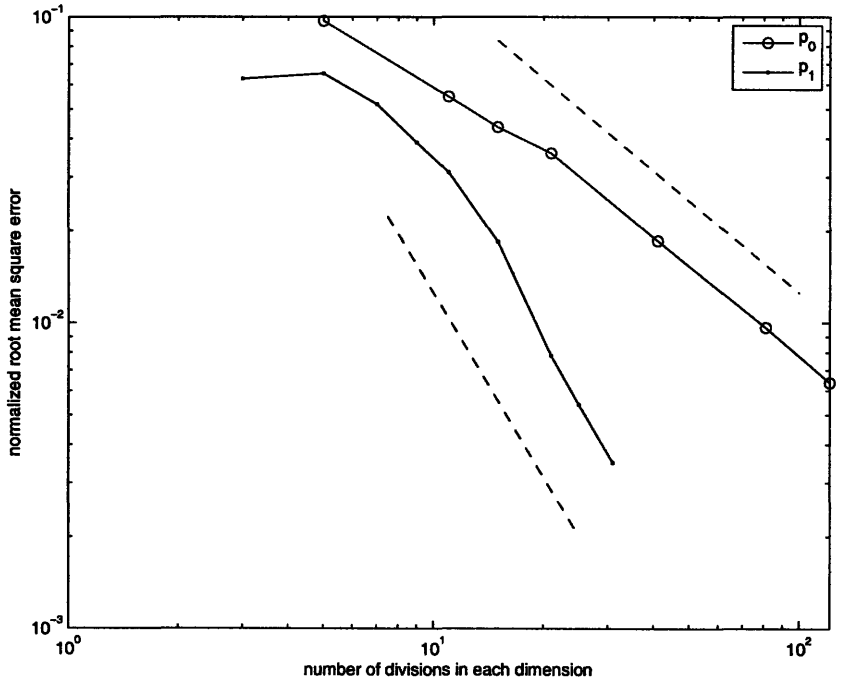

Figure 5.1: Error levels for collision integral as a function of number of elements used in each dimension. The dashed lines illustrate linear and quadratic convergence rates

5.2.2 Flow in a channel

In this section we present comparisons of our results with DSMC solutions of a transient Couette flow of a hard sphere gas, in which at $t = 0$, the system walls (at $x = -\ell/2$ and $x = \ell/2$) are impulsively accelerated to a velocity of ± 0.1. (This velocity was chosen to make comparison with DSMC possible; our method can solve this problem at arbitrary speeds for the same computational cost.) This flow represents a stringent test of the method, because it involves a discontinuity in the distribution function propagating into the computational domain.

All calculations presented in this section were performed using *nonuniform* elements; refinement in velocity space near $\mathbf{c} = 0$ was used in all cases. For $\mathrm{Kn} = 0.1$, refinement in physical space near the walls was also applied. Additionally, due to the propagating discontinuity in the $x-$direction, a finer discretization is used in the c_x direction than for the other velocity components. These mesh refinements significantly improved the quality of the results, although no attempt was made to optimize the mesh used.

Figures 5.2 through 5.7 show the velocity and shear stress profiles at a variety of Knudsen numbers, using p_1 elements. The discretization uses 20 elements in each dimension, except for c_x which uses 40. The results are compared against DSMC, and show an excellent level of agreement[10]. Figures 5.8 and 5.9 show velocity and shear stress profiles for the $\mathrm{Kn} = 1$ case using p_2 elements. This discretization uses 10 elements in each dimension, (20 for c_x), and overall uses 1/16 as many elements to obtain a similar degree of accuracy as in the p_1 case[11].

Figure 5.10 shows the steady-state results for pressure driven flow (using the linearized approach of [15]) for a Knudsen number of $\mathrm{Kn} = 2/\sqrt{\pi}$ compared to the previously published results of [31]. A total of 24^4 nonuniform elements were used.

[10]To ensure accurate results for comparison, the DSMC runs used 400 cells in physical space and a timestep such that a particle with a (dimensionless) normal velocity of 1 would take 10 timesteps to cross a cell. While we used a highly refined DSMC calculation to ensure that our results are correct, we base our performance characterization on a coarser discretization for DSMC.

[11]Note that a tensor product p_2 element has 3^4 shape functions compared to 2^4 for a p_1 element; this means that in this example the p_2 case has about 1/3 the number of degrees of freedom as the p_1 case.

