EFFICIENT NUMERICAL METHODS FOR KINETIC EQUATIONS WITH HIGH DIMENSIONS AND UNCERTAINTIES

by

Yubo Wang

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

Department of Mathematics

West Lafayette, Indiana

December 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Jingwei Hu, Chair

Department of Mathematics

Dr. Guang Lin, co-Chair

Department of Mathematics

Dr. Jianlin Xia

Department of Mathematics

Dr. Haizhao Yang

Department of Mathematics

Approved by:

Dr. Plamen Stefanov, Associate Head for Graduate Studies

To my family.

ACKNOWLEDGMENTS

This journey would not have been possible without the support and help of my advisor, collaborators, committee professors, my family and friends.

First and foremost, I would like to express the deepest appreciation to my advisor Prof. Jingwei Hu for introducing me to the world of kinetic equations. Even though my initial background in scientific computing was not strong, she guided me with great patience and supports in all the time of research. Her motivation, enthusiasm and immense knowledge also inspired me and sharpened my thinking to bring the work to a higher level. Furthermore, I would like to thank for her financial support, which helped me in focus on researching with all my effort.

Besides my advisor, I would also like to show gratitude to my collaborator, Prof. Lorenzo Pareschi and Prof. Lukas Einkemmer for their insightful comments and invaluable suggestions.

My sincere thanks also goes to Prof. Guang Lin, Prof. Jianlin Xia and Prof. Haizhao Yang for being my advisory committee members and giving me precious feedbacks.

I am also extremely grateful to the Department of Mathematics and Purdue University for physical and technical supports throughout my PhD study. Without their support and funding, this thesis could not have reached its goal. I would say a special thank to the TA program which improved my teaching and presenting skills greatly.

Last but not least, none of this could have happened without my family and I must express my very profound gratitude to my parents, Xiaoping Wang and Sixin Wang, and my spouse, Siqi Liang. To my parents, I want to thank for supporting me spiritually throughout my life. Their unconditional love and encouragements are with me in whatever I pursue. Most importantly, a huge thank to my spouse for accompanying me on this journey with ups and downs. She kept me going and providing me with endless supports.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

In this thesis, we focus on two challenges arising in kinetic equations, high dimensions and uncertainties. To reduce the dimensions, we proposed efficient methods for linear Boltzmann and full Boltzmann equations based on dynamic low-rank frameworks. For linear Boltzmann equation, we proposed a method that is based on macro-micro decomposition of the equation; the low-rank approximation is only used for the micro part of the solution. The time and spatial discretizations are done properly so that the overall scheme is second-order accurate (in both the fully kinetic and the limit regime) and asymptotic-preserving (AP). That is, in the diffusive regime, the scheme becomes a macroscopic solver for the limiting diffusion equation that automatically captures the low-rank structure of the solution. Moreover, the method can be implemented in a fully explicit way and is thus significantly more efficient compared to the previous state of the art. We demonstrate the accuracy and efficiency of the proposed low-rank method by a number of four-dimensional (two dimensions in physical space and two dimensions in velocity space) simulations. We further study the adaptivity of low-rank methods in full Boltzmann equation. We proposed a highly efficient adaptive low-rank method in Boltzmann equation for computations of steady state solutions. The main novelties of this approach are: On one hand, to the best of our knowledge, the dynamic low-rank integrator hasn't been applied to full Boltzmann equation till date. The full collision operator is local in spatial variable while the convection part is local in velocity variable. This separated nature is well-suited for low-rank methods. Compared with full grid method (finite difference, finite volume,...), the dynamic low-rank method can avoid the full computations of collision operators in each spatial grid/elements. Resultingly, it can achieve much better efficiency especially for some low rank flows (e.g. normal shock wave). On the other hand, our adaptive low-rank method uses a novel dynamic thresholding strategy to adaptively control the computational rank to achieve better efficiency especially for steady state solutions. We demonstrate the accuracy and efficiency of the proposed adaptive low rank method by a number of 1D/2D Maxwell molecule benchmark tests.

On the other hand, for kinetic equations with uncertainties, we focus on non-intrusive sampling methods where we are able to inherit good properties (AP, positivity preserving)

from existing deterministic solvers. We propose a control variate multilevel Monte Carlo method for the kinetic BGK model of the Boltzmann equation subject to random inputs. The method combines a multilevel Monte Carlo technique with the computation of the optimal control variate multipliers derived from local or global variance minimization problems. Consistency and convergence analysis for the method equipped with a second-order positivity-preserving and asymptotic-preserving scheme in space and time is also performed. Various numerical examples confirm that the optimized multilevel Monte Carlo method outperforms the classical multilevel Monte Carlo method especially for problems with discontinuities.

1. INTRODUCTION

Kinetic equations play an important role in describing the non-equilibrium dynamics of gas or systems comprised of large number of particles from a statistical viewpoint [1]. At the mesoscopic level, they can explain the macroscopic quantities and provide rich information when the well-known fluid mechanical laws of Navier-Stokes and Fourier break down and become inadequate to represent the system. There are various applications in fields such as rarefied gas dynamics [2], plasma physics [3], semiconductor modeling [4] and biological and social sciences [5]. The most fundamental example of kinetic equation, Boltzmann equation [6] bridges microscopic Newtonian mechanics and macroscopic continuum mechanics by taking into account of particle transport and binary collisions. Denote the probability distribution function by $f(t, \mathbf{x}, \mathbf{v})$, where t is time, $\mathbf{x}$ is space, and $\mathbf{v}$ is (particle) velocity. The dimensionless Boltzmann equation reads [1], [7]:

$$\partial_t f + \mathbf{v} \cdot \nabla_{\mathbf{x}} f = \frac{1}{\varepsilon} \mathcal{Q}(f, f), \quad \mathbf{x} \in \Omega_{\mathbf{x}} \subset \mathbb{R}^{d_x}, \ \mathbf{v} \in \mathbb{R}^{d_v}, \tag{1.0.1}$$

where ε is the Knudsen number, defined as the ratio of the mean free path and the typical length scale. ε varies from $\mathcal{O}(1)$, the kinetic regime to $\varepsilon \ll 1$, the fluid regime. When $\varepsilon \to 0$, formally by Chapman-Enskog expansion, one can derive the compressible Euler limit or Compressible Navier-Stokes limit depending on the leading order taken; d_x and d_v are the dimensions of spatial and velocity domain respectively; and $\mathcal{Q}(f, f)$, is the collision operator, a high-dimensional, nonlinear, non-local quadratic integral operator acting only in the velocity space:

$$\mathcal{Q}(g, f) = \int_{\mathbb{R}^{d_v}} \int_{S^{d_v - 1}} B_\sigma(|\mathbf{v} - \mathbf{v}_*|, \cos\chi)[g(\mathbf{v}'_*) f(\mathbf{v}') - g(\mathbf{v}_*) f(\mathbf{v})] \, \mathrm{d}\sigma \, \mathrm{d}\mathbf{v}_*, \tag{1.0.2}$$

where $f(\mathbf{v})$ and $f(\mathbf{v}')$ are short for $f(t, \mathbf{x}, \mathbf{v})$ and $f(t, \mathbf{x}, \mathbf{v}')$ (similarly for $g(\mathbf{v}'_*)$ and $g(\mathbf{v}_*)$); the post-collisional velocities $(\mathbf{v}', \mathbf{v}'_*)$ are defined in terms of pre-collisional velocities $(\mathbf{v}, \mathbf{v}_*)$ through the conservation of momentum and energy during the collision:

$$\mathbf{v}' = \frac{\mathbf{v} + \mathbf{v}_*}{2} + \frac{|\mathbf{v} - \mathbf{v}_*|}{2}\sigma, \quad \mathbf{v}'_* = \frac{\mathbf{v} + \mathbf{v}_*}{2} - \frac{|\mathbf{v} - \mathbf{v}_*|}{2}\sigma, \tag{1.0.3}$$

with σ being a vector over the unit sphere S^{d_v-1}; and B_σ is the Boltzmann collision kernel, a non-negative function that depends on $|\mathbf{v}-\mathbf{v}_*|$ and cosine of deviation angle χ (angle between $\mathbf{v}-\mathbf{v}_*$ and $\mathbf{v}'-\mathbf{v}'_*$): $\cos\chi = \sigma \cdot \widehat{\mathbf{v}-\mathbf{v}_*}$.

It can be shown that $\mathcal{Q}(f,f)$ satisfies the conservation of mass, momentum, and energy. Denote $\Phi(\mathbf{v}) = \left[1, \mathbf{v}, \frac{1}{2}|\mathbf{v}|^2\right]^T$ by collision invariants, then there hold the conservation properties:

$$\int_{\mathbb{R}^{d_v}} \mathcal{Q}(f,f)\Phi(\mathbf{v})\,\mathrm{d}\mathbf{v} = 0 \tag{1.0.4}$$

$$\int_{\mathbb{R}^{d_v}} f(t,\mathbf{x},\mathbf{v})\Phi(\mathbf{v})\,\mathrm{d}\mathbf{v} = \begin{bmatrix} \rho(t,\mathbf{x}) \\ \mathbf{m}(t,\mathbf{x}) \\ E(t,\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \rho(t,\mathbf{x}) \\ \rho(t,\mathbf{x})\boldsymbol{U}(t,\mathbf{x}) \\ \frac{d_v}{2}\rho(t,\mathbf{x})T(t,\mathbf{x}) + \frac{1}{2}\rho(t,\mathbf{x})|\boldsymbol{U}(t,\mathbf{x})|^2 \end{bmatrix}, \tag{1.0.5}$$

where $\rho(t,\mathbf{x})$, $\boldsymbol{U}(t,\mathbf{x})$, $T(t,\mathbf{x})$, $\mathbf{m}(t,\mathbf{x})$ and $E(t,\mathbf{x})$ are the density, velocity, temperature, momentum and total energy at time t and position $\mathbf{x}$.

Yet the numerical computations of full Boltzmann equation (1.0.1) pose great challenges, which is mainly due to multidimensional structure of collision operators ($(2d_v-1)$-fold integral). Historically, there are stochastic and deterministic approaches for the numerical computations of collision operators. Stochastic methods are mainly based on the direct simulation Monte Carlo (DSMC) method [8], [9] and are widely used due to the fact that it can avoid the curse of dimensionality. However, it becomes extremely expensive to avoid slow convergence and fluctuations in results in certain cases near the continuum-fluid regime, especially when Mach number is small.

Deterministic methods rely on discretizations of governing differential equations on representative grids and have undergone considerable developments [10]. One of the most popular methods is represented by the discrete velocity models of the Boltzmann equation. This method uses a fixed set of discrete velocity quadrature points to approximate the continuous velocity space [11], [12]. Another approach is the Fourier spectral methods [13], [14]. They compute the collision operators in the frequency domain using Fourier transform technique.

These methods can not only possess spectral accuracy, but also can reduce computational complexity through fast Fourier methods.

Additionally, one can avoid the complexity of Boltzmann collision operator $\mathcal{Q}(f,f)$ by introducing simplified full Boltzmann equation variants. The BGK model, initially proposed by Bhatnagar, Gross, and Krook [15], has been widely used in many disciplines of science and engineering [2], [5], [16]. It simplifies the full Boltzmann binary collision operator yet possesses most of its key properties. In dimensionless form, the equation reads:

$$\partial_t f + \mathbf{v}\cdot\nabla_{\mathbf{x}} f = \frac{1}{\varepsilon}\left(\mathcal{M}[\rho,\boldsymbol{U},T]-f\right), \quad \mathbf{x}\in\Omega_{\mathbf{x}}\subset\mathbb{R}^{d_x},\ \mathbf{v}\in\mathbb{R}^{d_v}, \tag{1.0.6}$$

where ε is the Knudsen number, consistent with the one in full Boltzmann equation(1.0.1); $\mathcal{M}[\rho,\boldsymbol{U},T]$ is the so-called Maxwellian function given by

$$\mathcal{M}[\rho,\boldsymbol{U},T](t,\mathbf{x},\mathbf{v}) = \frac{\rho(t,\mathbf{x})}{(2\pi T(t,\mathbf{x}))^{\frac{d_v}{2}}}\exp\left(-\frac{|\mathbf{v}-\boldsymbol{U}(t,\mathbf{x})|^2}{2T(t,\mathbf{x})}\right), \tag{1.0.7}$$

where $\rho,\boldsymbol{U},T$ are the density, velocity and temperature defined from (1.0.5). BGK model preserves the compressible Euler limit as $\varepsilon\to 0$. It satisfies the conservation property (1.0.5) and

$$\int_{\mathbb{R}^{d_v}} M[f](t,\mathbf{x},\mathbf{v})\Phi(\mathbf{v})\,\mathrm{d}\mathbf{v} = \int_{\mathbb{R}^{d_v}} f(t,\mathbf{x},\mathbf{v})\Phi(\mathbf{v})\,\mathrm{d}\mathbf{v} \tag{1.0.8}$$

There are other variants of simplified Boltzmann equation such as linearized Boltzmann (LB) [17] and ES-BGK model [18] that consider different ways to simplify the full Boltzmann collision operators. However, the multi-dimensional nature still raise huge challenges in computational complexities ($(d_x+d_v$ dimensions in phase space)). For spatial discretization, for example in full Boltzmann equation (1.0.1), historically there are works with finite volume methods(FVM), finite difference methods(FDM) and finite element method [19]–[22]. Recently, the discontinuous Galerkin (DG) method are also applied in Boltzmann equations to achieve high-order accuracy [23], [24]. However, these methods are all rely on the full grid simulation (the collision operator need to evaluated at every spatial grid point or element), which could be computational expensive as mesh refine.

To deal with the expensive multi-dimensional structure, recently, a class of dynamic low rank method have been applied to solving kinetic equations including the Vlasov equation [25], [26], Boltzmann-BGK equation [27], [28] and radiation transfer equation [29]. Employed with the projection splitting approach (see, e.g.[30]), the numerical integrator approximates the solution in a low rank manifold consisting of spatial and velocity basis. In this way, it can reduce the computational complexity significantly because operators only need to be evaluated in low rank manifold for every spatial/velocity basis.

Another challenging area of kinetic equations lies in the uncertain initial and boundary conditions. During the last decade, research on kinetic theory are mainly focused on the deterministic part, both theoretically and numerically [7], [10], [31] on Boltzmann equation and related kinetic models, while the uncertainty part are ignored. However, in reality, uncertainties may arise in initial/boundary conditions and parameters for these kinetic equations because of incomplete knowledge and imprecise measurement. Recently, there has been a significant interest to study the impact of random inputs to the kinetic equations, see [32] for an overview. To quantify the uncertainties mentioned above, works in solving kinetic equations are mainly based on the generalized polynomial chaos based stochastic Galerkin (gPC-sG) approximation, which has been successfully applied to many physical and engineering problems, see for instance, the overviews in [33], [34]. The gPC-sG method, essentially a spectral method in the random domain, yields to a large deterministic systems of equations. However, lack of regularity poses a serious problem in the loss of hyperbolicity of the resulting gPC-sG system[35]. Despite that these deterministic methods show some promise, they suffer from the disadvantage that they are highly intrusive. On one hand, existing codes for computing the deterministic kinetic problems need to be completely reconfigured to implement the gPC based method. On the other hand, intrusiveness may induce some bad approximations even for deterministic solvers with good properties. Due to Gibb's phenomenon, gPc based methods may induce approximations with negative density for problems with discontinuity, where traditional deterministic solver fails for this case. Moreover, for kinetic equations with high nonlinearity, like the BGK model, the Maxwellian (1.0.7) distribution function need to be reconstructed repeatedly at each step, which is cumbersome.

Another class of methods, statistical sampling methods, most notably based on Monte Carlo(MC) sampling, are also widely used for computational uncertainty quantification in numerical solutions of PDEs. The non-intrusiveness enables the approximation solutions to inherit properties, like positivity preserving, of existing deterministic kinetic solvers, which makes the parallel computing feasible for implementation. For discontinuous solutions, MC type methods can help prevent from Gibb's phenomenon. However, the asymptotic convergence rate $N_{\text{sample}}^{-\frac{1}{2}}$ is non-improvable by the central limit theorem, where N_{sample} is number of samples. Variance reduction technique can be adopted like the Multilevel Monte Carlo(MLMC) method [36], where the approximation of statistical expectation breaks up into telescopic sums of expectations of consecutive mesh size, see [37] for an application in scalar hyperbolic conservation law with random initial data. Moreover, as an improvement of MLMC method, the control variate MC method, see from [38],made use of different asymptotic models and reduced the variance by introducing a new parameter in MLMC.

The first contribution of this thesis is to develop an efficient numerical method to solve the linear transport equation based on dynamic low rank approximation to achieve increased computational efficiency as well as drastic reduction of memory. An additional goal in the first contribution is to capture the corresponding asymptotic limit. Although it has been shown in [39] that this can, in principle, be achieved within a low-rank approximation, it comes at the cost of a fully implicit scheme. The key difference from previous works lies therefore in that, instead of applying the low rank approximation to the unknown distribution function directly, we start with a macro-micro decomposition of the equation and apply the low rank method only to the micro part of the solution. This approach naturally captures the diffusion limit using a more efficient implicit-explicit (IMEX) discretization strategy. In addition, the micro part of the solution becomes low rank in the diffusion limit, hence the method is particularly efficient in this regime.

We mention that the design of aforementioned numerical schemes that are consistent with certain asymptotic limits falls into the general umbrella of the so-called asymptotic-preserving (AP) schemes [40], which have been developed for various kinds of kinetic and hyperbolic equations in the past decades, see [41]–[43] for an overview. In particular, for the linear transport equation, the use of macro-micro decomposition to achieve the AP property

in the diffusive regime first appeared in [44]. The stability of the scheme was proved in [45] using energy estimates. Comparing to [44], the new difficulty arising in the context of the dynamical low-rank method is to justify the asymptotic limit under the additional projection operator splitting, which we carefully study in this paper. Furthermore, the usual way to generalize the first order (in time) scheme to high order using IMEX Runge-Kutta (RK) schemes, as in [46], [47], cannot be applied to the low rank case again due to the operator splitting. Hence another contribution is to propose an AP dynamical low-rank method that remains second order in both kinetic and diffusive regimes.

The second contribution of this thesis is to develop a highly efficient adaptive low rank method in Boltzmann equation for computations of steady state solutions. The full Boltzmann collision operator is local in spatial variable, which is well-suited for low-rank methods. We employ the fast Fourier spectral method [30] to solve the Boltzmann collision operator, which is the fastest algorithm been reported to date. With the proposed low-rank method, the complexity can be reduced from $\mathcal{O}(N_x^{d_x} M_{FF} N_v^{d_v} \log N_v)$ to $\mathcal{O}(r^3(r+N_x^{d_x}+N_v^{d_v})+ r^2 M_{FF} N_v^{d_v} \log N_v)$, where r is the computational rank; N_x and N_v are number of spatial/velocity points in each dimension; M_{FF} is the number of discretization points over S^{d_v-1} and $M_{FF} \ll N_v^{d_v-1}$. Another novelty lies in the adaptivity with a dynamic thresholding strategy. The fixed-rank dynamic low rank method [48] will pose a potential large computational complexity for large initial rank r. The complexity is even worse for problems with increasing ranks since a large initial rank is needed. Recently, the adaptive dynamic low rank method [49] has been proposed with a fixed thresholding strategy for time dependent Boltzmann equations. We developed a simple but efficient adaptive dynamic low rank algorithm with dynamic thresholding especially for steady state solutions of Boltzmann equation. Because of the special structure of steady state solutions, one only need that the boundary conditions to be accurately enforced. This method can efficiently compute steady state solutions by monitoring residual errors. It will automatically use a low rank solution with the same order accuracy to replace the high-rank one. Compared with the fixed rank methods, this method adaptively use small rank for low accuracy solutions and increase rank when high accuracy solution is needed. The efficiency and accuracy can be verified in various 1D/2D benchmark

tests including normal shock wave, Fourier flow, lid driven cavity flow and thermally driven cavity flow problems.

The last contribution of this thesis is to propose a control variates multilevel Monte Carlo method for BGK equation with randomness as well as the analysis for convergence and consistency. Following the regularity results from [50], we construct a direct analogue to the BGK model with randomness. Due to the non-intrusiveness of MC type methods, approximations of statistical moments can preserve properties from deterministic solvers. We adopted the Runge-Kutta IMEX scheme from [51] and proposed a second order positivity preserving, entropy decaying and asymptotic preserving scheme for the BGK equation. We show the consistency and convergence results with MC, MLMC method coupled with the above numerical scheme involving both discretization and statistical errors. Moreover, the idea of control variate MC method [38] is extended to the MLMC method with some convergence analysis. Lastly, we employ the Chu reduction [52] to increase computational efficiency in spatial discretization.

The rest of this thesis is organized as follows. In Chapter 2, we introduce the dynamic low rank method and present the proposed efficient numerical method in linear transport equation. The dynamic low rank method is extended with adaptivity for full Boltzmann equation in Chapter 3. Chapter 4 discusses the control variates multilevel Monte Carlo method for BGK equation with randomness. The thesis is concluded in chapter 5.

2. ASYMPTOTIC DYNAMICAL LOW-RANK METHODS FOR LINEAR TRANSPORT EQUATION

In this chapter, we focus on dealing with the high dimensional problem for kinetic equations by employing the dynamic low-rank method. At the same time, we emphasize on preserving the limiting behaviors of kinetic equations. We introduce the asymptotic-preserving dynamic low-rank method in linear transport equation. The structure of this chapter is organized as follows. In Section 2.1, we briefly describe the linear transport equation and its macro-micro decomposition. Section 2.2 is the main part of the paper where we introduce the dynamical low-rank method. Both the first and second order schemes along with their AP property are discussed in detail. We further confirmed that in the fluid limit, the solution is low-rank. Section 2.3 provides a simple Fourier analysis for the solution to the linear transport equation. Section 2.4 presents several numerical tests for the two-dimensional equation, where we carefully examine the accuracy, efficiency, rank dependence, and AP property of the proposed method. The paper is concluded in Section 2.5. Most of the results in this chapter are extracted from [53].