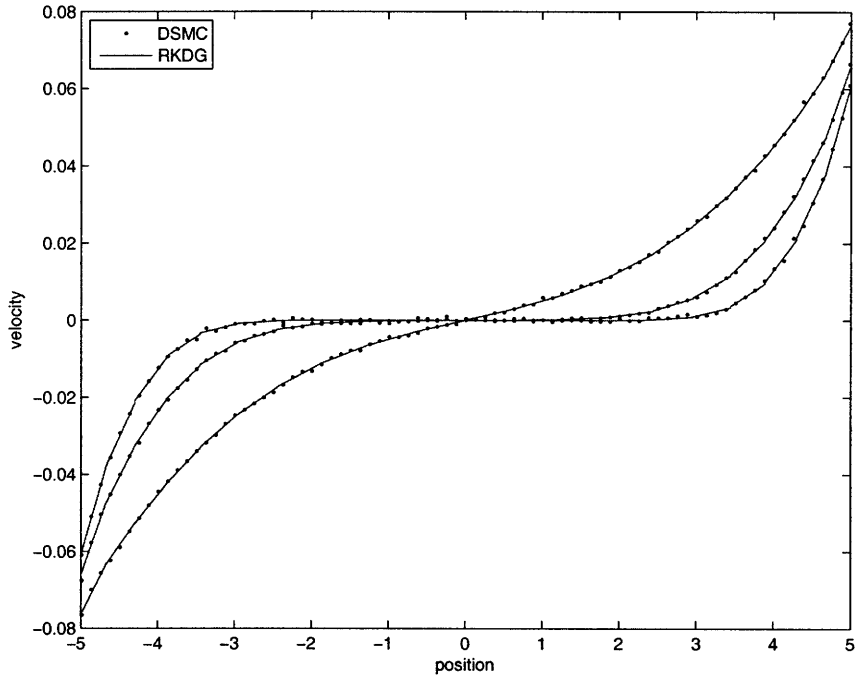

Figure 5.2: Velocity profiles for Kn $= 10^{-1}$ using 40×20^3 p_1 elements

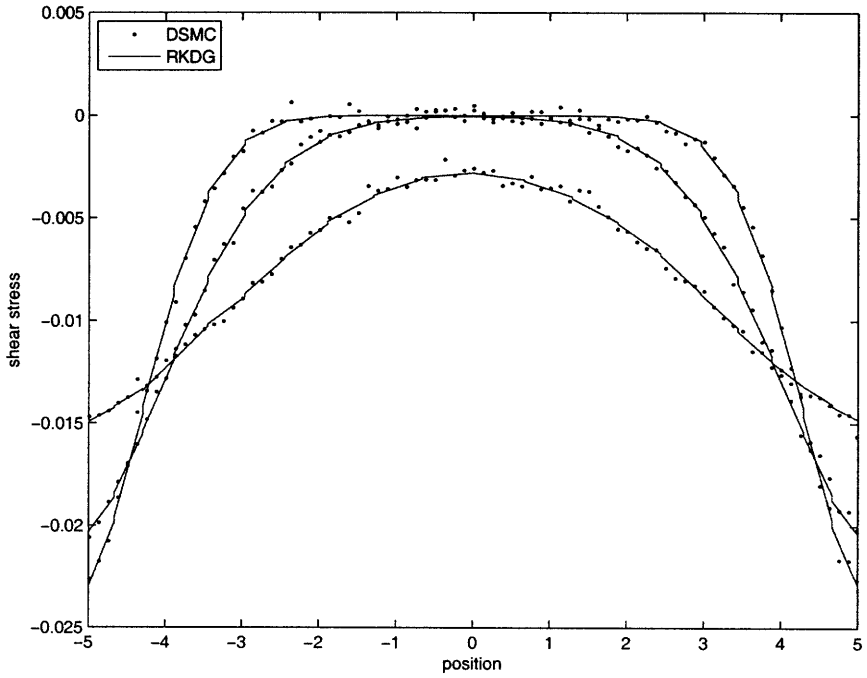

Figure 5.3: Shear stress profiles for Kn $= 10^{-1}$ using 40×20^3 p_1 elements

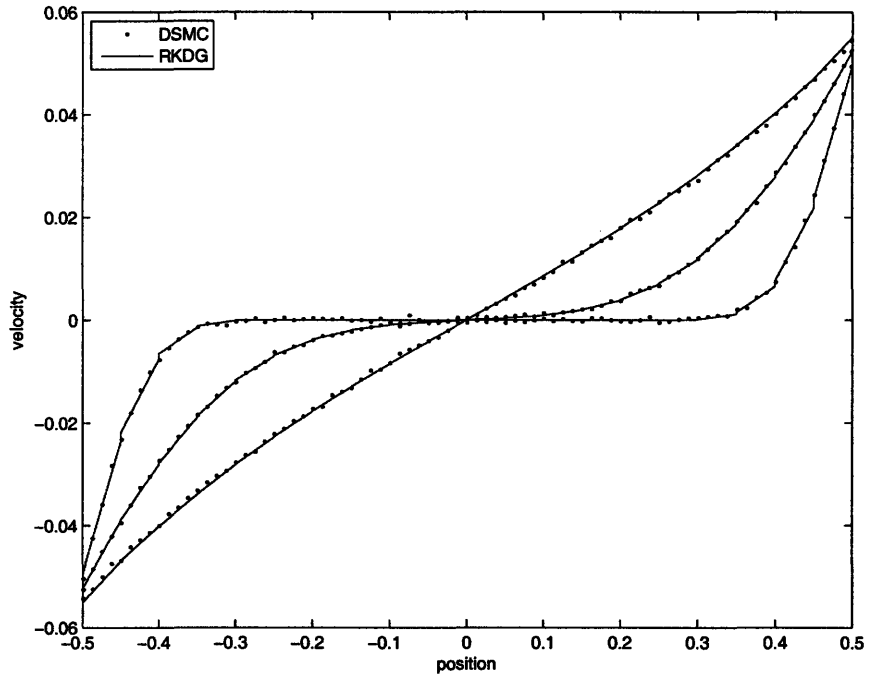

Figure 5.4: Velocity profiles for Kn = 1 using 40×20^3 p_1 elements

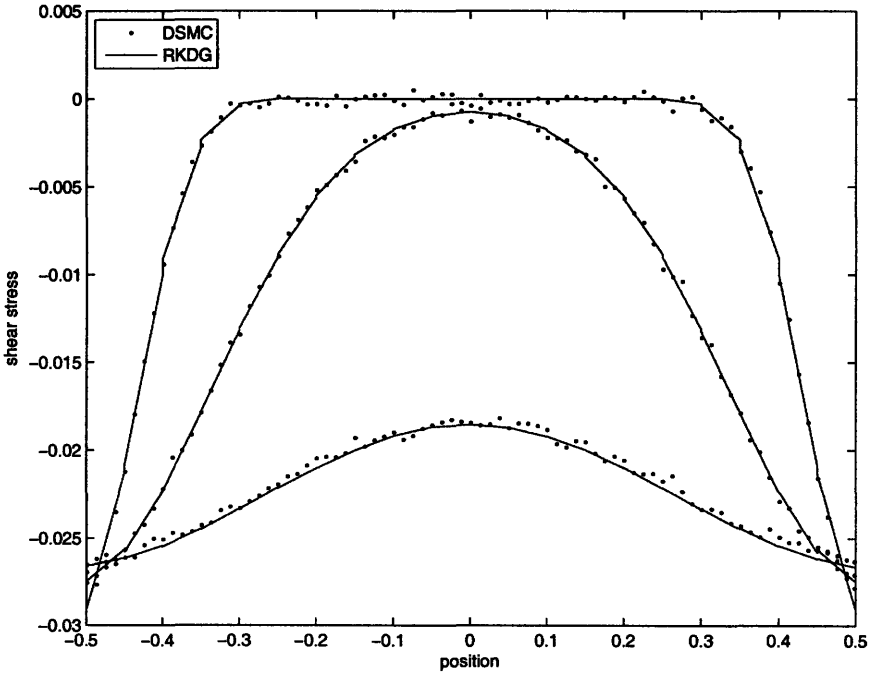

Figure 5.5: Shear stress profiles for Kn = 1 using 40×20^3 p_1 elements

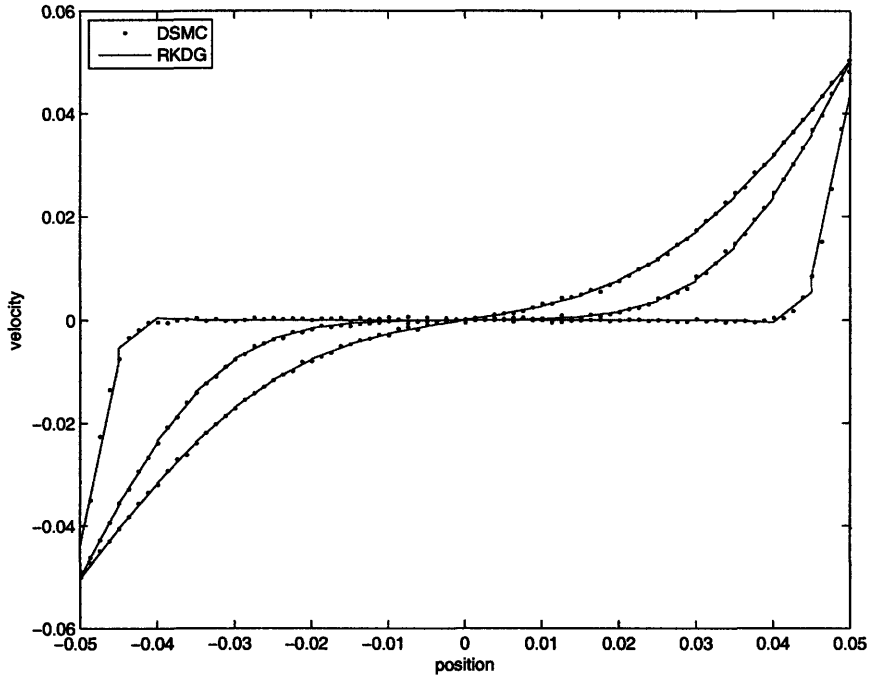

Figure 5.6: Velocity profiles for Kn = 10 using 40×20^3 p_1 elements

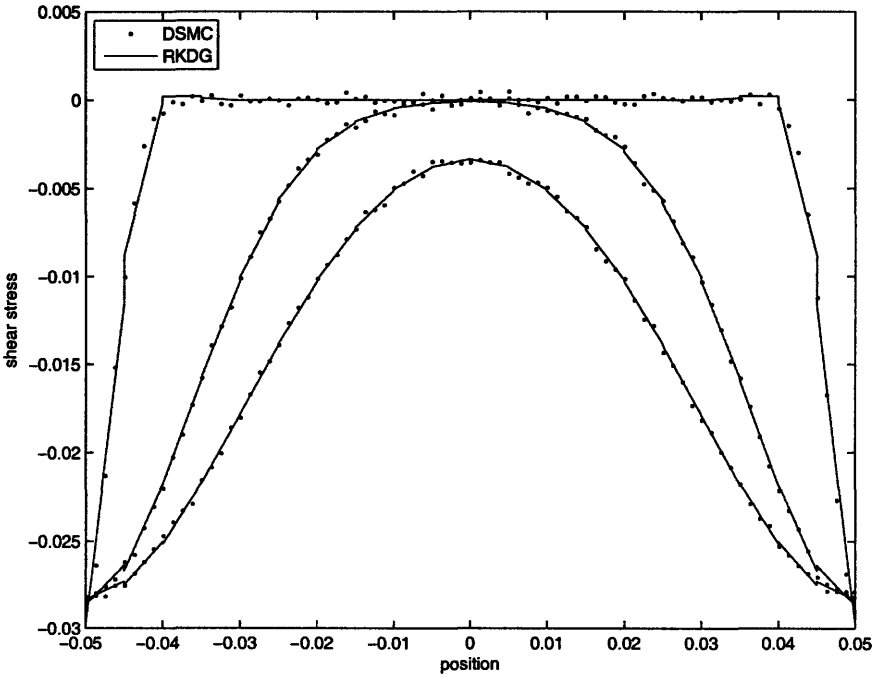

Figure 5.7: Shear stress profiles for Kn = 10 using 40×20^3 p_1 elements.

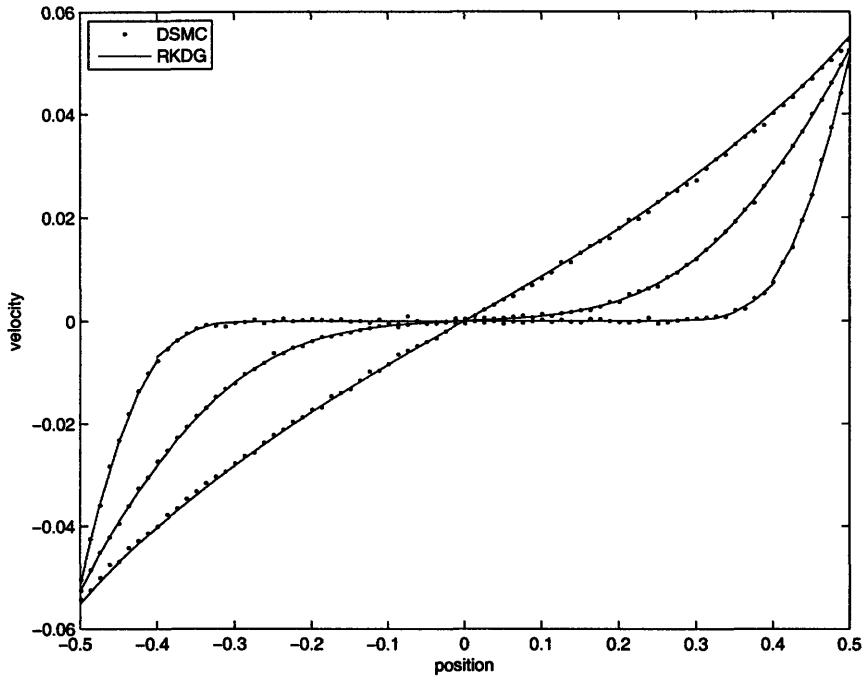

Figure 5.8: Velocity profiles for Kn = 1 using 20×10^3 p_2 elements

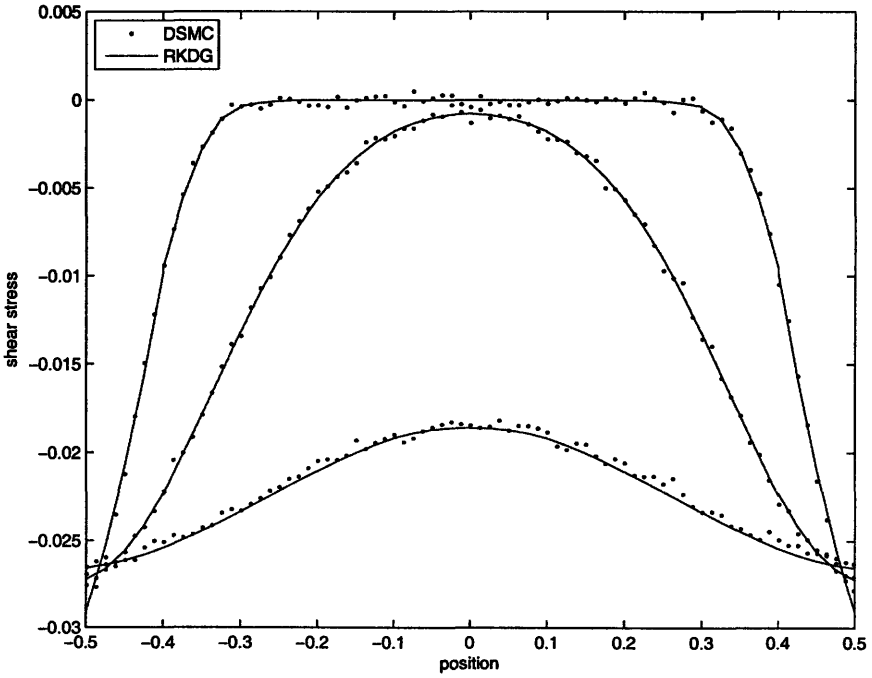

Figure 5.9: Shear stress profiles for Kn = 1 using 20×10^3 p_2 elements

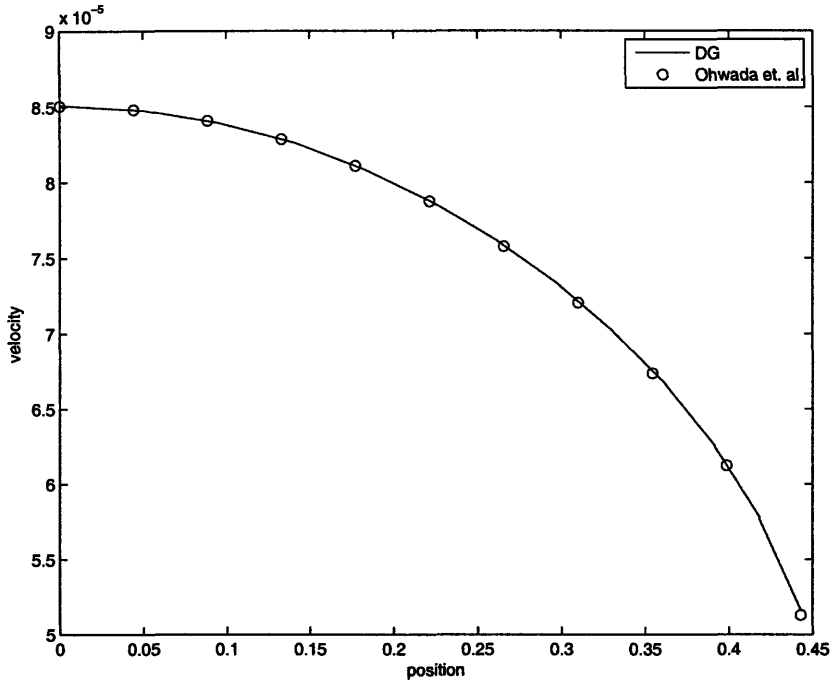

Figure 5.10: DG results for pressure driven flow (Kn $= 2/\sqrt{\pi} \approx 1.1$) using 24^4 p_1 elements

5.2.3 Effect of number of Monte Carlo samples

The use of a Monte Carlo integration method means that there will be some level of statistical uncertainty inherent in our evaluation of the collision integral. To find the effect of this uncertainty on the solution we have measured the effect of the number of Monte Carlo samples[12] used on the flow velocity and the shear stress. Specifically, we perform a steady-state shear flow calculation (using a Chapman-Enskog distribution for boundary conditions to minimize edge effects) and measure the mean standard deviation of the flow velocity and shear stress over a set of nodes in physical space[13]. These values are normalized by the boundary velocity and the correct value for the shear stress, respectively, and will be referred to as the relative

[12]Here, the number of samples refers to the total number of updates done in equation (5.13). When using order p elements, this will be $4\left(N_{MB}+N_d\right)(p+1)^4$; there are 4 pre- or post-collision velocities for each of the $(N_{MB}+N_d)$ collision events, and the sum in (5.13) is updated for each of the $(p+1)^4$ shape functions in the corresponding elements. In the current implementation, N_{MB} and N_d are allocated at runtime, so the number of samples reported is an approximation.

[13]Note that the nodes in the middle of the domain will typically have a smaller variance in these quantities; this is because f^d is smaller there as the mean flow velocity is closer to zero, so our choice for f^{MB} is a better approximation to the distribution function. This could be alleviated, though at the cost of a slightly more involved implementation than that discussed in section 5.1.3.

statistical uncertainties. For simplicity, in this section, all tests use a uniform mesh.

Figure 5.11 shows the dependence of the velocity and shear stress relative statistical uncertainties on the number of Monte Carlo samples per collision time[14] per degree of freedom for p_0 and p_1 elements using the *same* [15] discretization. The asymptotic convergence rate appears to be approaching $N^{-1/2}$, which is expected of Monte Carlo integration. In other words, the statistical uncertainty in the collision integral evaluation seems to have an (asymptotically) proportional effect on the statistical uncertainty of the hydrodynamic quantities of interest. When the number of samples is small, this trend no longer holds; in fact the method becomes unstable if too few samples are used.

Also, note that in both cases, higher order methods require fewer samples per degree of freedom to obtain a given uncertainty level (though higher order methods require more total samples for a given degree of uncertainty). In cases where extremely low levels of statistical uncertainty are required, this consideration might affect the choice of polynomial order and discretization used.

We can also see that the relative statistical uncertainty in the shear stress is typically an order of magnitude larger than the uncertainty in the velocity. Both, however, are very small; typical values are of $O(10^{-3})$ and $O(10^{-2})$ respectively which, as shown in the previous comparisons of figures (5.2)-(5.10), are essentially imperceptible.

Figure 5.12 shows how the number of elements used affects that level of relative statistical uncertainty for p_1 elements. We see that, when using more elements, fewer samples are needed per degree of freedom to obtain a fixed degree of relative statistical uncertainty (though the total number of samples needed is larger when more elements are used).

We have not yet touched on perhaps the most important aspect of our method for evaluating the collision integral: its performance for low-speed flows. Figure 5.13 illustrates how the degree of relative statistical uncertainty is affected by the

[14]The statistical uncertainty is not primarily affected by the number of samples per timestep, but rather the number of samples per collision time.

[15]As opposed to, for example, discretizations that would lead to the same degree of discretization error.

characteristic flow velocity (i.e. deviation from equilibrium). Two wall velocities (± 0.1 and ± 0.01) are shown; the figure shows that the relative statistical uncertainty does not change significantly between these two cases. This is in sharp contrast to DSMC, for which lower flow velocities are associated with much higher relative statistical uncertainties [22] – for the same number of samples the level of relative statistical uncertainty in a DSMC calculation would have increased by a factor of 10 while the number of samples required to bring the relative statistical uncertainty to the same level would have increased by a factor of 100.

Our preliminary results show that the effect of the number of Monte Carlo samples on the *average* value of the local flow velocity and shear stress is small – on the order 1% for the shear stress over the range shown in figure 5.11 and essentially negligible for more than 50 samples per collision time per degree of freedom. The effect on the flow velocity is even smaller.

Finally, we note that in the present method, only a relatively small number of samples are necessary per collision time per degree of freedom to obtain small relative statistical uncertainties. As the timestep is typically significantly less than a collision time, even fewer samples are needed per timestep; this makes the present method for evaluating the collision integral extremely computationally efficient.

5.2.4 Limiting

In general, RKDG methods require the use of a numerical limiter to ensure stability[17], although we have not found this necessary for the (linear) flux function of the Boltzmann equation. However, the lack of a flux limiter leads to a non-physical overshoot in the flow velocity and other hydrodynamic quantities at very short timescales due to the propagating discontinuity created by the impulsive wall acceleration in our Couette flow problem. An example is shown in figure 5.14; in this case collisionless Couette flow was simulated both without a flux limiter and with the flux limiter described in [17] (the flux limiter was only used for the initial 10 timesteps, during which time the propagating discontinuity has the sharpest effect on the velocity profile).

Without the flux limiter, there is a non-physical overshoot apparent at short

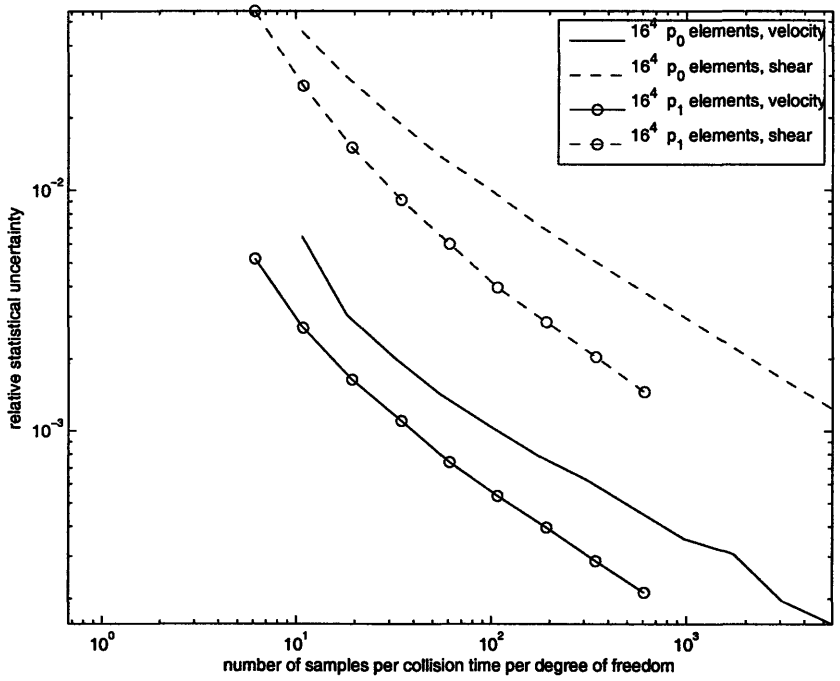

Figure 5.11: Scaling of relative statistical uncertainty in the flow velocity and shear stress with number of Monte Carlo samples per collision time per degree of freedom. Results obtained using 16^4 p_0 and p_1 elements

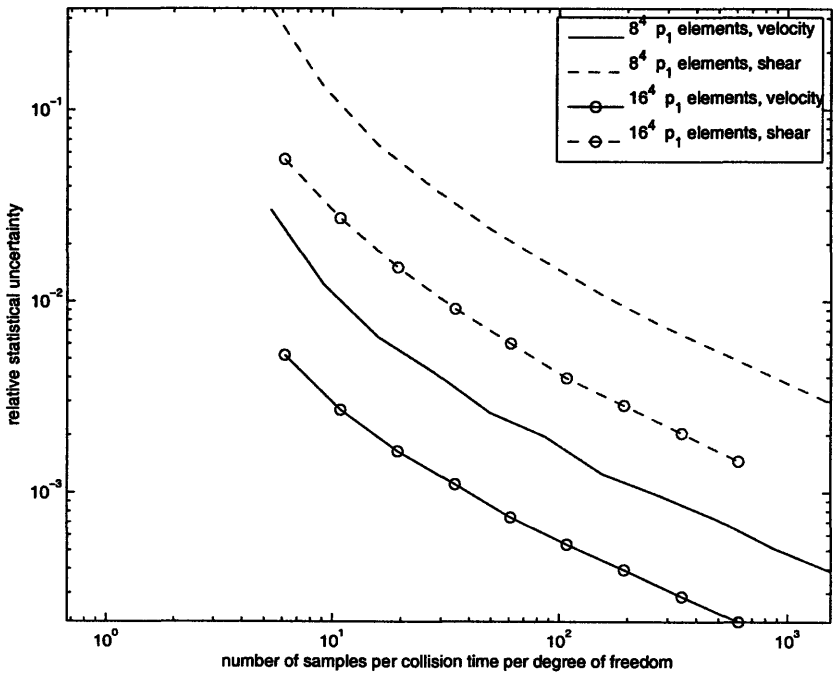

Figure 5.12: Scaling of relative statistical uncertainty in flow velocity and shear stress fluctuations with number of Monte Carlo samples per collision time per degree of freedom. Results use 8^4 and 16^4 p_1 elements

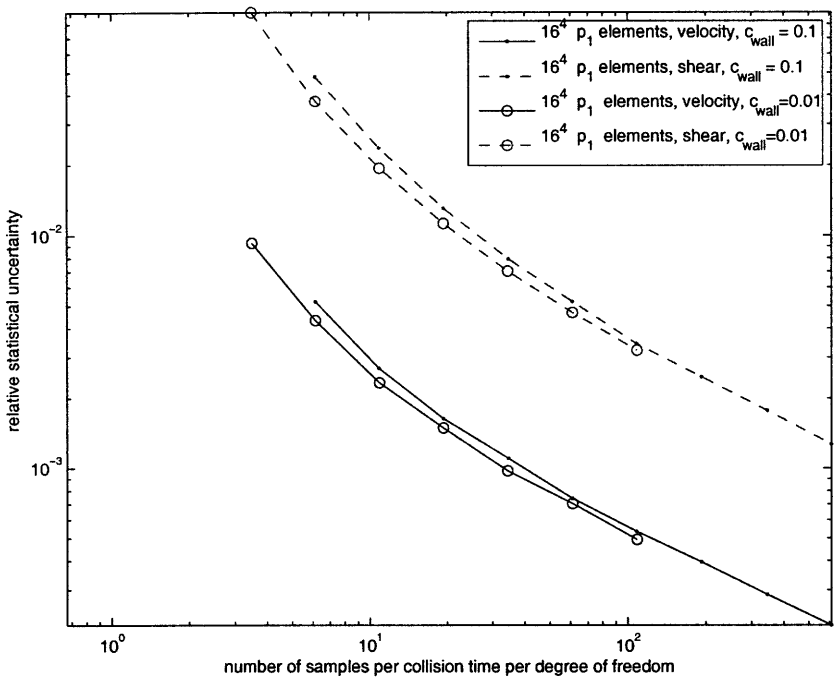

Figure 5.13: Effect of varying wall velocity on relative statistical uncertainty for present method

times that disappears at longer timescales. Introducing a flux limiter removes this overshoot. However, we observe that, except at very short timescales, we closely match the analytic results either with or without the flux limiter (to the point where the three different solutions plotted are essentially indistinguishable). Discerning the full effect of slope limiting on the present method of solving the Boltzmann equation will require further research, though our initial exploration suggests that, while the use of a slope limiter is not necessary, in certain cases it can give qualitatively better results.

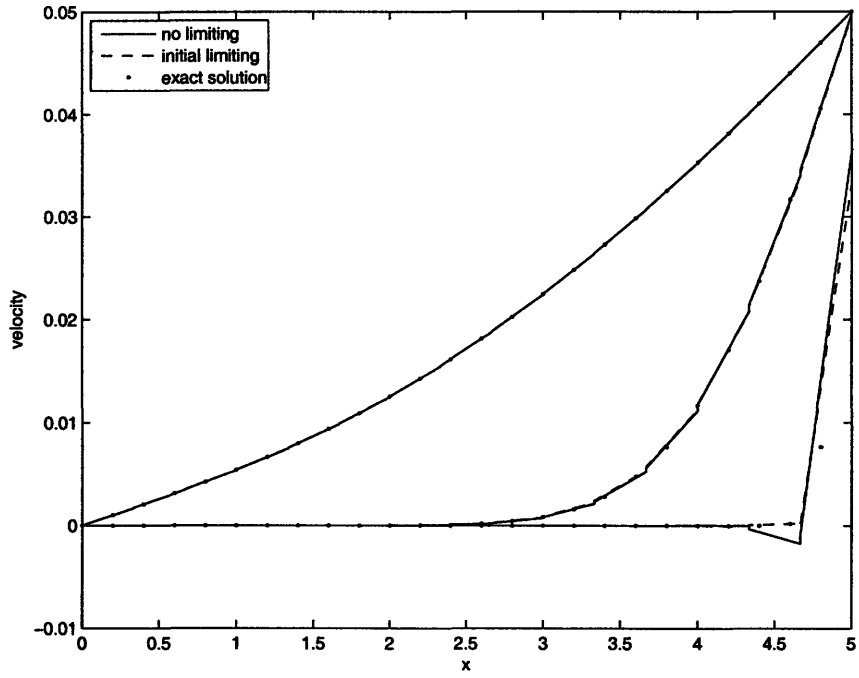

Figure 5.14: Effect of limiting on velocity profile for collisionless shear flow

Chapter 6

An iterative DG method

6.1 Time integration to steady-state

In practice, one is often interested in steady-state solutions for the Boltzmann equation. It is certainly possible to obtain these by integrating in time until steady-state is reached, however this is typically inefficient. With the RKDG method, the timestep is limited by a CFL condition, which can be approximated[1] for our one dimensional case by [17]

$$\Delta t \lesssim \frac{1}{2p+1} \frac{\Delta x_{\min}}{c_{x,\max}} \tag{6.1}$$

where p is the polynomial order, $c_{x,\max}$ is the maximum velocity in the x direction considered and $\Delta x_{\min}$ is the smallest element length in the x direction. We note that this is only an approximation for the true stability limit; however it will serve for our discussion purposes.