2.1 The linear transport equation and its macro-micro decomposition

In many circumstances, rather than binary particle-wise interactions, one is more interested in particles interacting with a background medium. Then the following linear Boltzmann equation is more appropriate

$$\partial_t f + \mathbf{v}\cdot\nabla_{\mathbf{x}} f = \int_{\mathbb{R}^{d_v}} \left(s(\mathbf{v}_*,\mathbf{v})f(\mathbf{v}_*) - s(\mathbf{v},\mathbf{v}_*)f(\mathbf{v})\right)\mathrm{d}\mathbf{v}_*, \quad \mathbf{x}\in\Omega_{\mathbf{x}}\subset\mathbb{R}^{d_x},\ \mathbf{v}\in\mathbb{R}^{d_v}, \tag{2.1.1}$$

where $s(\mathbf{v},\mathbf{v}_*)$ describes the transition rate from $\mathbf{v}$ to $\mathbf{v}_*$ and may take various forms depending on the approximation. The linear transport equation falls into this umbrella and we focus in the following time dependent linear transport equation in diffusive scaling:

$$\partial_t f + \frac{1}{\varepsilon}\mathbf{v}\cdot\nabla_{\mathbf{x}} f = \frac{\sigma^S}{\varepsilon^2}\left(\frac{1}{4\pi}\langle f\rangle_{\mathbf{v}} - f\right) - \sigma^A f + G, \tag{2.1.2}$$

where position $\mathbf{x} = (x, y, z) \in \Omega_{\mathbf{x}} \subset \mathbb{R}^3$, and velocity $\mathbf{v} = (\xi, \eta, \gamma) \in \mathbb{S}^2$ which is confined to the unit sphere[1]. Here, we take $d_x = 3$ and $d_v = 2$. $\langle\ \rangle_{\mathbf{v}}$ denotes the integration over $\mathbb{S}^2$ with respect to $\mathbf{v}$. $\sigma^S(\mathbf{x}) \geq \sigma^S_{\min} > 0$ and $\sigma^A(\mathbf{x}) \geq 0$ are the scattering and absorption coefficients, and $G(\mathbf{x})$ is a given source term. Here, unlike the Knudsen number defined for (1.0.1) and (1.0.6), in this chapter, ε is defined as the re-scaled collision length, which can range between the kinetic regime $\varepsilon \sim O(1)$ and the diffusive regime $\varepsilon \ll 1$.

The density $\rho = \frac{1}{4\pi}\langle f \rangle_{\mathbf{v}}$ is defined as the angular average of f. In the limit $\varepsilon \to 0$, ρ satisfies a diffusion equation which can be seen via the Chapman-Enskog expansion. Indeed, (2.1.2) can be written as

$$f = \rho - \varepsilon \frac{1}{\sigma^S} \mathbf{v} \cdot \nabla_{\mathbf{x}} f - \varepsilon^2 \frac{1}{\sigma^S} \left(\partial_t f + \sigma^A f - G \right) = \rho - \varepsilon \frac{1}{\sigma^S} \mathbf{v} \cdot \nabla_{\mathbf{x}} \rho + O(\varepsilon^2). \tag{2.1.3}$$

On the other hand, taking $\frac{1}{4\pi}\langle\ \rangle_{\mathbf{v}}$ of (2.1.2) yields

$$\partial_t \rho + \frac{1}{4\pi\varepsilon} \nabla_{\mathbf{x}} \cdot \langle \mathbf{v} f \rangle_{\mathbf{v}} = -\sigma^A \rho + G, \tag{2.1.4}$$

which, upon substitution of (2.1.3), becomes

$$\partial_t \rho - \nabla_{\mathbf{x}} \cdot (D \nabla_{\mathbf{x}} \rho) = -\sigma^A \rho + G + O(\varepsilon), \tag{2.1.5}$$

with the diffusion matrix D given by

$$D = \frac{1}{4\pi\sigma^S} \langle \mathbf{v} \otimes \mathbf{v} \rangle_{\mathbf{v}} = \frac{1}{3\sigma^S} I_{3\times 3}. \tag{2.1.6}$$

Therefore, as $\varepsilon \to 0$ the limit of (2.1.2) is the diffusion equation

$$\partial_t \rho - \nabla_{\mathbf{x}} \cdot \left(\frac{1}{3\sigma^S} \nabla_{\mathbf{x}} \rho \right) = -\sigma^A \rho + G. \tag{2.1.7}$$

[1]↑In the context of radiative transfer, $\mathbf{v}$ is usually referred to as angle or direction.

In the macro-micro decomposition [44], we write f as

$$f(t,\mathbf{x},\mathbf{v}) = \rho(t,\mathbf{x}) + \varepsilon g(t,\mathbf{x},\mathbf{v}), \tag{2.1.8}$$

where ρ is the macro part of the solution and g is the micro part. Note that $\langle g\rangle_{\mathbf{v}} = 0$. Substituting (2.1.8) into (2.1.2) and taking $\frac{1}{4\pi}\langle\,\rangle_{\mathbf{v}}$, we obtain

$$\partial_t \rho + \frac{1}{4\pi}\nabla_{\mathbf{x}}\cdot\langle \mathbf{v} g\rangle_{\mathbf{v}} = -\sigma^A \rho + G. \tag{2.1.9}$$

Subtracting (2.1.9) from (2.1.2) yields

$$\partial_t g + \frac{1}{\varepsilon}\left(I - \frac{1}{4\pi}\langle\,\rangle_{\mathbf{v}}\right)(\mathbf{v}\cdot\nabla_{\mathbf{x}} g) + \frac{1}{\varepsilon^2}\mathbf{v}\cdot\nabla_{\mathbf{x}}\rho = -\frac{\sigma^S}{\varepsilon^2} g - \sigma^A g. \tag{2.1.10}$$

The coupled system (2.1.9) and (2.1.10) is the macro-micro decomposition of the linear transport equation (2.1.2). In the limit $\varepsilon \to 0$, we have from (2.1.10):

$$g = -\frac{1}{\sigma^S}\mathbf{v}\cdot\nabla_{\mathbf{x}}\rho, \tag{2.1.11}$$

which, when substituting into (2.1.9), yields the same diffusion equation (2.1.7).

2.2 The dynamical low-rank method for the linear transport equation

We first constrain $g(t,\mathbf{x},\mathbf{v})$ to a low rank manifold $\mathbb{M}$ such that

$$g(t,\mathbf{x},\mathbf{v}) = \sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}}(t,\mathbf{x}) S_{\mathrm{ij}}(t) V_{\mathrm{j}}(t,\mathbf{v}), \tag{2.2.1}$$

where r is called the rank and the basis functions $\{X_{\mathrm{i}}\}_{1\le \mathrm{i}\le r}$ and $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$ are orthonormal:

$$\langle X_{\mathrm{i}}, X_k\rangle_{\mathbf{x}} = \delta_{\mathrm{i}k},\ \ \langle V_{\mathrm{j}}, V_k\rangle_{\mathbf{v}} = \delta_{\mathrm{j}k}, \tag{2.2.2}$$

with $\langle\cdot,\cdot\rangle_{\mathbf{x}}$ and $\langle\cdot,\cdot\rangle_{\mathbf{v}}$ being the inner products on $L^2(\Omega_{\mathbf{x}})$ and $L^2(\mathbb{S}^2)$, respectively.

With this low rank approximation, (2.1.9) becomes

$$\partial_t \rho = -\frac{1}{4\pi} \sum_{\mathrm{i,j}=1}^{r} \nabla_{\mathbf{x}} \cdot (X_{\mathrm{i}} S_{\mathrm{ij}} \langle \mathbf{v} V_{\mathrm{j}} \rangle_{\mathbf{v}}) - \sigma^A \rho + G. \tag{2.2.3}$$

For (2.1.10), we write

$$\partial_t g = -\frac{1}{\varepsilon} \left(I - \frac{1}{4\pi} \langle \, \rangle_{\mathbf{v}} \right) (\mathbf{v} \cdot \nabla_{\mathbf{x}} g) - \frac{1}{\varepsilon^2} \mathbf{v} \cdot \nabla_{\mathbf{x}} \rho - \frac{\sigma^S}{\varepsilon^2} g - \sigma^A g := \mathrm{RHS}. \tag{2.2.4}$$

Equation (2.2.4), however, does not uniquely specify the dynamics of the low-rank factors X_{i}, S_{ij}, and V_{j}. We therefore impose the following gauge conditions [54]:

$$\langle \partial_t X_{\mathrm{i}}, X_k \rangle_{\mathbf{x}} = 0, \ \langle \partial_t V_{\mathrm{j}}, V_k \rangle_{\mathbf{v}} = 0. \tag{2.2.5}$$

Let us emphasize that the resulting dynamics of g is independent of the specific gauge conditions chosen. However, using (2.2.5) is convenient as it allows us to easily obtain evolution equations in terms of the low-rank factors. To that end, we now project the right hand side of (2.2.4) onto the tangent space of $\mathbb{M}$:

$$\partial_t g = P_g(\mathrm{RHS}), \tag{2.2.6}$$

where the orthogonal projector P_g can be written as

$$P_g(\mathrm{RHS}) = \sum_{\mathrm{j}=1}^{r} \langle V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{v}} V_{\mathrm{j}} - \sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}} \langle X_{\mathrm{i}} V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{x},\mathbf{v}} V_{\mathrm{j}} + \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}} \langle X_{\mathrm{i}}, \mathrm{RHS} \rangle_{\mathbf{x}}. \tag{2.2.7}$$

Using (2.2.7) and the gauge conditions we can in principle derive evolution equations for X_{i}, S_{ij}, and V_{j}. However, this process requires inverting the matrix $S = (S_{\mathrm{ij}})$. Since an accurate approximation mandates that S has small singular values, the resulting problem is severely ill-conditioned. Thus, we will use the projector splitting scheme introduced in [48]. For a corresponding mathematical analysis see [55]. This scheme has been extensively used in the literature, see e.g. [25], [29], [56], and extensions to various tensor formats have also

been proposed [56], [57]. The main idea is to split equation (2.2.6) into the following three subflows

$$\partial_t g = \sum_{\mathrm{j}=1}^{r} \langle V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{v}} V_{\mathrm{j}},$$
$$\partial_t g = -\sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}} \langle X_{\mathrm{i}} V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{x},\mathbf{v}} V_{\mathrm{j}},$$
$$\partial_t g = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}} \langle X_{\mathrm{i}}, \mathrm{RHS} \rangle_{\mathbf{x}}.$$

This is particularly convenient as for the first subflow V_{j} is constant (in time), for the third subflow X_{i} is constant, and for the second subflow both X_{i} and V_{j} are constant. Thus, we can write

$$\partial_t K_{\mathrm{j}} = \langle V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{v}}, \tag{2.2.8}$$
$$\partial_t S_{\mathrm{ij}} = -\langle X_{\mathrm{i}} V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{x},\mathbf{v}}, \tag{2.2.9}$$
$$\partial_t L_{\mathrm{i}} = \langle X_{\mathrm{i}}, \mathrm{RHS} \rangle_{\mathbf{x}}, \tag{2.2.10}$$

where

$$K_{\mathrm{j}}(t, \mathbf{x}) = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}}(t, \mathbf{x}) S_{\mathrm{ij}}(t), \qquad L_{\mathrm{i}}(t, \mathbf{v}) = \sum_{\mathrm{j}=1}^{r} S_{\mathrm{ij}}(t) V_{\mathrm{j}}(t, \mathbf{v}). \tag{2.2.11}$$

After solving each subflow we use a QR decomposition to obtain X_{i} and S_{ij} from K_{j} and S_{ij} and V_{j} from L_{i}, respectively.

2.2.1 A first order in time scheme

Our goal is to solve the coupled system (2.2.3) and (2.2.6) using the projector splitting integrator outlined in the previous section. We now proceed by deriving the evolution equations corresponding to the subflows given by equations (2.2.8)-(2.2.10).

- K**-step**: Solve $\partial_t K_{\mathrm{j}} = \langle V_{\mathrm{j}}, \mathrm{RHS}\rangle_{\mathbf{v}}$ with $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$ unchanged.

$$\begin{aligned}\partial_t K_{\mathrm{j}} =& \langle V_{\mathrm{j}}, \mathrm{RHS}\rangle_{\mathbf{v}} \\ =& -\frac{1}{\varepsilon}\sum_{l=1}^{r}\left(\langle \mathbf{v}V_{\mathrm{j}}V_l\rangle_{\mathbf{v}} - \frac{1}{4\pi}\langle V_{\mathrm{j}}\rangle_{\mathbf{v}}\langle \mathbf{v}V_l\rangle_{\mathbf{v}}\right)\cdot\nabla_{\mathbf{x}}K_l \\ & -\frac{1}{\varepsilon^2}\langle \mathbf{v}V_{\mathrm{j}}\rangle_{\mathbf{v}}\cdot\nabla_{\mathbf{x}}\rho - \left(\frac{\sigma^S}{\varepsilon^2}+\sigma^A\right)K_{\mathrm{j}}.\end{aligned} \tag{2.2.12}$$

- L**-step**: Solve $\partial_t L_{\mathrm{i}} = \langle X_{\mathrm{i}}, \mathrm{RHS}\rangle_{\mathbf{x}}$ with $\{X_{\mathrm{i}}\}_{1\le \mathrm{i}\le r}$ unchanged.

$$\begin{aligned}\partial_t L_{\mathrm{i}} =& \langle X_{\mathrm{i}}, \mathrm{RHS}\rangle_{\mathbf{x}} \\ =& -\frac{1}{\varepsilon}\sum_{k=1}^{r}\left(\mathbf{v}L_k - \frac{1}{4\pi}\langle \mathbf{v}L_k\rangle_{\mathbf{v}}\right)\cdot\langle X_{\mathrm{i}}\nabla_{\mathbf{x}}X_k\rangle_{\mathbf{x}} \\ & -\frac{1}{\varepsilon^2}\mathbf{v}\cdot\langle X_{\mathrm{i}}\nabla_{\mathbf{x}}\rho\rangle_{\mathbf{x}} - \sum_{k=1}^{r}\left\langle X_{\mathrm{i}}\left(\frac{\sigma^S}{\varepsilon^2}+\sigma^A\right)X_k\right\rangle_{\mathbf{x}}L_k.\end{aligned} \tag{2.2.13}$$

- S**-step**: Solve $\partial_t S_{\mathrm{ij}} = -\langle X_{\mathrm{i}}V_{\mathrm{j}}, \mathrm{RHS}\rangle_{\mathbf{x},\mathbf{v}}$ with both $\{X_{\mathrm{i}}\}_{1\le \mathrm{i}\le r}$ and $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$ unchanged.

$$\begin{aligned}\partial_t S_{\mathrm{ij}} =& -\langle X_{\mathrm{i}}V_{\mathrm{j}}, \mathrm{RHS}\rangle_{\mathbf{x},\mathbf{v}} \\ =& \frac{1}{\varepsilon}\sum_{k,l=1}^{r}\left(\langle \mathbf{v}V_{\mathrm{j}}V_l\rangle_{\mathbf{v}} - \frac{1}{4\pi}\langle V_{\mathrm{j}}\rangle_{\mathbf{v}}\langle \mathbf{v}V_l\rangle_{\mathbf{v}}\right)\cdot\langle X_{\mathrm{i}}\nabla_{\mathbf{x}}X_k\rangle_{\mathbf{x}}S_{kl} \\ & +\frac{1}{\varepsilon^2}\langle \mathbf{v}V_{\mathrm{j}}\rangle_{\mathbf{v}}\cdot\langle X_{\mathrm{i}}\nabla_{\mathbf{x}}\rho\rangle_{\mathbf{x}} + \sum_{k=1}^{r}\left\langle X_{\mathrm{i}}\left(\frac{\sigma^S}{\varepsilon^2}+\sigma^A\right)X_k\right\rangle_{\mathbf{x}}S_{k\mathrm{j}}.\end{aligned} \tag{2.2.14}$$

Therefore, for the overall system, we can construct a simple first order in time scheme. Suppose at time step t^n, we have $(X_{\mathrm{i}}^n, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^n, \rho^n)$. To obtain the solution $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{n+1}, \rho^{n+1})$ at t^{n+1} we proceed as follows:

1. K**-step**: Solve (2.2.12) for a full time step Δt, update from $(X_{\mathrm{i}}^n, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^n)$ to $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^{(1)})$ using ρ^n. Specifically, given $K_{\mathrm{j}}^n = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}}^n S_{\mathrm{ij}}^n$, we discretize (2.2.12) using a first order IMEX scheme (i.e., forward-backward Euler scheme) as

$$\begin{aligned}\frac{K_{\mathrm{j}}^{n+1}-K_{\mathrm{j}}^n}{\Delta t} =& -\frac{1}{\varepsilon}\sum_{l=1}^{r}\left(\langle \mathbf{v}V_{\mathrm{j}}^nV_l^n\rangle_{\mathbf{v}} - \frac{1}{4\pi}\langle V_{\mathrm{j}}^n\rangle_{\mathbf{v}}\langle \mathbf{v}V_l^n\rangle_{\mathbf{v}}\right)\cdot\nabla_{\mathbf{x}}K_l^n \\ & -\frac{1}{\varepsilon^2}\left(\langle \mathbf{v}V_{\mathrm{j}}^n\rangle_{\mathbf{v}}\cdot\nabla_{\mathbf{x}}\rho^n + \sigma^S K_{\mathrm{j}}^{n+1}\right) - \sigma^A K_{\mathrm{j}}^n,\end{aligned} \tag{2.2.15}$$

where the term $\sigma^S K_{\mathrm{j}}$ is treated implicitly to overcome the stiffness induced by a small ε. We then perform the QR decomposition of K_{j}^{n+1} to obtain the updated basis functions X_{i}^{n+1} and the matrix $S_{\mathrm{ij}}^{(1)}$:

$$K_{\mathrm{j}}^{n+1} = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}}^{n+1} S_{\mathrm{ij}}^{(1)}. \tag{2.2.16}$$

2. L**-step**: Solve (2.2.13) for a full time step Δt, update from $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n}, S_{\mathrm{ij}}^{(1)})$ to $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{(2)})$ using ρ^n. Specifically, given $L_{\mathrm{i}}^{n} = \sum_{\mathrm{j}=1}^{r} S_{\mathrm{ij}}^{(1)} V_{\mathrm{j}}^{n}$, we discretize (2.2.13) (similar to (2.2.12)) as follows

$$\begin{aligned}\frac{L_{\mathrm{i}}^{n+1} - L_{\mathrm{i}}^{n}}{\Delta t} =& -\frac{1}{\varepsilon}\sum_{k=1}^{r}\left(\mathbf{v} L_k^n - \frac{1}{4\pi}\langle \mathbf{v} L_k^n\rangle_{\mathbf{v}}\right)\cdot\langle X_{\mathrm{i}}^{n+1}\nabla_{\mathbf{x}} X_k^{n+1}\rangle_{\mathbf{x}} \\ & -\frac{1}{\varepsilon^2}\left(\mathbf{v}\cdot\langle X_{\mathrm{i}}^{n+1}\nabla_{\mathbf{x}}\rho^n\rangle_{\mathbf{x}} + \sum_{k=1}^{r}\left\langle X_{\mathrm{i}}^{n+1}\sigma^S X_k^{n+1}\right\rangle_{\mathbf{x}} L_k^{n+1}\right) - \sum_{k=1}^{r}\left\langle X_{\mathrm{i}}^{n+1}\sigma^A X_k^{n+1}\right\rangle_{\mathbf{x}} L_k^n.\end{aligned} \tag{2.2.17}$$

We then perform the QR decomposition of L_{i}^{n+1} to obtain the updated basis V_{j}^{n+1} and matrix $S_{\mathrm{ij}}^{(2)}$:

$$L_{\mathrm{i}}^{n+1} = \sum_{\mathrm{j}=1}^{r} S_{\mathrm{ij}}^{(2)} V_{\mathrm{j}}^{n+1}. \tag{2.2.18}$$

3. S**-step**: Solve (2.2.14) for a full time step Δt, update from $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{(2)})$ to $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{n+1})$ using ρ^n. Specifically, given $S_{\mathrm{ij}}^{(2)}$, we discretize (2.2.14) (similar to (2.2.12)) as follows

$$\begin{aligned}\frac{S_{\mathrm{ij}}^{n+1} - S_{\mathrm{ij}}^{(2)}}{\Delta t} =& \frac{1}{\varepsilon}\sum_{k,l=1}^{r}\left(\langle \mathbf{v} V_{\mathrm{j}}^{n+1} V_l^{n+1}\rangle_{\mathbf{v}} - \frac{1}{4\pi}\langle V_{\mathrm{j}}^{n+1}\rangle_{\mathbf{v}}\langle \mathbf{v} V_l^{n+1}\rangle_{\mathbf{v}}\right)\cdot\langle X_{\mathrm{i}}^{n+1}\nabla_{\mathbf{x}} X_k^{n+1}\rangle_{\mathbf{x}} S_{kl}^{(2)} \\ & + \frac{1}{\varepsilon^2}\left(\langle \mathbf{v} V_{\mathrm{j}}^{n+1}\rangle_{\mathbf{v}}\cdot\langle X_{\mathrm{i}}^{n+1}\nabla_{\mathbf{x}}\rho^n\rangle_{\mathbf{x}} + \sum_{k=1}^{r}\langle X_{\mathrm{i}}^{n+1}\sigma^S X_k^{n+1}\rangle_{\mathbf{x}} S_{\mathrm{kj}}^{n+1}\right) \\ & + \sum_{k=1}^{r}\langle X_{\mathrm{i}}^{n+1}\sigma^A X_k^{n+1}\rangle_{\mathbf{x}} S_{\mathrm{kj}}^{(2)}.\end{aligned} \tag{2.2.19}$$

4. ρ**-step**: Solve (2.2.3) for a full time step Δt, update from ρ^n to ρ^{n+1} using $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{n+1})$. Specifically, given ρ^n, we discretize (2.2.3) as

$$\frac{\rho^{n+1}-\rho^n}{\Delta t} = -\frac{1}{4\pi}\sum_{\mathrm{i,j}=1}^{r} \nabla_{\mathbf{x}} \cdot \left(X_{\mathrm{i}}^{n+1} S_{\mathrm{ij}}^{n+1} \langle \mathbf{v} V_{\mathrm{j}}^{n+1} \rangle_{\mathbf{v}}\right) - \sigma^A \rho^n + G. \tag{2.2.20}$$

For clarity, we will refer to the above scheme as the K-L-S-ρ **scheme** in the following.