For flows in the diffusive regime (small Knudsen numbers), the (dimensionless) time to reach steady-state scales as

$$\tau_{ss} \sim \ell^2 \tag{6.2}$$

[1]This approximation assumes a $p+1$ stage RK time integration, and is typically within 5% of the numerically estimated threshold values when using a single velocity[17]. We have also neglected the factor of $\frac{\sqrt{\pi}}{2}$ appearing in the Boltzmann equation for simplicity.

where ℓ is the system length (normalized by λ^*); thus the minimum number of timesteps required to reach steady-state scales as

$$\mathbb{N} \sim \frac{(2p+1)\ell^2 c_{x,\max}}{\Delta x_{\min}} \tag{6.3}$$

In the high Knudsen number limit, the time to reach steady state scales as

$$\tau_{ss} \sim \frac{\ell}{c_{x,\min}} \tag{6.4}$$

where $c_{x,\min}$ is the smallest velocity for which steady-state will be reached.

Regardless of the regime one is working in, the relative effect of refining the mesh in the x dimension (and leaving the other parameters unchanged) is the same; the maximum timestep will decrease, while the time to reach steady state remains fixed. Thus, (assuming the computational cost per element per timestep is constant) the overall computational cost will scale as

$$\left(\frac{1}{\Delta x_{\min}}\right)^2 \tag{6.5}$$

The iterative method developed in this chapter will, for a fixed ℓ, require a number of iterations (roughly) independent of $\Delta x_{\min}$; in other words, in the above case of refinement in the x dimension and constant cost per element, the computational cost will scale as

$$\frac{1}{\Delta x_{\min}} \tag{6.6}$$

which is clearly a significant improvement.

We also note that the iterative method has other advantages besides a better scaling as the discretization is refined. Typically, the iterative method will require far fewer iterations (compared to the required number of timesteps when time integrating) to reach steady state. While the precise comparisons are difficult to make, a representative number of iterations required to reach steady state can be found in section 6.7.

6.2 Previous work

Our work is based on the iterative approach described in [39] in which the collision integral is split into gain and loss terms

$$\left[\frac{df}{dt}\right]_{\text{coll}} = \frac{\sqrt{\pi}}{2}\left[\iint f' f_1' \mathrm{g}\sigma d^2\Theta d^3\mathbf{c}_1 - f\iint f_1 \mathrm{g}\sigma d^2\Theta d^3\mathbf{c}_1\right] \tag{6.7}$$

$$\equiv G - fL \tag{6.8}$$

At each iteration I, the distribution function is chosen such that the (discretized) Boltzmann equation would be satisfied if G and L remained unchanged from the previous iteration; that is (ignoring discretization) f^I is chosen such that

$$\frac{\sqrt{\pi}}{2}\mathbf{c}\cdot\frac{\partial f^I}{\partial \mathbf{r}} = G^{I-1} - f^I L^{I-1} \tag{6.9}$$

at each point in velocity and physical space. Here,

$$G^{I-1} = \frac{\sqrt{\pi}}{2}\iint f'^{,I-1} f_1'^{,I-1} \mathrm{g}\sigma d^2\Theta d^3\mathbf{c}_1 \tag{6.10}$$

$$L^{I-1} = \frac{\sqrt{\pi}}{2}\iint f_1^{I-1} \mathrm{g}\sigma d^2\Theta d^3\mathbf{c}_1 \tag{6.11}$$

In other words, at each iteration, we select shape coefficient coefficients such that the Boltzmann equation is satisfied using an *approximation* for the collision integral based on G and L from the previous iteration. It is clear that, if convergence is reached, that is if $f^I = f^{I-1}$, this method will yield a solution to the discretized Boltzmann equation. We note that this approximation is implicit, that is, it involves information at iteration I. Although, to our knowledge, rigorous numerical analysis of this formulation has not been performed, an implicit formulation seems to be required for stability [39].

The combination of this iterative method with the variance reduction techniques described in this work is discussed in [7]. The work in [7] used a finite volume discretization for the distribution function. In the remainder of this chapter, we will

show how this method can be extended to the DG formulation. We will also demonstrate the relationship between this iterative method and a more general Newton's method.

6.3 Discontinuous Galerkin formulation

The focus of this chapter is on the treatment of the collision integral. The natural extension of the above iterative approach to the DG method discussed in Chapter 5 is to use

$$\mathbb{I}^I_{\text{iterative}}(v) \equiv \iint v(G^{I-1} - f^I L^{I-1})\, d^3\mathbf{c}\, d^3\mathbf{x} \tag{6.12}$$

as our approximation for

$$\mathbb{I}^I(v) \equiv \iint v \left[\frac{df}{dt}\right]^I_{\text{coll}} d^3\mathbf{c} d^3\mathbf{x} \tag{6.13}$$

We note that using the weak form of the collision integral (1.19), we can write

$$\int v G^{I-1}\, d^3\mathbf{c} = \frac{\sqrt{\pi}}{4} \iiint (v_1' + v_2')\, f_1^{I-1} f_2^{I-1} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2 \tag{6.14}$$

while from above we have

$$L^{I-1} = \frac{\sqrt{\pi}}{2} \iint f_1^{I-1} \mathrm{g}\sigma d^2\Theta d^3\mathbf{c}_1 \tag{6.15}$$

Substituting definition (2.13), taking f^{MB} to be independent of I and taking advantage of the fact that the weak form of the collision integral is identically zero

for f^{MB} yields

$$\begin{aligned}
\mathbb{I}^I_{\text{iterative}}(v) &\approx \iint \left(vG^{I-1} - vf^I L^{I-1}\right) d^3\mathbf{c}\, d^3\mathbf{x} \\
&= \frac{\sqrt{\pi}}{4} \iiiint (v_1' + v_2') \left(2f_1^{d,I-1} f_2^{MB} + f_1^{d,I-1} f_2^{d,I-1}\right) \\
&\quad \times g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \\
&\quad - \frac{\sqrt{\pi}}{2} \iiiint v_2 \left(f_1^{d,I-1} f_2^{MB} + f_1^{MB} f_2^{d,I} + f_1^{d,I-1} f_2^{d,I}\right) \\
&\quad \times g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \qquad (6.16) \\
&\equiv \mathcal{G}(v) - \mathcal{L}(v) \qquad (6.17)
\end{aligned}$$

We will consider the two terms in the above expression separately. The first term can be rewritten as

$$\begin{aligned}
\mathcal{G}(v) \equiv \frac{\sqrt{\pi}}{2} &\iiiint (v_1' + v_2')\, f_1^{d,I-1} f_2^{MB} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \\
+ \frac{\sqrt{\pi}}{4} &\iiiint (v_1' + v_2')\, f_1^{d,I-1} f_2^{d,I-1} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \qquad (6.18)
\end{aligned}$$

while

$$\begin{aligned}
\mathcal{L}(v) = \frac{\sqrt{\pi}}{2} &\iiiint v_2 f_1^{d,I-1} f_2^{MB} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \\
+ \frac{\sqrt{\pi}}{2} &\iiiint v_2 f_1^{MB} f_2^{d,I} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \\
+ \frac{\sqrt{\pi}}{2} &\iiiint v_2 f_1^{d,I-1} f_2^{d,I} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \qquad (6.19)
\end{aligned}$$

We now expand the term $f_2^{d,I}$ in terms of the shape function coefficients using

equation (5.5) to obtain

$$\begin{aligned}\mathcal{L}(v) = &\frac{\sqrt{\pi}}{2}\iiiint v_2 f_1^{d,I-1} f_2^{MB} g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \\ &+ \sum_i \hat{f}_i^{d,I} \frac{\sqrt{\pi}}{2} \iiiint v_2 \phi_{i,2} f_1^{MB} g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \\ &+ \sum_i \hat{f}_i^{d,I} \frac{\sqrt{\pi}}{2} \iiiint v_2 \phi_{i,2} f_1^{d,I-1} g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \end{aligned} \tag{6.20}$$

Recall that v and ϕ are defined to be zero outside of the element in question.

Now, perform a standard manipulation by noting that formally interchanging $\mathbf{c}_1$ and $\mathbf{c}_2$ makes no difference in the value of integral, and of course "two quantities that are equal to each other are each equal to one-half their sum" [37]

$$\begin{aligned}\mathcal{L}(v) = &\frac{\sqrt{\pi}}{2}\iiiint v_2 f_1^{d,I-1} f_2^{MB} g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \\ &+ \sum_i \hat{f^{d,I}}_i \frac{\sqrt{\pi}}{2} \iiiint v_2 \phi_{i,2} f_1^{MB} g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \\ &+ \sum_i \hat{f^{d,I}}_i \frac{\sqrt{\pi}}{4} \iiiint \left(\frac{v_1 \phi_{i,1}}{f_1^{d,I-1}} + \frac{v_2 \phi_{i,2}}{f_2^{d,I-1}} \right) f_1^{d,I-1} f_2^{d,I-1} g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \end{aligned} \tag{6.21}$$

This, along with (6.18) are the final expressions we will be using.

6.4 Evaluation of terms

In the present work, equations (6.18) and (6.21) are evaluated using the same importance sampling technique used in Chapter 5. This approach allows the calculation of the correct collision integral at step $I-1$, while evaluating the terms needed to evaluate the *estimate* for the collision integral at step I; the evaluation of the correct collision integral allows one to check the convergence of the method, and is beneficial if one decided to implement a line search or similar algorithm[32].

This is certainly not the only possible implementation; in fact, as we will see shortly, the present implementation requires the division by f^d (with a probability

proportional to $|f^d|$), in other words there is a small probability of dividing by a number close to zero. A better importance sampling strategy is almost certainly possible; but this has not yet been explored.

6.4.1 Terms in $\mathcal{G}$

We follow the standard procedure explained in Chapter 5 to write

$$\frac{\sqrt{\pi}}{2}\iiiint (v_1' + v_2')\left(f_1^{d,I-1} f_2^{MB}\right) g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \approx \frac{\sqrt{\pi}}{2}\frac{\chi_{MB,d}}{N}\sum_{j=1}^{N}\left(v_{1,j}' + v_{2,j}'\right) g_j\sigma_j \mathrm{sgn}\left(f_{1,j}^{d,I-1}\right) \quad (6.22)$$

where the collision parameters $\{\mathbf{x}_j, \mathbf{c}_{1,j}, \mathbf{c}_{2,j}, \Theta_j\}$ are chosen from the probability distribution function $f_2^{MB}|f_1^{d,I-1}|/\chi_{MB,d}$.

We also write

$$\frac{\sqrt{\pi}}{4}\iiiint (v_1' + v_2')\left(f_1^{d,I-1} f_2^{d,I-1}\right) g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \approx \frac{\sqrt{\pi}}{4}\frac{\chi_{d,d}}{N}\sum_{j=1}^{N}\left(v_{1,j}' + v_{2,j}'\right) g_j\sigma_j \mathrm{sgn}\left(f_{1,j}^{d,I-1}\right)\mathrm{sgn}\left(f_{2,j}^{d,I-1}\right) \quad (6.23)$$

when the collision parameters are chosen with a probability $|f_1^{d,I-1}||f_2^{d,I-1}|/\chi_{d,d}$.