2.2.2 AP property of the first order scheme

In this subsection, we analyze the AP property of the first order scheme introduced in the previous section. Our conclusion is summarized in the following proposition.

Proposition 2.2.1. *In the limit $\varepsilon \to 0$, the first order IMEX K-L-S-ρ scheme (i.e., (2.2.15), (2.2.17), (2.2.19), and (2.2.20)) becomes the forward Euler scheme for the limiting diffusion equation (2.1.7), provided that for the initial value we have $\xi, \eta, \gamma \in span(\{V_{\mathrm{j}}^0\}_{\mathrm{j}=1}^r)$.*

Remark 2.2.1. *If, for a given initial value $(X_{\mathrm{i}}^0, S_{\mathrm{ij}}^0, V_{\mathrm{j}}^0)$, one of the conditions $\xi, \eta, \gamma \in span(\{V_{\mathrm{j}}^0\}_{\mathrm{j}=1}^r)$ is not satisfied, we can simply add them to the approximation space. For example, if $\xi \notin span(\{V_{\mathrm{j}}^0\}_{\mathrm{j}=1}^r)$, we consider*

$$\tilde{X}^0 = [X_1^0, \ldots, X_r^0, h], \qquad \tilde{S}^0 = \begin{bmatrix} S^0 & 0 \\ 0 & 0 \end{bmatrix}, \qquad \tilde{V}^0 = [V_1^0, \ldots, V_r^0, \xi],$$

where h is an arbitrary function. We then orthogonalize $\tilde{X}^0$ and $\tilde{V}^0$ (e.g. using the Gram-Schmidt process) and use the result as the initial value in our algorithm. This increases the rank to at most $r+3$.

Proof. In the **K-step**, let $\varepsilon \to 0$, we have from (2.2.15):

$$K_{\mathrm{j}}^{n+1} = -\langle \mathbf{v} V_{\mathrm{j}}^n \rangle_{\mathbf{v}} \cdot \frac{\nabla_{\mathbf{x}} \rho^n}{\sigma^S}. \tag{2.2.21}$$

Without loss of generality, we assume that the three components of $\frac{\nabla_{\mathbf{x}}\rho^n}{\sigma^S}$: $\frac{\partial_x\rho^n}{\sigma^S}$, $\frac{\partial_y\rho^n}{\sigma^S}$ and $\frac{\partial_z\rho^n}{\sigma^S}$ are linearly independent[2]. Then after the QR decomposition of K_{j}^{n+1}, the span of the new basis $\{X_{\mathrm{i}}^{n+1}\}_{1\le \mathrm{i}\le 3}$ would be the same as span$\{\frac{\partial_x\rho^n}{\sigma^S}, \frac{\partial_y\rho^n}{\sigma^S}, \frac{\partial_z\rho^n}{\sigma^S}\}$. In other words, we can write

$$X^{n+1} := \begin{bmatrix} X_1^{n+1} & X_2^{n+1} & X_3^{n+1} & X_4^{n+1} & \cdots & X_r^{n+1} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\partial_x\rho^n}{\sigma^S} & \frac{\partial_y\rho^n}{\sigma^S} & \frac{\partial_z\rho^n}{\sigma^S} & X_4^{n+1} & \cdots & X_r^{n+1} \end{bmatrix}}_{:=X_0} D_1, \tag{2.2.22}$$

where D_1 is an invertible $r \times r$ matrix.

In the **L-step**, let $\varepsilon \to 0$, we have from (2.2.17):

$$\sum_{k=1}^{r} \langle X_{\mathrm{i}}^{n+1} \sigma^S X_k^{n+1} \rangle_{\mathbf{x}} L_k^{n+1} = -\mathbf{v} \cdot \langle X_{\mathrm{i}}^{n+1} \nabla_{\mathbf{x}} \rho^n \rangle_{\mathbf{x}}. \tag{2.2.23}$$

Since the matrix $A := (\langle X_{\mathrm{i}}^{n+1} \sigma^S X_k^{n+1} \rangle_{\mathbf{x}})_{1\le \mathrm{i}\le r, 1\le k\le r}$ is symmetric positive definite (since $\sigma^S > 0$), hence invertible (whose inverse, say, is matrix $B = (b_{k\mathrm{i}})_{1\le k\le r, 1\le \mathrm{i}\le r}$), we have

$$L_k^{n+1} = -\mathbf{v} \cdot \left(\sum_{\mathrm{i}=1}^{r} b_{k\mathrm{i}} \langle X_{\mathrm{i}}^{n+1} \nabla_{\mathbf{x}} \rho^n \rangle_{\mathbf{x}} \right). \tag{2.2.24}$$

After the QR decomposition of L_k^{n+1}, we can write (by a similar argument as above)

$$V^{n+1} := \begin{bmatrix} V_1^{n+1} & V_2^{n+1} & V_3^{n+1} & V_4^{n+1} & \cdots & V_r^{n+1} \end{bmatrix} = \underbrace{\begin{bmatrix} \xi & \eta & \gamma & V_4^{n+1} & \cdots & V_r^{n+1} \end{bmatrix}}_{:=V_0} D_2, \tag{2.2.25}$$

where D_2 is an invertible $r \times r$ matrix.

In the **S-step**, let $\varepsilon \to 0$, we have from (2.2.19):

$$\begin{aligned} \sum_{k=1}^{r} \langle X_{\mathrm{i}}^{n+1} \sigma^S X_k^{n+1} \rangle_{\mathbf{x}} S_{\mathrm{kj}}^{n+1} &= -\langle \mathbf{v} V_{\mathrm{j}}^{n+1} \rangle_{\mathbf{v}} \cdot \langle X_{\mathrm{i}}^{n+1} \nabla_{\mathbf{x}} \rho^n \rangle_{\mathbf{x}} \\ &= -\langle X_{\mathrm{i}}^{n+1} \mathbf{v} \cdot \nabla_{\mathbf{x}} \rho^n V_{\mathrm{j}}^{n+1} \rangle_{\mathbf{x},\mathbf{v}}. \end{aligned} \tag{2.2.26}$$

[2]↑If they are linearly dependent, say, span$\{\frac{\partial_x\rho^n}{\sigma^S}, \frac{\partial_y\rho^n}{\sigma^S}, \frac{\partial_z\rho^n}{\sigma^S}\}$ = span$\{\frac{\partial_x\rho^n}{\sigma^S}\}$, then one just needs to replace the second and third components of X_0 by X_2^{n+1} and X_3^{n+1} and the same analysis carries over.

We may write (2.2.26) as $AS^{n+1} = C$. Since the matrix A is invertible, we know that the matrix S^{n+1} is unique. We next claim that the S^{n+1} defined as

$$S^{n+1} := D_1^{-1} \begin{bmatrix} -I_{3\times 3} & 0 \\ 0 & 0 \end{bmatrix} D_2^{-T}, \tag{2.2.27}$$

satisfies (2.2.26), where the middle matrix is of size $r \times r$, with $-I_{3\times 3}$ in the first 3×3 block and zero elsewhere. Indeed, using (2.2.22) and (2.2.25) we have

$$g^{n+1} = \sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}}^{n+1} S_{\mathrm{ij}}^{n+1} V_{\mathrm{j}}^{n+1} = X^{n+1} S^{n+1} (V^{n+1})^T = X_0 \begin{bmatrix} -I_{3\times 3} & 0 \\ 0 & 0 \end{bmatrix} V_0^T = -\mathbf{v} \cdot \frac{\nabla_{\mathbf{x}} \rho^n}{\sigma^S}. \tag{2.2.28}$$

Therefore,

$$(X^{n+1})^T \sigma^S X^{n+1} S^{n+1} (V^{n+1})^T V^{n+1} = -(X^{n+1})^T (\mathbf{v} \cdot \nabla_{\mathbf{x}} \rho^n) V^{n+1}, \tag{2.2.29}$$

which, upon taking $\langle\ \rangle_{\mathbf{x},\mathbf{v}}$, yields

$$\langle (X^{n+1})^T \sigma^S X^{n+1} \rangle_{\mathbf{x}} S^{n+1} = -\langle (X^{n+1})^T (\mathbf{v} \cdot \nabla_{\mathbf{x}} \rho^n) V^{n+1} \rangle_{\mathbf{x},\mathbf{v}}, \tag{2.2.30}$$

which is precisely (2.2.26).

On the other hand, substituting (2.2.28) into (2.2.20) gives

$$\frac{\rho^{n+1} - \rho^n}{\Delta t} = \nabla_{\mathbf{x}} \cdot \left(\frac{1}{3\sigma^S} \nabla_{\mathbf{x}} \rho^n \right) - \sigma^A \rho^n + G, \tag{2.2.31}$$

which is the forward Euler scheme for the limiting diffusion equation (2.1.7). □

2.2.3 Some other first order schemes and their AP property

From the operator splitting point of view, the previously introduced K-L-S-ρ scheme is certainly not the only first order scheme. In fact, one can switch the order of K, L, and S steps arbitrarily and still obtains a first order scheme. For example, the L-K-S-ρ scheme is

also first order and preserves the same asymptotic limit as the K-L-S-ρ scheme (since the proof of Proposition 2.2.1 still holds if one switches the K and L steps). Nonetheless, for some other first order schemes, such as L-S-K-ρ, S-L-K-ρ, K-S-L-ρ, and S-K-L-ρ schemes, their AP property needs to be examined individually. Fortunately, as we will show in the following, by slightly different arguments these schemes all have the same asymptotic limit as the K-L-S-ρ scheme.

- **L-S-K-ρ scheme** and **S-L-K-ρ scheme**.

  After the first two substeps (L-S or S-L), the span of the updated basis $\{V_{\mathrm{j}}^{n+1}\}_{1\le \mathrm{j}\le r}$ will contain $\mathbf{v}$. After the substep K, one has $K_{\mathrm{j}}^{n+1} = -\langle \mathbf{v} V_{\mathrm{j}}^{n+1}\rangle_{\mathbf{v}} \cdot \frac{\nabla_{\mathbf{x}}\rho^n}{\sigma^S}$. Hence,

$$g^{n+1} = \sum_{\mathrm{j}=1}^{r} K_{\mathrm{j}}^{n+1} V_{\mathrm{j}}^{n+1} = -\sum_{\mathrm{j}=1}^{r} \langle \mathbf{v} V_{\mathrm{j}}^{n+1}\rangle_{\mathbf{v}} V_{\mathrm{j}}^{n+1} \cdot \frac{\nabla_{\mathbf{x}}\rho^n}{\sigma^S} = -\mathbf{v}\cdot\frac{\nabla_{\mathbf{x}}\rho^n}{\sigma^S}. \tag{2.2.32}$$

  Substituting g^{n+1} into the last ρ step recovers (2.2.31).

- **K-S-L-ρ scheme** and **S-K-L-ρ scheme**.

  After the first two substeps (K-S or S-K), one has

$$\begin{bmatrix} X_1^{n+1} & \cdots & X_r^{n+1}\end{bmatrix} = \begin{bmatrix} \frac{\partial_x\rho^n}{\sigma^S} & \frac{\partial_y\rho^n}{\sigma^S} & \frac{\partial_z\rho^n}{\sigma^S} & X_4^{n+1} & \cdots & X_r^{n+1}\end{bmatrix} D_1, \tag{2.2.33}$$

  where D_1 is an invertible $r\times r$ matrix. After the substep L, one has

$$\sum_{k=1}^{r} \langle X_{\mathrm{i}}^{n+1}\sigma^S X_k^{n+1}\rangle_{\mathbf{x}} L_k^{n+1} = -\mathbf{v}\cdot\langle X_{\mathrm{i}}^{n+1}\nabla_{\mathbf{x}}\rho^n\rangle_{\mathbf{x}}, \tag{2.2.34}$$

  and $\{L_k^{n+1}\}_{1\le k\le r}$ is uniquely determined since the matrix $\langle X_{\mathrm{i}}^{n+1}\sigma^S X_k^{n+1}\rangle_{\mathbf{x}}$ is invertible. We now claim that $\{L_k^{n+1}\}_{1\le k\le r}$ defined as follows

$$\begin{bmatrix} L_1^{n+1} & \cdots & L_r^{n+1}\end{bmatrix} := -\begin{bmatrix} \xi & \eta & \gamma & 0 & \cdots & 0\end{bmatrix} D_1^{-T}. \tag{2.2.35}$$

satisfies (2.2.34). Indeed, for such L_k, one has

$$g^{n+1} = \sum_{k=1}^{r} X_k^{n+1} L_k^{n+1} = -\mathbf{v} \cdot \frac{\nabla_{\mathbf{x}} \rho^n}{\sigma^S} \implies \sum_{k=1}^{r} \sigma^S X_k^{n+1} L_k^{n+1} = -\mathbf{v} \cdot \nabla_{\mathbf{x}} \rho^n, \tag{2.2.36}$$

which, upon projection onto the space spanned by $\{X_{\mathrm{i}}^{n+1}\}_{1\le \mathrm{i} \le r}$, yields (2.2.34). On the other hand, substituting g^{n+1} into the last ρ step recovers (2.2.31).

Remark 2.2.2. *The discussion in this subsection implies that one has the flexibility to choose the updating order of K, L and S, while still maintaining the AP property. This flexibility is crucial in designing second order schemes, where one needs to properly compose these steps to achieve high order as well as preserve the asymptotic limit.*

2.2.4 A second order in time scheme and its AP property

We now extend the first order scheme to second order. Due to the operator splitting necessary in the low rank method, a straightforward application of the IMEX-RK scheme as used in [46], [47] does not work (there a coupled system for ρ and g is solved simultaneously; in the present work ρ has to be "frozen" while updating g). In the following, we propose a scheme that maintains second order in both kinetic and diffusive regimes. It is a proper combination of the almost symmetric Strang splitting [58], [59] and the IMEX-RK scheme.

Suppose at time step t^n, we have $(X_{\mathrm{i}}^n, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^n, \rho^n)$. To obtain the solution $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{n+1}, \rho^{n+1})$ at t^{n+1}, we proceed as follows:

1. **ρ-step**: Solve (2.2.3) for a half time step $\Delta t/2$, update from ρ^n to $\rho^{n+\frac{1}{2}}$ using $(X_{\mathrm{i}}^n, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^n)$.

2. **K-step**: Solve (2.2.12) for a half time step $\Delta t/2$, update from $(X_{\mathrm{i}}^n, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^n)$ to $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^{(1)})$ using $\rho^{n+\frac{1}{2}}$.

3. **L-step**: Solve (2.2.13) for a half time step $\Delta t/2$, update from $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^n, S_{\mathrm{ij}}^{(1)})$ to $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+\frac{1}{2}}, S_{\mathrm{ij}}^{(2)})$ using $\rho^{n+\frac{1}{2}}$.

4. **S-step**: Solve (2.2.14) for a half time step $\Delta t/2$, update from $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+\frac{1}{2}}, S_{\mathrm{ij}}^{(2)})$ to $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+\frac{1}{2}}, S_{\mathrm{ij}}^{n+\frac{1}{2}})$ using $\rho^{n+\frac{1}{2}}$.

5. S**-step**: Solve (2.2.14) for a half time step $\Delta t/2$, update from $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+\frac{1}{2}}, S_{\mathrm{ij}}^{n+\frac{1}{2}})$ to $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+\frac{1}{2}}, S_{\mathrm{ij}}^{(3)})$ using $\rho^{n+\frac{1}{2}}$.

6. L**-step**: Solve (2.2.13) for a half time step $\Delta t/2$, update from $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+\frac{1}{2}}, S_{\mathrm{ij}}^{(3)})$ to $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{(4)})$ using $\rho^{n+\frac{1}{2}}$.

7. K**-step**: Solve (2.2.12) for a half time step $\Delta t/2$, update from $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{(4)})$ to $(X_{\mathrm{i}}^{n+1}, V_{\mathrm{j}}^{n+1}, S_{\mathrm{ij}}^{n+1})$ using $\rho^{n+\frac{1}{2}}$.

8. ρ**-step**: Solve (2.2.3) for a full time step Δt, update from ρ^n to ρ^{n+1} using $(X_{\mathrm{i}}^{n+\frac{1}{2}}, V_{\mathrm{j}}^{n+\frac{1}{2}}, S_{\mathrm{ij}}^{n+\frac{1}{2}})$.

More specifically, in step 1, we use the forward Euler scheme to discretize (2.2.3):

$$\frac{\rho^{n+\frac{1}{2}}-\rho^n}{\Delta t/2} = -\frac{1}{4\pi}\sum_{\mathrm{i,j}=1}^{r}\nabla_{\mathbf{x}}\cdot\left(X_{\mathrm{i}}^n S_{\mathrm{ij}}^n \langle \mathbf{v} V_{\mathrm{j}}^n\rangle_{\mathbf{v}}\right) - \sigma^A \rho^n + G. \tag{2.2.37}$$

In steps 2-7, we use a second order IMEX-RK scheme to discretize the system for K, L or S. Let us take step 2 for example,

$$\begin{aligned}
K_{\mathrm{j}}^{(p)} =& K_{\mathrm{j}}^n - \frac{\Delta t}{2}\sum_{q=1}^{p-1}\tilde{a}_{pq}\left(\frac{1}{\varepsilon}\sum_{l=1}^{r}\left(\langle \mathbf{v} V_{\mathrm{j}}^n V_l^n\rangle_{\mathbf{v}} - \frac{1}{4\pi}\langle V_{\mathrm{j}}^n\rangle_{\mathbf{v}}\langle \mathbf{v} V_l^n\rangle_{\mathbf{v}}\right)\cdot\nabla_{\mathbf{x}}K_l^{(q)} + \frac{1}{\varepsilon^2}\langle \mathbf{v} V_{\mathrm{j}}^n\rangle_{\mathbf{v}}\cdot\nabla_{\mathbf{x}}\rho^{n+\frac{1}{2}} + \sigma^A K_{\mathrm{j}}^{(q)}\right) \\
& - \frac{\Delta t}{2}\sum_{q=1}^{p} a_{pq}\left(\frac{\sigma^S}{\varepsilon^2}K_{\mathrm{j}}^{(q)}\right),\ p=1,\dots,s, \\
K_{\mathrm{j}}^{n+1} =& K_{\mathrm{j}}^n - \frac{\Delta t}{2}\sum_{p=1}^{s}\tilde{w}_p\left(\frac{1}{\varepsilon}\sum_{l=1}^{r}\left(\langle \mathbf{v} V_{\mathrm{j}}^n V_l^n\rangle_{\mathbf{v}} - \frac{1}{4\pi}\langle V_{\mathrm{j}}^n\rangle_{\mathbf{v}}\langle \mathbf{v} V_l^n\rangle_{\mathbf{v}}\right)\cdot\nabla_{\mathbf{x}}K_l^{(p)} + \frac{1}{\varepsilon^2}\langle \mathbf{v} V_{\mathrm{j}}^n\rangle_{\mathbf{v}}\cdot\nabla_{\mathbf{x}}\rho^{n+\frac{1}{2}} + \sigma^A K_{\mathrm{j}}^{(p)}\right) \\
& - \frac{\Delta t}{2}\sum_{p=1}^{s} w_p\left(\frac{\sigma^S}{\varepsilon^2}K_{\mathrm{j}}^{(p)}\right),
\end{aligned} \tag{2.2.38}$$

where $\tilde{A}=(\tilde{a}_{pq})$, $\tilde{a}_{pq}=0$ for $q\geq p$ and $A=(a_{pq})$, $a_{pq}=0$ for $q>p$ are $s\times s$ matrices. Along with $\tilde{\mathbf{w}}=(\tilde{w}_1,\dots,\tilde{w}_s)^T$, $\mathbf{w}=(w_1,\dots,w_s)^T$, they can be represented by a double Butcher tableau:

$$\begin{array}{c|c} \tilde{\mathbf{c}} & \tilde{A} \\ \hline & \tilde{\mathbf{w}}^T \end{array} \qquad \begin{array}{c|c} \mathbf{c} & A \\ \hline & \mathbf{w}^T \end{array} \tag{2.2.39}$$

where $\tilde{\mathbf{c}} = (\tilde{c}_1, \ldots, \tilde{c}_s)^T$, $\mathbf{c} = (c_1, \ldots, c_s)^T$ are defined as

$$\tilde{c}_p = \sum_{q=1}^{p-1} \tilde{a}_{pq}, \quad c_p = \sum_{q=1}^{p} a_{pq}. \tag{2.2.40}$$

Here we employ the ARS(2,2,2) scheme whose double tableau is given by

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \gamma & \gamma & 0 & 0 \\ 1 & \delta & 1-\delta & 0 \\ \hline & \delta & 1-\delta & 0 \end{array} \qquad \begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \gamma & 0 & \gamma & 0 \\ 1 & 0 & 1-\gamma & \gamma \\ \hline & 0 & 1-\gamma & \gamma \end{array} \qquad \gamma = 1 - \frac{\sqrt{2}}{2}, \quad \delta = 1 - \frac{1}{2\gamma}. \tag{2.2.41}$$