6.4.2 Terms in $\mathcal{L}$

The only term in $\mathcal{L}$ that can be evaluated explicitly from information known at step $I-1$ is

$$\frac{\sqrt{\pi}}{2}\iiiint v_2 f_1^{d,I-1} f_2^{MB} g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \approx \frac{\sqrt{\pi}}{2}\frac{\chi_{MB,d}}{N}\sum_{j=1}^{N} v_{2,j} g_j\sigma_j \mathrm{sgn}\left(f_{1,j}^{d,I-1}\right) \quad (6.24)$$

where the collision parameters $\{\mathbf{x}_j, \mathbf{c}_{1,j}, \mathbf{c}_{2,j}, \Theta_j\}$ are chosen with a probability $f_2^{MB}|f_1^{d,I-1}|/\chi_{MB,d}$.

The other two terms in $\mathcal{L}$ involve f^d at step I; these must be evaluated by summing

over shape functions. The terms can be calcuated by

$$\frac{\sqrt{\pi}}{2}\iiiint v_1\phi_{i,1}f_2^{MB}g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \approx \frac{\sqrt{\pi}}{2}\frac{\chi_{MB,d}}{N}\sum_{j=1}^{N} v_{1,j}g_j\sigma_j\phi_{i,1,j}\frac{1}{|f_{1,j}^{d,I-1}|} \tag{6.25}$$

where the collision parameters are chosen with a probability $|f_1^{d,I-1}|f_2^{MB}/\chi_{MB,d}$, and

$$\frac{\sqrt{\pi}}{4}\iiiint \left(\frac{v_2\phi_{i,2}}{f_2^{d,I-1}}+\frac{v_1\phi_{i,1}}{f_1^{d,I-1}}\right) f_1^{d,I-1}f_2^{d,I-1}g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \approx$$
$$\frac{\sqrt{\pi}}{4}\frac{\chi_{d,d}}{N}\sum_{j=1}^{N}\left(\frac{v_{2,j}\phi_{i,2,j}}{f_{2,j}^{d,I-1}}+\frac{v_{1,j}\phi_{i,1,j}}{f_{1,j}^{d,I-1}}\right)\mathrm{sgn}\left(f_{1,j}^{d,I-1}\right)\mathrm{sgn}\left(f_{2,j}^{d,I-1}\right) \tag{6.26}$$

when the collision parameters are chosen with a probability $|f_1^{d,I-1}||f_2^{d,I-1}|/\chi_{d,d}$.

6.5 Implementation

Our focus in this chapter has been on the development of a scheme to approximate a weak form of an implicit collision integral formulation. The approximation that we have developed involves the (unknown) shape function coefficients $\hat{f}_i^{d,I}$. The advection terms (the left side of the discretized Boltzmann equation) are easily expressed in terms of the shape function coefficients. This leads to a set of linear equations that can be solved for the shape function coefficients $\hat{f}_i^{d,I}$. Forming and solving the matrices involved is a standard procedure, so we will not discuss it at length here.

We do comment on the structure of the linear systems encountered. First, we note that (with our present tensor product mesh) there is no coupling between elements that have a different position in velocity space[2]; in other words we can solve for the shape function coefficients $\hat{f}_i^d$ for each set of elements in physical space with a particular range in velocity space independently of elements with a different range in velocity space.

[2]These terms are coupled via the collision integral but there is no coupling when solving the set of linear equations.

Assume that we have N_c elements in each of 3 dimensions of velocity space and N_x elements in each of D dimensions of physical space and we are using p^{th} order tensor product shape functions. In three physical dimensions, we will have to solve N_c^3 sets of linear equations. Each of these sets of linear equations will have $N_x^D (p+1)^{D+3}$ unknowns, and each corresponding matrix will have (approximately) $2DN_x^D (p+1)^{2(D+3)}$ nonzero terms. This number of nonzero terms results from the fact that, for each element, all of the shape function coefficients in the the upwind adjoining elements will contribute to the flux, and only the shape function coefficients in that particular element will contribute to the collision integral terms.

For the cases in one physical dimension presented here, the cost in evaluating the collision integral terms has been much larger than that required to solve the resulting systems of linear equations.

This iterative method as presented here has a significant storage cost – one would typically store the terms (6.22), (6.23), (6.24), (6.25) and (6.26) for all elements when evaluating the collision integral. However, we note that (6.25) and (6.26) have to be evaluated for all *pairs* of shape functions within a particular element. Thus the overall storage cost will scale as $N_x^D N_c^3 (p+1)^{2(D+3)}$. For a fine discretization in a large number of dimensions, this might strain the memory (or mass storage) limits of the computer being used. There are several possibilities for alleviating this; for example one could trade off CPU time for storage by evaluating (6.25) and (6.26) for different elements at different times. For example, due to the upwind scheme used, the resulting linear equations can often be solved by back-substitution.

6.6 Newton's method

6.6.1 Some definitions

In this chapter, we derived expression (6.17) based on a previous "empirically" developed iterative method. In this section, we will show how the present method is related to Newton's method, a standard iterative method for solving nonlinear sys-

tems of equations [32].

We begin by defining

$$\mathcal{I}\left[f^d, f^d; f^{MB}\right](v) \equiv \frac{\sqrt{\pi}}{4}\iiiint \left(v_1' + v_2' - v_1 - v_2\right)\left(2f_1^d f_2^{MB} + f_1^d f_2^d\right) g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \quad (6.27)$$

$\mathcal{I}$ is the moment with v of the collision integral for the deviational distribution function f^d.

We then define the Jacobian term by considering the change in $\mathcal{I}$ when f^d is changed. In particular, we will consider the modification of the coefficient for the shape function ψ by an amount ϵ.

$$\begin{aligned} J_{\psi,v} &\equiv \lim_{\epsilon\to 0}\frac{\mathcal{I}[f^d + \epsilon\psi, f^d + \epsilon\psi; f^{MB}](v) \;- \mathcal{I}[f^d, f^d; f^{MB}](v)}{\epsilon} \\ &= \frac{\sqrt{\pi}}{4}\iiiint \left(v_1' + v_2' - v_1 - v_2\right)\left(2\psi_1 f_2^{MB} + \psi_1 f_2^d + f_1^d\psi_2\right) g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \end{aligned} \quad (6.28)$$

The term $J_{\psi,v}$ tells us the sensitivity of $\mathcal{I}(v)$ to a change in the coefficient of the shape function ψ. In general, J will be nonzero for any two shape functions that share a common region in physical space and zero otherwise.

With this Jacobian matrix, we can approximate the collision integral at iteration I using a linear expansion

$$\begin{aligned} \mathcal{I}_{\text{Newton}}\left[f^d, f^d; f^{MB}\right](v) &\equiv \iint v\left[\frac{df}{dt}\right]^{I}_{\text{coll, Newton}} d^3\mathbf{c}\, d^3\mathbf{x} \\ &= \iint v\left[\frac{df}{dt}\right]^{I-1}_{\text{coll}} d^3\mathbf{c}\, d^3\mathbf{x} + \sum_i J^{I-1}_{\phi_i,v}\left(\hat{f}_i^{d,I} - \hat{f}_i^{d,I-1}\right) \end{aligned} \quad (6.29)$$

Here, i indexes all shape functions in all elements (or equivalently, all shape functions in all elements sharing a common region in physical space with the element of interest). Substituting (6.28) for J, and using (5.10) for the collision integral term at $I-1$, and

using (5.5) to simplify yields

$$\mathcal{I}_{\text{Newton}}\left[f^{d,I}, f^{d,I}; f^{MB}\right](v) = \frac{\sqrt{\pi}}{4}\iiiint (v_1' + v_2' - v_1 - v_2) \times \left(2f_1^{d,I} f_2^{MB} + f_1^{d,I-1} f_2^{d,I-1} + \left[f_1^{d,I} - f_1^{d,I-1}\right] f_2^{d,I-1} + f_1^{d,I-1}\left[f_2^{d,I} - f_2^{d,I-1}\right]\right) \times g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \quad (6.30)$$

which can be rewritten as

$$\mathcal{I}_{\text{Newton}}\left[f^{d,I}, f^{d,I}; f^{MB}\right](v) = \mathcal{I}\left[f^{d,I}, f^{d,I}; f^{MB}\right](v) - \mathcal{I}\left[f^{d,I} - f^{d,I-1}, f^{d,I} - f^{d,I-1}; 0\right](v) \quad (6.31)$$

Let us again separate this into two terms, writing

$$\mathcal{I}_{\text{Newton}}\left[f^{d,I}, f^{d,I}; f^{MB}\right](v) = \mathcal{G}_{\text{Newton}}(v) - \mathcal{L}_{\text{Newton}}(v) \quad (6.32)$$

where we have defined

$$\mathcal{G}_{\text{Newton}}(v) = \frac{\sqrt{\pi}}{4}\iiiint (v_1' + v_2') \times \left(2f_1^{d,I} f_2^{MB} + f_1^{d,I-1} f_2^{d,I-1} + \left[f_1^{d,I} - f_1^{d,I-1}\right] f_2^{d,I-1} + f_1^{d,I-1}\left[f_2^{d,I} - f_2^{d,I-1}\right]\right) \times g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \quad (6.33)$$

and

$$\mathcal{L}_{\text{Newton}}(v) = \frac{\sqrt{\pi}}{4}\iiiint (v_1 + v_2) \times \left(2f_1^{d,I} f_2^{MB} + f_1^{d,I-1} f_2^{d,I-1} + \left[f_1^{d,I} - f_1^{d,I-1}\right] f_2^{d,I-1} + f_1^{d,I-1}\left[f_2^{d,I} - f_2^{d,I-1}\right]\right) \times g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \quad (6.34)$$

Using standard techniques, we can write $\mathcal{L}_{\text{Newton}}$ in an alternative form

$$\mathcal{L}_{\text{Newton}}(v) = \frac{\sqrt{\pi}}{2} \iiiint v_2 \left(f_1^{d,I} f_2^{MB} + f_1^{MB} f_2^{d,I} + f_1^{d,I-1} f_2^{d,I} + \left[f_1^{d,I} - f_1^{d,I-1} \right] f_2^{d,I-1} \right) \times g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \quad (6.35)$$

6.6.2 Approximation to Newton's method

Let us now consider a modification to Newton's method, in which we retain only the block diagonal[3] terms in the Jacobian matrix, that is we write

$$\mathcal{I}_{\text{modified}} \left[f^d, f^d; f^{MB} \right](v) = \iint v \left[\frac{df}{dt} \right]_{\text{coll}}^{I-1} d^3\mathbf{c} \, d^3\mathbf{x} + \sum_s J_{\phi_s,v}^{I-1} \left(\hat{f}_s^{d,I} - \hat{f}_s^{d,I-1} \right) \quad (6.36)$$

where, in contrast to (6.29), s indexes only those shape functions that are nonzero in the *same* element as v; we denote this element by Ω_v. We can also write

$$\mathcal{I}_{\text{modified}} \left[f^d, f^d; f^{MB} \right](v) = \frac{\sqrt{\pi}}{4} \iiiint (v_1' + v_2' - v_1 - v_2) \times \left(2 f_1^{d,I-1} f_2^{MB} + f_1^{d,I-1} f_2^{d,I-1} + 2 \left[\check{f}_1^{d,I} - f_1^{d,I-1} \right] f_2^{MB} + \left[\check{f}_1^{d,I} - f_1^{d,I-1} \right] f_2^{d,I-1} + f_1^{d,I-1} \left[\check{f}_2^{d,I} - f_2^{d,I-1} \right] \right) \times g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x}$$

where

$$\check{f}_1^{d,I} = \begin{cases} f_1^{d,I} & \text{if } \mathbf{c}_1 \in \Omega_v \\ f_1^{d,I-1} & \text{otherwise} \end{cases} \quad (6.37)$$

$$\check{f}_2^{d,I} = \begin{cases} f_2^{d,I} & \text{if } \mathbf{c}_2 \in \Omega_v \\ f_2^{d,I-1} & \text{otherwise} \end{cases} \quad (6.38)$$

[3]We assume that the terms are ordered with the element index most significant, and the index of the shape function least significant. In other words, we are only considering the terms in the Jacobian that correspond to shape functions in the same element.

The corresponding relations for $\mathcal{L}_{\text{modified}}(v)$ and $\mathcal{G}_{\text{modified}}(v)$ are

$$\mathcal{L}_{\text{modified}}(v) = \frac{\sqrt{\pi}}{2} \iiiint v_2 \left(\check{f}_1^{d,I} f_2^{MB} + f_1^{MB} \check{f}_2^{d,I} + f_1^{d,I-1} \check{f}_2^{d,I} + \left[\check{f}_1^{d,I} - f_1^{d,I-1} \right] f_2^{d,I-1} \right) \times g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \quad (6.39)$$

and

$$\mathcal{G}_{\text{modified}}(v) = \frac{\sqrt{\pi}}{4} \iiiint (v_1' + v_2') \times \left(2\check{f}_1^{d,I} f_2^{MB} + f_1^{d,I-1} f_2^{d,I-1} + \left[\check{f}_1^{d,I} - f_1^{d,I-1} \right] f_2^{d,I-1} + f_1^{d,I-1} \left[\check{f}_2^{d,I} - f_2^{d,I-1} \right] \right) \times g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \quad (6.40)$$

where we have made use of the relationship $f_1^{d,I-1} f_2^{MB} + \left[\check{f}_1^{d,I} - f_1^{d,I-1} \right] f_2^{MB} = \check{f}_1^{d,I} f_2^{MB}$.

6.6.3 Connection between present method and Newton's method

We will now show that the iterative method developed in this chapter can be related to Newton's method. Specifically, we show that this method corresponds to Newton's method in the case where only the block diagonal terms in the Jacobian are kept (after some approximation), i.e. the modified method of section 6.6.2.

To see this more clearly, let us begin by comparing

$$\mathcal{G}_{\text{modified}}(v) = \frac{\sqrt{\pi}}{4} \iiiint (v_1' + v_2') \times \left(2\check{f}_1^{d,I} f_2^{MB} + f_1^{d,I-1} f_2^{d,I-1} + \left[\check{f}_1^{d,I} - f_1^{d,I-1} \right] f_2^{d,I-1} + f_1^{d,I-1} \left[\check{f}_2^{d,I} - f_2^{d,I-1} \right] \right) \times g\sigma \, d^2\Theta \, d^3\mathbf{c}_1 \, d^3\mathbf{c}_2 \, d^3\mathbf{x} \quad [6.40]$$

and

$$\mathcal{G}(v) \equiv \frac{\sqrt{\pi}}{2} \iiiint (v_1' + v_2')\, f_1^{d,I-1} f_2^{MB} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x}$$
$$+ \frac{\sqrt{\pi}}{4} \iiiint (v_1' + v_2')\, f_1^{d,I-1} f_2^{d,I-1} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \qquad [6.18]$$

Let us examine the situation for a particular shape function v (and again denote the element in which v is nonzero by Ω_v). In a "typical" collision event for which $\mathbf{c}_1' \in \Omega_v$ or $\mathbf{c}_2' \in \Omega_v$ (in other words, a "typical" collision event for which the integrand of $\mathcal{G}(v)$ and $\mathcal{G}_{\text{modified}}(v)$ is not necessarily zero) neither $\mathbf{c}_1$ nor $\mathbf{c}_2$ will be in element Ω_v. We can see that, if we consider only these "typical" collision events, $\mathcal{G}(v)$ is clearly equal to $\mathcal{G}_{\text{modified}}(v)$ (because $\check{f}_1^{d,I} = f_1^{d,I-1}$ and $\check{f}_2^{d,I} = f_2^{d,I-1}$ in this case)

The approximation for $\mathcal{G}(v)$ (compared to $\mathcal{G}_{\text{modified}}(v)$) in our present iterative method, is in the collision events for which either $\mathbf{c}_1$ or $\mathbf{c}_2$ lies in the element Ω_v; however these collision events are expected to be rare. For example, if both $\mathbf{c}_1$ and $\mathbf{c}_2$ lie in Ω_v, g will be small. On the other hand, if g is large, there is only a small range of Θ for which $\mathbf{c}_1$ and either $\mathbf{c}_1'$ or $\mathbf{c}_2'$ can be in the element Ω_v. A similar argument holds for $\mathbf{c}_2$.

We now move on to compare

$$\mathcal{L}_{\text{modified}}(v) =$$
$$\frac{\sqrt{\pi}}{2} \iiiint v_2 \left(\check{f}_1^{d,I} f_2^{MB} + f_1^{MB} \check{f}_2^{d,I} + f_1^{d,I-1} \check{f}_2^{d,I} + \left[\check{f}_1^{d,I} - f_1^{d,I-1} \right] f_2^{d,I-1} \right) \times$$
$$g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \qquad [6.39]$$

and $\mathcal{L}(v)$, given by

$$\mathcal{L}(v) = \frac{\sqrt{\pi}}{2} \iiiint v_2 f_1^{d,I-1} f_2^{MB} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x}$$
$$+ \frac{\sqrt{\pi}}{2} \iiiint v_2 f_1^{MB} f_2^{d,I} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x}$$
$$+ \frac{\sqrt{\pi}}{2} \iiiint v_2 f_1^{d,I-1} f_2^{d,I} g\sigma\, d^2\Theta\, d^3\mathbf{c}_1\, d^3\mathbf{c}_2\, d^3\mathbf{x} \qquad [6.19]$$

In a "typical" collision event for which $\mathbf{c}_2$ lies in Ω_v (that is, for which v_2 is not necessarily zero), $\mathbf{c}_1$ will not be in element Ω_v. In this "typical" case, $\check{f}_1^{d,I} = f_1^{d,I-1}$ and $\check{f}_2^{d,I} = f_2^{d,I}$. Clearly, when one considers only "typical" collisions, $\mathcal{L}_{\text{modified}}(v)$ and $\mathcal{L}(v)$ are equal. Our approximation for $\mathcal{L}(v)$ (as compared to $\mathcal{L}_{\text{modified}}(v)$) lies in the case where both $\mathbf{c}_1$ and $\mathbf{c}_2$ lie in the element Ω_v; of course collisions of this type are expected to be have a small effect because g will be small in this case.

In short, our present iterative method can be viewed as a case of Newton's method, in which we retain (after some approximation) only the block-diagonal terms. Of course, the block diagonal terms in the Jacobian correspond to the terms that could change the value of the distribution function in the element Ω_v; we can compare this to the expression (6.9) in which only the change in the distribution function at a point affects the (approximation for the) collision integral at that point. Physically, this is reasonable; it corresponds to the assumption that modifying the distribution function in a region will have a greater effect on the collision integral within that region than elsewhere. The neglected terms (compared to the modified Newton's method) correspond to lower-probability collision events; in the limit where the volume of the element Ω_v goes to zero the neglected collision events will have essentially zero probability. In Chapter 7, we will discuss further improvements to this iterative algorithm.

We close by noting that using Newton's method provides a coupling among *all* of the shape function coefficients $\hat{f}_i^{d,I}$, this requires a large amount of storage, as well as the solution of a large linear system. Retaining only the block diagonal terms in the Jacobian effectively decouples the elements with different velocities, thus greatly simplifying the solution process.

6.7 Results

The primary advantage of the present iterative method is that it requires far fewer iterations to reach convergence than the number of timesteps needed to time-integrate to steady-state. This is illustrated in figure 6.2 using p_0 elements. (Verification of the

present iterative method using higher order elements is in progress.) Of particular interest is that, with the present iterative method, the convergence rate is (roughly) independent of the discretization needed. In contrast, using time integration, the number of timesteps required increases as a finer discretization is used. Figure 6.2 shows results using relatively coarse meshes; as the degree of refinement increases, the advantage of the iterative approach becomes more pronounced.

The number of iterations / timesteps required to reach steady state is not the only consideration in evaluating a method. As we are using Monte Carlo integration to evaluate the collision integral, the effect of statistical uncertainty in the collision integral on the statistical uncertainty of the solution needs to be evaluated. It was found that the (present) iterative method is much more sensitive to this noise than time integration; figure 6.3 shows the number of Monte Carlo samples needed to obtain a given level of convergence, but with the level of statistical uncertainty in the velocity fixed, for the iterative and the time integration method. We can see that the additional sensitivity of the iterative method means that more Monte Carlo samples are necessary, thus negating its advantage. We note that figure 6.3 is only for a particular refinement; assuming that the number of samples required to obtain a given degree of statistical uncertainty scale with the discretization in the same manner for the iterative and time-integration method, the advantage illustrated in figure 6.2 will lead to the iterative method being preferable for finer discretizations. Based on figure 6.3, we can estimate this crossover point at about 100 elements in the x-dimension or when the minimum element size is approximately 0.1. As one would typically use a nonuniform discretization, for example concentrating elements near the boundaries to capture the Knudsen layer, the crossover point at which the iterative method becomes preferable would occur at a smaller number of elements.

The present iterative method appears to be competitive with time integration for a range of problems, particularly when using finer discretizations; however, its higher sensitivity to noise diminishes its principal advantage – namely the fact that it converges in a very small number of iterations, with the number (roughly) independent of the discretization used. In Chapter 7, we will discuss some possible ways

to alleviate this problem. In particular, one could use the connection to Newton's method discussed in this chapter to devise an iterative method that converges in even fewer iterations. Additionally, there are several possibilities for reducing the degree of statistical uncertainty in the collision integral (even evaluating it by deterministic methods); this will allow better use of the faster convergence rate (measured in number of iterations) of the iterative method.

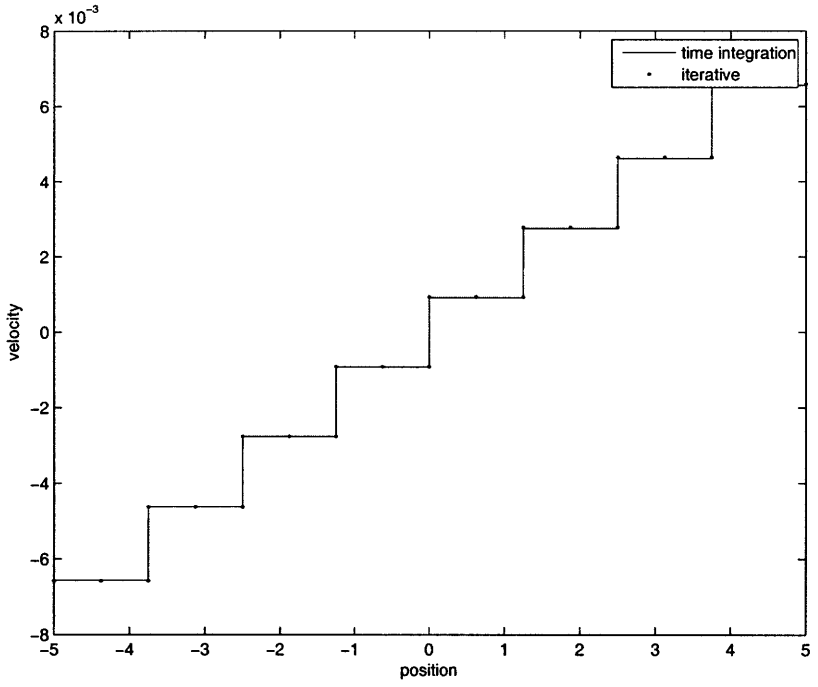

Figure 6.1: Comparison of steady state velocities for iterative and time-integration method. Results for 8^4 uniform p_0 elements with Kn = 0.1