Finally, in step 8, we use the midpoint scheme to discretize (2.2.3):

$$\frac{\rho^{n+1} - \rho^n}{\Delta t} = -\frac{1}{4\pi} \sum_{\mathrm{i,j}=1}^{r} \nabla_{\mathbf{x}} \cdot \left(X_{\mathrm{i}}^{n+\frac{1}{2}} S_{\mathrm{ij}}^{n+\frac{1}{2}} \langle \mathbf{v} V_{\mathrm{j}}^{n+\frac{1}{2}} \rangle_{\mathbf{v}} \right) - \sigma^A \rho^{n+\frac{1}{2}} + G. \tag{2.2.42}$$

Let us analyze the AP property of the above second order scheme. First, steps 2-4 (K-L-S) are (almost) the same as steps 1-3 in the first order K-L-S-ρ scheme (as discussed in Section 2.2.2), hence as $\varepsilon \to 0$, one has

$$g^{n+\frac{1}{2}} = \sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}}^{n+\frac{1}{2}} S_{\mathrm{ij}}^{n+\frac{1}{2}} V_{\mathrm{j}}^{n+\frac{1}{2}} = -\mathbf{v} \cdot \frac{\nabla_{\mathbf{x}} \rho^{n+\frac{1}{2}}}{\sigma^S}. \tag{2.2.43}$$

Furthermore, steps 5-6 (S-L-K) are (almost) the same as steps 1-3 in the first order S-L-K-ρ scheme (as discussed in Section 2.2.3), hence as $\varepsilon \to 0$, one has

$$g^{n+1} = \sum_{\mathrm{j}=1}^{r} K_{\mathrm{j}}^{n+1} V_{\mathrm{j}}^{n+1} = -\mathbf{v} \cdot \frac{\nabla_{\mathbf{x}} \rho^{n+\frac{1}{2}}}{\sigma^S}. \tag{2.2.44}$$

Finally, substituting (2.2.44) into (2.2.37) and (2.2.43) into (2.2.42), we have after the first time step ($n \geq 1$):

$$\begin{aligned} \frac{\rho^{n+\frac{1}{2}} - \rho^n}{\Delta t/2} &= \nabla_{\mathbf{x}} \cdot \left(\frac{1}{3\sigma^S} \nabla_{\mathbf{x}} \rho^{n-\frac{1}{2}} \right) - \sigma^A \rho^n + G, \\ \frac{\rho^{n+1} - \rho^n}{\Delta t} &= \nabla_{\mathbf{x}} \cdot \left(\frac{1}{3\sigma^S} \nabla_{\mathbf{x}} \rho^{n+\frac{1}{2}} \right) - \sigma^A \rho^{n+\frac{1}{2}} + G, \end{aligned} \tag{2.2.45}$$

which is a second-order explicit RK scheme for the limiting diffusion equation (2.1.7). Therefore, the scheme is AP.

Remark 2.2.3. *There are many other choices to construct the second order scheme by altering the order of K, L and S, as long as the steps 2-4 are symmetric with respect to steps 5-7. Note that the AP property is always guaranteed due to the flexibility in the first order scheme.*

2.2.5 Fully discrete scheme

It remains for us to specify the discretization in the physical space and velocity space. This is the purpose of this section.

Velocity discretization

For the velocity space $\mathbb{S}^2$, we adopt the discrete velocity method[3]. The velocity points $\{\mathbf{v}_{\mathrm{i}}\}_{\mathrm{i}=1,\dots,N_{\mathbf{v}}}$ and weights $\{w_{\mathrm{i}}\}_{\mathrm{i}=1,\dots,N_{\mathbf{v}}}$ are chosen according to the Lebedev quadrature on $\mathbb{S}^2$. Then all the integrals of the form $\langle F(\mathbf{v}) \rangle_{\mathbf{v}}$ are approximated as

$$\langle F(\mathbf{v}) \rangle_{\mathbf{v}} \approx \sum_{\mathrm{i}=1}^{N_{\mathbf{v}}} w_{\mathrm{i}} F(\mathbf{v}_{\mathrm{i}}). \tag{2.2.46}$$

[3]↑In the context of radiative transfer, this is usually referred to as discrete ordinates or S_N method.

Spatial discretization

For the physical space $\Omega_{\mathbf{x}}$, we assume the third dimension is homogeneous and the domain is rectangular so that we consider $\mathbf{x} = (x, y) \in [a, b] \times [c, d]$. For simplicity, we assume periodic boundary condition.

To obtain the asymptotic limit in a more compact stencil, we adopt the 2D staggered grid proposed in [60]. We divide the x and y directions uniformly into N_x and N_y cells with size $\Delta x = (b-a)/N_x$, $\Delta y = (c-d)/N_y$, respectively. We denote the vertices by $x_k = a + k\Delta x$, $y_l = c + l\Delta y$ $(k = 0, \ldots, N_x,\ l = 0, \ldots, N_y)$, and the cell centers by $x_{k+\frac{1}{2}} = a + (k + \frac{1}{2})\Delta x$, $y_{l+\frac{1}{2}} = c + (l + \frac{1}{2})\Delta y$ $(k = 0, \ldots, N_x - 1,\ l = 0, \ldots, N_y - 1)$. We then place the unknowns ρ and g as in Figure 2.1. Namely,

- ρ is located at the vertices (x_k, y_l) and cell centers $(x_{k+\frac{1}{2}}, y_{l+\frac{1}{2}})$, i.e., the red dots in the figure;
- g (hence $\{K_{\mathrm{i}}, X_{\mathrm{i}}\}_{\mathrm{i}=1,\ldots,r}$) is located at the face centers $(x_{k+\frac{1}{2}}, y_l)$ and $(x_k, y_{l+\frac{1}{2}})$, i.e., the blue diamonds in the figure.

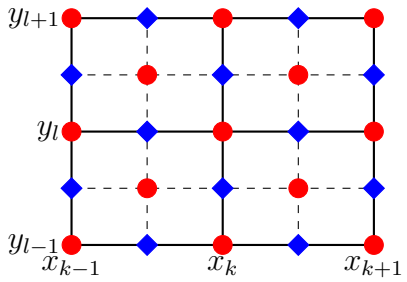

Figure 2.1. The staggered grids. ρ is located at the red dots; g (hence $\{K_{\mathrm{i}}, X_{\mathrm{i}}\}_{\mathrm{i}=1,\ldots,r}$) is located at the blue diamonds.

In the following, we describe a second order finite difference method in space. We use simplified notations such as $\rho_{k,l} = \rho(x_k, y_l)$, $(K_{\mathrm{i}})_{k+\frac{1}{2},l} = K_{\mathrm{i}}(x_{k+\frac{1}{2}}, y_l)$ to denote numerical solutions evaluated at the corresponding grid points. We use the first order K-L-S-ρ scheme in time. The discussion for other time discretization methods is similar.

- K-step

Note that the system (2.2.15), in matrix form, can be written as

$$\frac{\boldsymbol{K}^{n+1}-\boldsymbol{K}^{n}}{\Delta t}=-\boldsymbol{V}^{1}\partial_{x}\boldsymbol{K}^{n}-\boldsymbol{V}^{2}\partial_{y}\boldsymbol{K}^{n}+\ldots, \tag{2.2.47}$$

where $\boldsymbol{K}^{n}=[K_{1}^{n},K_{2}^{n},\ldots,K_{r}^{n}]^{T}$ and

$$\boldsymbol{V}^{1}_{\mathrm{j}l}=-\frac{1}{\varepsilon}\left(\langle\xi V_{\mathrm{j}}^{n}V_{l}^{n}\rangle_{\mathbf{v}}-\frac{1}{4\boldsymbol{\pi}}\langle V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}\langle\xi V_{l}^{n}\rangle_{\mathbf{v}}\right),\quad \boldsymbol{V}^{2}_{\mathrm{j}l}=-\frac{1}{\varepsilon}\left(\langle\eta V_{\mathrm{j}}^{n}V_{l}^{n}\rangle_{\mathbf{v}}-\frac{1}{4\boldsymbol{\pi}}\langle V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}\langle\eta V_{l}^{n}\rangle_{\mathbf{v}}\right). \tag{2.2.48}$$

It is clear that the matrices $\boldsymbol{V}^{1}$ and $\boldsymbol{V}^{2}$ are not necessarily symmetric hence the system might not be hyperbolic. Therefore, to get a reasonable spatial discretization for (2.2.15), we propose to discretize the original equation (2.2.4) and then project the resulting scheme.

Specifically, we first discretize (2.2.4) as

$$\begin{aligned}\partial_{t}g=&-\frac{1}{\varepsilon}\left(I-\frac{1}{4\boldsymbol{\pi}}\langle\,\rangle_{\mathbf{v}}\right)\left(\xi^{+}D_{+}^{x}g+\xi^{-}D_{-}^{x}g\right)-\frac{1}{\varepsilon}\left(I-\frac{1}{4\boldsymbol{\pi}}\langle\,\rangle_{\mathbf{v}}\right)\left(\eta^{+}D_{+}^{y}g+\eta^{-}D_{-}^{y}g\right)\\&-\frac{1}{\varepsilon^{2}}\left(\xi D_{c}^{x}\rho+\eta D_{c}^{y}\rho\right)-\left(\frac{\sigma^{S}}{\varepsilon^{2}}+\sigma^{A}\right)g,\end{aligned} \tag{2.2.49}$$

where $\xi^{+}=\max(0,\xi)$, $\xi^{-}=\min(0,\xi)$. A second order upwind operator is applied to the spatial derivatives of g and a central difference operator is applied to the spatial derivatives of ρ. More precisely, we use

$$\begin{aligned}D_{+}^{x}g(x,y)&=\frac{3g(x,y)-4g(x-\Delta x,y)+g(x-2\Delta x,y)}{2\Delta x},\\D_{-}^{x}g(x,y)&=\frac{-3g(x,y)+4g(x+\Delta x,y)-g(x+2\Delta x,y)}{2\Delta x},\end{aligned} \tag{2.2.50}$$

and

$$D_{c}^{x}\rho(x,y)=\frac{\rho(x+\frac{1}{2}\Delta x,y)-\rho(x-\frac{1}{2}\Delta x,y)}{\Delta x}. \tag{2.2.51}$$

Derivatives in y are defined similarly.