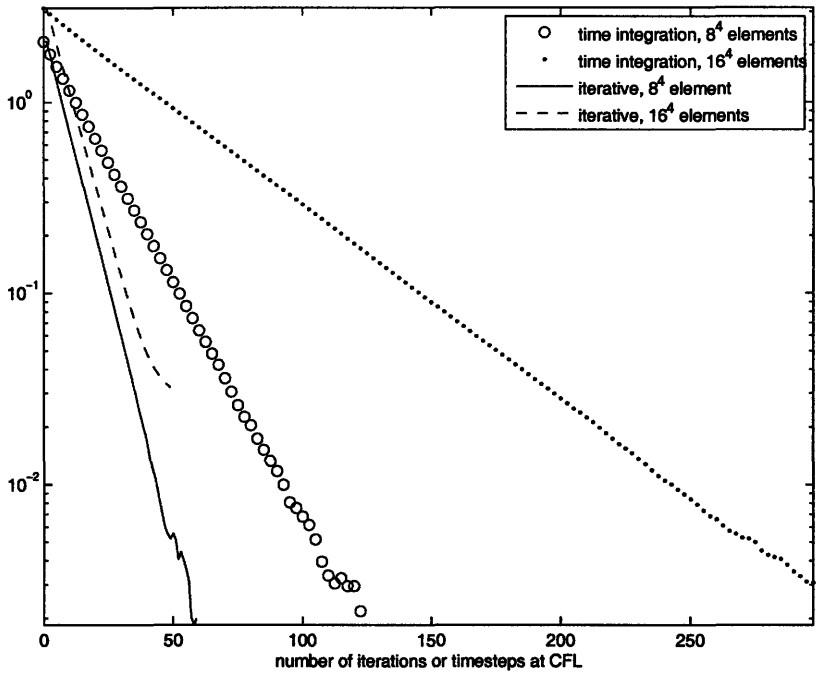

Figure 6.2: Comparison of convergence rates for iterative and time-integration method. RMS error level in the velocity is compared to the number of iterations (iterative method) or the number of timesteps at the CFL limit (time-integration). Results for 8^4 and 16^4 uniform p_0 elements with Kn = 0.1

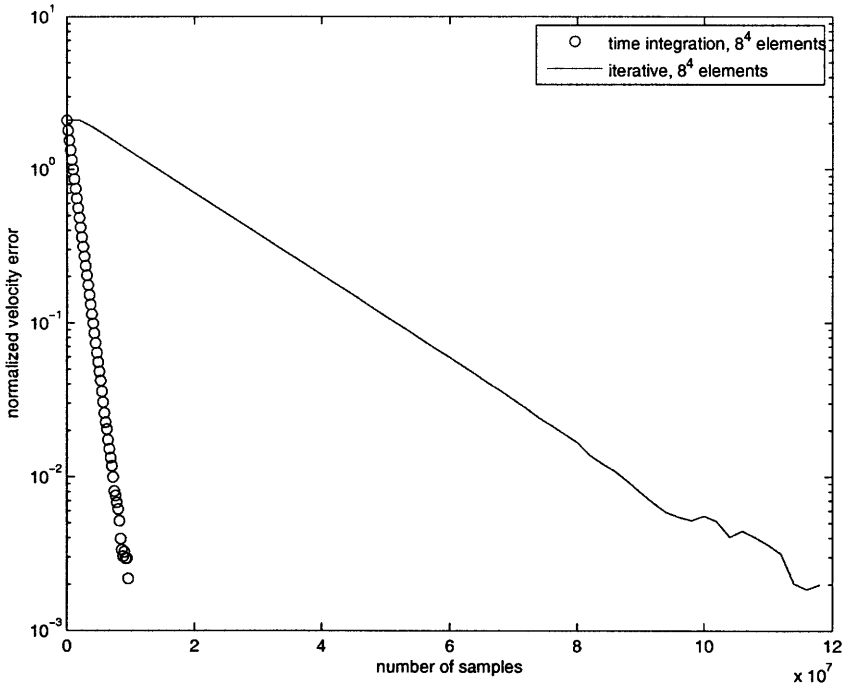

Figure 6.3: Comparison of convergence rates for iterative and time-integration method. RMS error level in the velocity is compared to the number of Monte Carlo samples necessary (for the same degree of statistical uncertainty for both iterative and time-integration methods). Results for 8^4 uniform p_0 elements with Kn = 0.1

Chapter 7

Conclusions

In this thesis, we have shown how numerical methods that are highly efficient for simulating low-signal dilute gas flows can be developed through the use of appropriate variance reduction techniques for evaluating the collision integral. In particular, control variate integration, manifested as simulating only the deviation from equilibrium, was combined with importance sampling to yield two independent methods – a particle simulation approach and PDE-based approach – for solving the Boltzmann equation. Both methods were shown to be capable of providing efficient and accurate solutions for a wide variety of testcases in zero and one physical dimension. Additionally, due to the general nature of the methods developed, it is expected that both can be extended to general flows in two or three physical dimensions.

Much of the efficiency of the present methods derives from simulating the deviation from equilibrium, and the fact that this deviation is typically small in cases of interest. However, the methods are applicable regardless of the magnitude of this deviation; the computational efficiency will be lower if the deviation is large, but the methods are still accurate. Additionally, though the hard sphere interaction potential is emphasized in this work, the use of other potentials with these methods is straightforward.

The particle method presented in Chapter 4 can be viewed as a direct extension to DSMC; the method is modified only in ways necessary to allow the simulation of the deviation from equilibrium. In the formulation presented in this thesis, the collision process leads to a growth in the number of particles, thus a cancellation routine was

required. This cancellation introduced an additional level of discretization in velocity space, and an associated degree of numerical error. The necessity for cancellation has been remedied in the work of [24], which has extended the present method by allowing for the underlying equilibrium distribution to change as a function of physical space and time.

The discontinuous Galerkin approach presented in Chapter 5 uses a well-known and extensively studied formulation for hyperbolic equations to discretize the Boltzmann equation [17]. The collision integral is accurately and efficiently evaluated using Monte Carlo integration combined with the variance reduction techniques presented in Chapter 2. As with the particle method, the DG approach provides an efficient and accurate means of obtaining low noise solutions of the Boltzmann equation.

Additionally, an iterative steady-state formulation within the DG framework is presented in Chapter 6. This has the potential of providing much more efficient solutions when resolving transient phenomena is not necessary. The method presented here is a natural extension of previous iterative methods for the Boltzmann equation to the present DG framework; this approach is shown to be a variant of Newton's method.

7.1 Comparison of direct and particle-based methods

As two different realizations of the same fundamental idea have been described in this thesis, the question arises as to which method is better. Naturally, the answer to this question will depend on the application of interest (and the possibility of future extensions to the present work). However, at present it appears that the particle formulation holds the advantage for high Knudsen numbers, and the DG method for lower Knudsen numbers (and possibly when very high levels of accuracy are required).

The particle formulation naturally and accurately handles discontinuities as a direct consequence of simulating the (deviational) distribution function with a set of

particles. In particular, discontinuities in velocity space are 'exactly' captured by the particle approach. It is difficult to make a concrete statement about discontinuities in physical space because discretization is introduced by the collision and cancellation process; however, the effect of intermolecular collisions is expected to be small in the high Knudsen number limit.

While the DG formulation can manage discontinuities, it is not as effective as the particle method. However, for smaller Knudsen numbers, discontinuities in the distribution function are expected to be of less consequence, and the ability to precisely represent smooth functions with the DG formulation are expected to make this method more effective. Additionally, in the limit where very high accuracy is required, the higher order convergence rate of the DG method might again prove to be a deciding advantage. We do need to point out that the statistical uncertainty in the collision integral negates some of this advantage, to the extent that even though the benefit from higher order approximations is evident (see figures 5.8 and 5.9), the asymptotic convergence rates can not be observed (see discussion for figure 5.1).

7.2 Directions for future work

7.2.1 Direct evaluation of the collision integral

While the Monte Carlo methods discussed in this thesis are highly efficient, they are still bound to a $N^{-1/2}$ convergence rate. For problems where extremely low numerical error is required, perhaps direct numerical evaluation of the collision integral using standard quadrature techniques may be a better approach. Possible candidates for this analysis include the linearized form of the collision integral for hard spheres [37, 14] in which the dependence on the scattering angle Θ can be integrated analytically (leading to a lower dimensionality). Although such an approach may present a significantly more complex algorithm and a higher computational overhead, provided the computational resources are available it may be competitive with the approaches proposed here when very low error levels are required.

If such a method were developed, the method could be extended beyond the linearized Boltzmann equation for hard spheres using control-variate integration. In essence, instead of solving for the deviation from an equilibrium distribution, as was done in this thesis, one would use a Monte Carlo approach to solve for the difference between the true solution and the solution obtained via the linearized Boltzmann equation for hard spheres. As before, if the linearized solution were a good approximation, this method would be highly efficient, while remaining valid in the general case.

7.2.2 Control variate using the deviation from a non-equilibrium distribution

The control variate approach used in this thesis does not necessarily have to be based on an equilibrium distribution; it only requires knowledge of a distribution (that presumably approximates the true distribution function) for which the collision integral is analytically known, or easily evaluated. For low Knudsen number flows, one expects Chapman-Enskog distribution function [20] (which corresponds to the Navier-Stokes equations) to provide a better approximation to the distribution function than a Maxwell Boltzmann distribution would. Additionally, the collision integral for the Chapman-Enskog distribution can be easily calculated. This yields the possibility of a hierarchical system for solving the Boltzmann equation – one solves the Navier-Stokes equations, then uses the techniques developed in this thesis to solve for the difference between the Boltzmann solution and the Navier-Stokes solution.

7.2.3 Improved iterative techniques

The iterative method presented in Chapter 6 can be much more efficient than time integration to steady state in some cases, but still has the potential for improvement. For example, the number of iterations required will be lower if the initial distribution is close to the correct distribution; one could use a solution based on a coarser mesh or lower order polynomial to obtain a good value for this initial distribution. Another

possibility is to use an alternative importance sampling formulation for the evaluation of (6.25) and (6.26); this has the potential to avoid division by $\left|f^d\right|$ and to reduce the degree of statistical uncertainty.

We have also noted that the iterative approach presented in this thesis neglects many terms in the Jacobian matrix in order to obtain a manageable set of equations – it is currently infeasible to include all terms. The terms not included are, in essence, set to zero. However, it is possible that a small set of the terms currently neglected could be included. For example, the $J_{\phi_0,v}$ terms, where ϕ_0 is the p_0 shape function, would likely be more significant than the others. One could also attempt to devise another useful low-rank approximation for this matrix. A better approximation for the Jacobian matrix could have significant consequences. In particular, we note that for low-speed flows, we expect the deviational terms to enter linearly[1] in equation (6.31); in other words, Newton's method utilizing the full Jacobian matrix should yield extremely rapid convergence. Of course, using a better approximation for the Jacobian matrix will lead to a (much) larger set of equations to be solved – one will no longer be able to solve independently for each velocity – however iterative solvers for linear sets of equations have the potential to do this efficiently.

[1]The $f_1^d f_2^d$ term will typically be small.

Appendix A

Shape functions

In this chapter, we briefly describe the shape functions used in this thesis. We refer the reader to any finite element reference (i.e. [26]) for a full discussion of shape functions; this appendix is only included for completeness.

The shape functions used in this thesis are tensor products of Legendre polynomials (although the implementation used does not depend on this choice of shape functions). That is, the shape functions can be written in the form

$$\phi(\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6) = \psi_{n_1}(\xi_1) \times \psi_{n_2}(\xi_2) \times \psi_{n_3}(\xi_3) \times \psi_{n_4}(\xi_4) \times \psi_{n_5}(\xi_5) \times \psi_{n_6}(\xi_6) \quad \text{(A.1)}$$

on the *reference* element. Here ψ_{n_i} is the Legendre polynomial of order n_i – that is the order of the Legendre polynomial used in the i^{th} dimension. Different values for the various n_i give rise to different shape functions.

Because the Legendre polynomials are defined on the range $[-1, 1]$, we have defined the shape functions on a reference element; the range of ξ_i in our reference element is defined as $[-1, 1]$. We can introduce a mapping between an element in our mesh and this reference element. For the particular case of rectangular elements (as used in this thesis), this mapping will be linear, and one can map (x, y, z, c_x, c_y, c_z) onto (ξ_1, ξ_2, ξ_3, ξ_4 ,ξ_5, ξ_6) respectively. For example, if the value for x in a particular element

ranged from $x_{\min}$ to $x_{\max}$, we would have the mapping

$$\xi_1 = 2\frac{x - x_{\min}}{x_{\max} - x_{\min}} - 1 \tag{A.2}$$

Of course, in the general case this mapping is not so simple and one can not typically map each dimension in phase space onto a single coordinate ξ_i.

The Legendre polynomials are well known; the first few are

$$\psi_0(\xi) = 1 \tag{A.3a}$$

$$\psi_1(\xi) = \xi \tag{A.3b}$$

$$\psi_2(\xi) = \frac{1}{2}\left(3\xi^2 - 1\right) \tag{A.3c}$$

$$\psi_3(\xi) = \frac{1}{2}\left(5\xi^3 - 3\xi\right) \tag{A.3d}$$

As tensor-product shape functions are used in the present work, a sum-factorization technique [26] can be used to efficiently evaluate the sums arising from Gaussian quadrature. However, in generating the random samples for the collision integral (see section 5.1.1 or appendix B) we can not use this approach; thus it may be more efficient to use a non-tensor product basis, as this could be constructed using fewer shape functions.

Appendix B

Generating collision samples for DG collision integral

This sections describes how one can efficiently generate random samples from the distribution $|f_1^d|f_2^{MB}/\chi_{MB,d}$ and $|f_1^d||f_2^d|/\chi_{d,d}$ for use in evaluating the collision integral (5.13). This implementation is used in the present work, however it is certainly not the only possibility. Generating these random samples consumes a significant fraction of the computational time; a more efficient algorithm would significantly improve the overall computational cost of the DG implementation.

For simplicity, this algorithm assumes that we are computing the collision integral for a set of cells with the *same* extent in physical space that, as a set, span velocity space. We also assume that f^{MB} is only a function of velocity.

We will use a combination of the alias method [13], which is a method for generating points from a discrete distribution in *constant* time (with a linear setup time), and the acceptance-rejection method [32] to correctly generate points from our continuous probability distribution[1]. We use the alias method to generate samples from a function that bounds our desired distribution from above, and then perform acceptance-rejection on these samples to obtain the correct distribution.

[1] A more efficient algorithm for this process could lead to a significant improvement of the overall speed of the code.

We begin with the distribution $|f_1^d| f_2^{MB} / \chi_{MB,d}$, where

$$\chi_{MB,d} = 4\pi \left(\int f_2^{MB} d^3\mathbf{c}_2 \right) \iint |f_1^d| d^3\mathbf{c}_1 d^3\mathbf{x} \tag{B.1}$$

1. We first generate an upper bound for $|f^d|$ in each element; let us denote this by $f^d_{max,k}$ where k indexes the element. As we use tensor products of Legendre polynomials, each shape function has a maximum magnitude of 1, so $f^d_{max,k} = \sum_i \left|\left(f_i^d\right)_k\right|$ is an upper bound. (In general, a tighter upper bound would be preferable.)

2. Loop:

   (a) We then use the alias method[13] to randomly pick a cell with a probability proportional to $C_k f_{max,k}$ where C_k is the volume of the cell k in *velocity* space.

   (b) A random point in phase space $\{\mathbf{x}, \mathbf{c}_1\}$ within the cell k is chosen using a uniform probability distribution

   (c) This point is either accepted with a probability $|f^d(\mathbf{x}, \mathbf{c}_1)| / f^d_{max,k}$ or rejected and a new cell is chosen and the process repeats.

3. Generate three Gaussian random numbers to find $\mathbf{c}_1$ [32].

4. Generate Θ on the unit sphere [2].

The distribution $|f_1^d||f_2^d|/\chi_{d,d}$ is slightly more complex because both f_1^d and f_2^d have a dependence on $\mathbf{x}$.

1. We use the same upper bound for $|f^d|$ in each element as in the previous algorithm.

2. Loop:

   (a) We then use the alias method to randomly pick a two cells independently with a probability $C_k f_{max,k}$. Let us denote the indices of the two cells selected by k_1 and k_2.

   (b) The vector $\{\mathbf{x}, \mathbf{c}_1, \mathbf{c}_2\}$ is chosen from a uniform distribution such that $\mathbf{c}_1$ is in cell k_1, $\mathbf{c}_2$ is in cell k_2 and $\mathbf{x}$ is in both cell k_1 and cell k_2 (recall that we have assumed that all cells have the same spatial extent).

   (c) This (entire vector) $\{\mathbf{x}, \mathbf{c}_1, \mathbf{c}_2\}$ is accepted with a probability $|f^d(\mathbf{x}, \mathbf{c}_1)|/f^d_{max,k_1} \times |f^d(\mathbf{x}, \mathbf{c}_2)|/f^d_{max,k_2}$ or (the entire vector is) rejected and a new *pair* of cells is chosen and the process repeats.

3. Generate Θ on the unit sphere.

Bibliography

[1] H. A. Al-Mohssen, N. G. Hadjiconstantinou, and I. Kevrekidis. Acceleration methods for course-grained numerical solution of the Boltzmann equation. In *Proceedings of ICNMM2006 Fourth International Conference on Nanochannels, Microchannels and Minichannels*, 2006.

[2] F. J. Alexander and A. L. Garcia. The direct simulation Monte Carlo method. *Computers in Physics*, 11(6):588 – 593, 1997.

[3] K. Aoki, S. Takata, H. Aikawa, and F. Golse. A rarefied gas flow caused by a discontinuous wall temperature. *Physics of Fluids*, 13(9):2645 – 2661, 2001.

[4] V. V. Aristov. *Direct Methods for Solving the Boltzmann Equation and Study of Nonequilibrium Flows.* Kluwer Academic Publishers, Boston, 2001.

[5] L. L. Baker. Efficient numerical solutions of the Boltzmann equation for low-speed flows. Master's thesis, Massachusetts Institute of Technology, February 2004.

[6] L. L. Baker and N. G. Hadjiconstantinou. Implicit hybrid simulation framework for steady-state dilute gas flows. *International Journal for Multiscale Computational Engineering*, 3:49–58, 2005.

[7] L. L. Baker and N. G. Hadjiconstantinou. A variance reduction approach for Monte Carlo solutions of the non-linear Boltzmann equation. In *Proceedings of the Third International conference on Microchannels and Minichannels.* ASME, 2005.

[8] L. L. Baker and N. G. Hadjiconstantinou. Variance reduction for Monte Carlo solutions of the Boltzmann equation. *Physics of Fluids*, 17(051703), 2005.

[9] L. L. Baker and N. G. Hadjiconstantinou. Variance reduction in particle methods for solving the Boltzmann equation. In *Proceedings of ICNMM2006 Fourth International Conference on Nanochannels, Microchannels and Minichannels*, 2006.

[10] L. L. Baker and N. G. Hadjiconstantinou. Efficient numerical solutions of the Boltzmann equation: A discontinuous Galerkin approach. *submitted to Physics of Fluids*, 2007.

[11] L. L. Baker and N. G. Hadjiconstantinou. Variance-reduced particle methods for solving the Boltzmann equation. *Journal of Computational and Theoretical Nanoscience*, to appear.

[12] G. A. Bird. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows.* Clarendon Press, Oxford, 1998.

[13] P. Bratley, B. L. Fox, and L. E. Schrage. *A Guide to Simulation.* Springer-Verlag, New York, second edition, 1987.

[14] C. Cercignani. *The Boltzmann Equation and its Applications.* Springer-Verlag, New York, 1988.

[15] C. Cercignani and A. Daneri. Flow of a rarefied gas between two parallel plates. *Journal of Applied Physics*, 34(12), 1963.

[16] J. Chun and D. L. Koch. A direct simulation Monte Carlo method for rarefied gas flows in the limit of small Mach number. *Physics of Fluids*, 17(107107), 2005.

[17] B. Cockburn and C. Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16(3), 2001.

[18] G. S. Fishman. *A First Course in Monte Carlo.* Duxbury, 2006.

[19] A. L. Garcia. *Numerical Methods for Physics.* Prentice Hall, New Jersey, second edition, 2000.

[20] A. L. Garcia and A. J. Alder. Generation of the Chapman-Enskog distribution. *Journal of Computational Physics*, 140:66 – 70, 1998.

[21] N. G. Hadjiconstantinou. The limits of Navier-Stokes theory and kinetic extensions for describing small-scale gaseous hydrodynamics. *Physics of Fluids*, 18(111301), 2006.

[22] N. G. Hadjiconstantinou, A. L. Garcia, M. Z. Bazant, and G. He. Statistical error in particle simulations of hydrodynamic phenomena. *Journal of Computational Physics*, 187:274 – 297, 2003.

[23] T. M. M. Homolle. Efficient particle methods for solving the Boltzmann equation. Master's thesis, Massachusetts Institute of Technology, February 2007.

[24] T. M. M. Homolle and N. G. Hadjiconstantinou. Low-variance deviational simulation Monte Carlo. *Physics of Fluids*, 19(041701), 2007.

[25] K. Huang. *Statistical Physics.* Taylor and Francis, New York, 2001.

[26] G. E. Karniadakis and A. J. Sherwin. *Spectral / hp Element Methods for CFD.* Oxford University Press, New York, second edition, 1999.

[27] A. R. Krommer and C. W. Ueberhuber. *Computational Integration.* SIAM, 1998.

[28] M. Krook and T. T. Wu. Exact solutions of the Boltzmann equation. *Physics of Fluids*, 20(10):1589 – 1595, 1977.

[29] A. M. Law and W. D. Kelton. *Simulation Modeling and Analysis.* McGraw-Hill, Inc., New York, second edition, 1991.

[30] A. Nordsieck and B. L. Hicks. Monte Carlo evaluation of the Boltzmann collision integral. *Rarefied Gas Dynamics*, 1:695 – 710, 1967.

[31] T. Ohwada, Y. Sone, and K Aoki. Numerical analysis of the Poiseuille and thermal transpiration flows between two parallel plates on the basis of the Boltzmann equation for hard-sphere molecules. *Physics of Fluids*, 1:2042 – 2049, 1989.

[32] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, New York, second edition, 2002.

[33] F. Reif. *Fundamentals of Statistical and Thermal Physics.* McGraw-Hill, Boston, 1965.

[34] Y. Sone. *Kinetic Theory and Fluid Dynamics.* Birkhauser, 2002.

[35] Y. Sone, T. Ohwada, and K. Aoki. Temperature jump and Knudsen layer in a rarefied gas over a plane wall: Numerical analysis of the linearized Boltzmann equation for hard-sphere molecules. *Physics of Fluids A*, 5(1):363 – 370, 1989.

[36] Z. Tan and P. L. Varghese. The $\Delta - \epsilon$ method for the Boltzmann equation. *Journal of Computational Physics*, 110:327–340, 1994.

[37] W. G. Vincenti and C. H. Kruger. *Physical Gas Dynamics.* Krieger Publishing Company, Florida, first edition, 1986.

[38] W. Wagner. A convergence proof for Bird's direct simulation Monte Carlo method for the Boltzmann equation. *Journal of Statistical Physics*, 66(3–4):1011–1044, 1992.

[39] S. M. Yen. Numerical solution of the nonlinear Boltzmann equation for nonequilibrium gas flow problems. *Annual Review of Fluid Mechanics*, 17:67 – 97, 1984.

[40] S. M. Yen, B. L. Hick, and R. M. Osten. Further developments of a Monte Carlo method for evaluation of Boltzmann collision integrals. In M. Becker and M. Fiebig, editors, *Ninth International Symposium on Rarefied Gas Dynamics*, pages A.12–1 – A.12–10, 1974.