We then project the equation (2.2.49) onto the space spanned by $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$, which yields

$$\begin{aligned}
\frac{(K_{\mathrm{j}}^{n+1})_{k+\frac{1}{2},l}-(K_{\mathrm{j}}^{n})_{k+\frac{1}{2},l}}{\Delta t} =& -\frac{1}{\varepsilon}\sum_{\mathrm{i}=1}^{r}\left(\langle \xi^{+}V_{\mathrm{j}}^{n}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}-\frac{1}{4\pi}\langle V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}\langle \xi^{+}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}\right)D_{+}^{x}(K_{\mathrm{i}}^{n})_{k+\frac{1}{2},l}\\
&-\frac{1}{\varepsilon}\sum_{\mathrm{i}=1}^{r}\left(\langle \xi^{-}V_{\mathrm{j}}^{n}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}-\frac{1}{4\pi}\langle V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}\langle \xi^{-}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}\right)D_{-}^{x}(K_{\mathrm{i}}^{n})_{k+\frac{1}{2},l}\\
&-\frac{1}{\varepsilon}\sum_{\mathrm{i}=1}^{r}\left(\langle \eta^{+}V_{\mathrm{j}}^{n}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}-\frac{1}{4\pi}\langle V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}\langle \eta^{+}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}\right)D_{+}^{y}(K_{\mathrm{i}}^{n})_{k+\frac{1}{2},l}\\
&-\frac{1}{\varepsilon}\sum_{\mathrm{i}=1}^{r}\left(\langle \eta^{-}V_{\mathrm{j}}^{n}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}-\frac{1}{4\pi}\langle V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}\langle \eta^{-}V_{\mathrm{i}}^{n}\rangle_{\mathbf{v}}\right)D_{-}^{y}(K_{\mathrm{i}}^{n})_{k+\frac{1}{2},l}\\
&-\frac{1}{\varepsilon^{2}}\langle \xi V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}D_{c}^{x}\rho_{k+\frac{1}{2},l}^{n}-\frac{1}{\varepsilon^{2}}\langle \eta V_{\mathrm{j}}^{n}\rangle_{\mathbf{v}}D_{c}^{y}\rho_{k+\frac{1}{2},l}^{n}\\
&-\frac{\sigma_{k+\frac{1}{2},l}^{S}}{\varepsilon^{2}}(K_{\mathrm{j}}^{n+1})_{k+\frac{1}{2},l}-\sigma_{k+\frac{1}{2},l}^{A}(K_{\mathrm{j}}^{n})_{k+\frac{1}{2},l}.
\end{aligned}\tag{2.2.52}$$

Here the scheme is given at the grid points $(x_{k+\frac{1}{2}},y_l)$. The scheme at the grid points $(x_k,y_{l+\frac{1}{2}})$ is similar.

- L-step and S-step

  One can add spatial discretization to (2.2.17) and (2.2.19) directly. First of all, we approximate the inner product $\langle\ \rangle_{\mathbf{x}}$ by a midpoint rule:

$$\langle F(x,y)\rangle_{\mathbf{x}}=\int_{[a,b]^2}F\,\mathrm{d}x\mathrm{d}y\approx\frac{1}{2}\Delta x\Delta y\sum_{k=1}^{N_x}\sum_{l=1}^{N_y}(F_{k+\frac{1}{2},l}+F_{k,l+\frac{1}{2}}).\tag{2.2.53}$$

  Then we approximate the spatial derivatives of ρ and X_{i} at $(x_{k+\frac{1}{2}},y_l)$ and $(x_k,y_{l+\frac{1}{2}})$ by

$$\partial_x\rho_{k+\frac{1}{2},l}\approx\frac{\rho_{k+1,l}-\rho_{k,l}}{\Delta x},\qquad \partial_x(X_{\mathrm{i}})_{k+\frac{1}{2},l}\approx\frac{(X_{\mathrm{i}})_{k+\frac{3}{2},l}-(X_{\mathrm{i}})_{k-\frac{1}{2},l}}{2\Delta x},\tag{2.2.54}$$

$$\partial_x\rho_{k,l+\frac{1}{2}}\approx\frac{\rho_{k+\frac{1}{2},l+\frac{1}{2}}-\rho_{k-\frac{1}{2},l+\frac{1}{2}}}{\Delta x},\qquad \partial_x(X_{\mathrm{i}})_{k,l+\frac{1}{2}}\approx\frac{(X_{\mathrm{i}})_{k+1,l+\frac{1}{2}}-(X_{\mathrm{i}})_{k-1,l+\frac{1}{2}}}{2\Delta x}.\tag{2.2.55}$$

  Derivatives in y are treated similarly.

- ρ-step

At the grid points (x_k, y_l), (2.2.20) is discretized as

$$\begin{aligned}\frac{\rho_{k,l}^{n+1}-\rho_{k,l}^{n}}{\Delta t} =& -\frac{1}{4\pi}\sum_{\mathrm{i,j}=1}^{r}\frac{(X_{\mathrm{i}}^{n+1})_{k+\frac{1}{2},l}-(X_{\mathrm{i}}^{n+1})_{k-\frac{1}{2},l}}{\Delta x}S_{\mathrm{ij}}^{n+1}\langle \xi V_{\mathrm{j}}^{n+1}\rangle_{\mathbf{v}}\\ &-\frac{1}{4\pi}\sum_{\mathrm{i,j}=1}^{r}\frac{(X_{\mathrm{i}}^{n+1})_{k,l+\frac{1}{2}}-(X_{\mathrm{i}}^{n+1})_{k,l-\frac{1}{2}}}{\Delta y}S_{\mathrm{ij}}^{n+1}\langle \eta V_{\mathrm{j}}^{n+1}\rangle_{\mathbf{v}}-\sigma_{k,l}^{A}\rho_{k,l}^{n}+G_{k,l}.\end{aligned} \tag{2.2.56}$$

The scheme at the grid points $(x_{k+\frac{1}{2}}, y_{l+\frac{1}{2}})$ is similar.

AP property of the fully discrete scheme

Similar to the semi-discrete case, in the limit $\varepsilon \to 0$, the K-L-S steps yield

$$\begin{aligned}g_{k+\frac{1}{2},l}^{n+1} &= \sum_{\mathrm{i,j}=1}^{r}(X_{\mathrm{i}}^{n+1})_{k+\frac{1}{2},l}S_{\mathrm{ij}}^{n+1}V_{\mathrm{j}}^{n+1} = -\frac{1}{\sigma_{k+\frac{1}{2},l}^{S}}\left(\xi\frac{\rho_{k+1,l}^{n}-\rho_{k,l}^{n}}{\Delta x}+\eta\frac{\rho_{k+\frac{1}{2},l+\frac{1}{2}}^{n}-\rho_{k+\frac{1}{2},l-\frac{1}{2}}^{n}}{\Delta y}\right),\\ g_{k,l+\frac{1}{2}}^{n+1} &= \sum_{\mathrm{i,j}=1}^{r}(X_{\mathrm{i}}^{n+1})_{k,l+\frac{1}{2}}S_{\mathrm{ij}}^{n+1}V_{\mathrm{j}}^{n+1} = -\frac{1}{\sigma_{k,l+\frac{1}{2}}^{S}}\left(\xi\frac{\rho_{k+\frac{1}{2},l+\frac{1}{2}}^{n}-\rho_{k-\frac{1}{2},l+\frac{1}{2}}^{n}}{\Delta x}+\eta\frac{\rho_{k,l+1}^{n}-\rho_{k,l}^{n}}{\Delta y}\right),\end{aligned} \tag{2.2.57}$$

which, when substituting into (2.2.56), give

$$\begin{aligned}\frac{\rho_{k,l}^{n+1}-\rho_{k,l}^{n}}{\Delta t} =& \frac{1}{3}\frac{1}{\Delta x^2}\left(\frac{\rho_{k+1,l}^{n}-\rho_{k,l}^{n}}{\sigma_{k+\frac{1}{2},l}^{S}}-\frac{\rho_{k,l}^{n}-\rho_{k-1,l}^{n}}{\sigma_{k-\frac{1}{2},l}^{S}}\right)+\frac{1}{3}\frac{1}{\Delta y^2}\left(\frac{\rho_{k,l+1}^{n}-\rho_{k,l}^{n}}{\sigma_{k,l+\frac{1}{2}}^{S}}-\frac{\rho_{k,l}^{n}-\rho_{k,l-1}^{n}}{\sigma_{k,l-\frac{1}{2}}^{S}}\right)\\ &-\sigma_{k,l}^{A}\rho_{k,l}^{n}+G_{k,l}.\end{aligned} \tag{2.2.58}$$

This is an explicit standard 5-point finite difference scheme applied to the limiting diffusion equation (2.1.7) at grid points (x_k, y_l). The limiting scheme at grid points $(x_{k+\frac{1}{2}}, y_{l+\frac{1}{2}})$ can be considered similarly. Therefore, the fully discrete scheme is also AP.

2.3 A Fourier analysis of the low-rank structure of the solution

In this section, we analyze the behavior of the solution to the linear transport equation by performing a simple Fourier analysis. Our focus is in the kinetic regime because the rank is already proved to be small in the diffusive regime.

For simplicity, we consider the 1D slab geometry $x \in [0, 2\pi]$ with periodic boundary condition, and $v \in [-1, 1]$ (so $\langle\ \rangle_v = \int_{-1}^{1} \cdot\, \mathrm{d}v$). Also we assume $\sigma^A = G = 0$. Then the macro-micro system of the linear transport equation reads:

$$\begin{aligned} \partial_t \rho &= -\frac{1}{2}\langle v \partial_x g \rangle_v, \\ \partial_t g &= -\frac{1}{\varepsilon}\left(I - \frac{1}{2}\langle\ \rangle_v\right)(v\partial_x g) - \frac{1}{\varepsilon^2} v \partial_x \rho - \frac{\sigma^S}{\varepsilon^2} g. \end{aligned} \tag{2.3.1}$$

Projecting the above system onto the Fourier space of x yields

$$\begin{aligned} \partial_t \hat{\rho}_k &= -\frac{1}{2}\mathrm{i}k\langle v \hat{g}_k \rangle_v, \\ \partial_t \hat{g}_k &= -\frac{1}{\varepsilon}\mathrm{i}k\left(v\hat{g}_k - \frac{1}{2}\langle v\hat{g}_k\rangle_v\right) - \frac{1}{\varepsilon^2}\mathrm{i}vk\hat{\rho}_k - \frac{1}{\varepsilon^2}\sum_{k_1=-\infty}^{\infty} \hat{g}_{k-k_1}\hat{\sigma}_{k_1}, \end{aligned} \tag{2.3.2}$$

where $\hat{\rho}_k(t)$, $\hat{g}_k(t, v)$ and $\hat{\sigma}_k$ are the Fourier coefficients of ρ, g and σ^S, respectively.

For a constant σ^S we have

$$\hat{\sigma}_0 = \sigma^S, \quad \hat{\sigma}_k = 0,\ k \neq 0, \tag{2.3.3}$$

and the system (2.3.2) reduces to

$$\begin{aligned} \partial_t \hat{\rho}_k &= -\frac{1}{2}\mathrm{i}k\langle v \hat{g}_k \rangle_v, \\ \partial_t \hat{g}_k &= -\frac{1}{\varepsilon}\mathrm{i}k\left(v\hat{g}_k - \frac{1}{2}\langle v\hat{g}_k\rangle_v\right) - \frac{1}{\varepsilon^2}\mathrm{i}vk\hat{\rho}_k - \frac{1}{\varepsilon^2}\sigma^S\hat{g}_k. \end{aligned} \tag{2.3.4}$$

Hence all the frequency modes are decoupled. It is clear that if initially

$$\rho(0, x) = \sum_{k=-m_0}^{m_0} \hat{\rho}_k(0)\mathrm{e}^{\mathrm{i}kx}, \quad g(0, x, v) = \sum_{k=-m_0}^{m_0} \hat{g}_k(0, v)\mathrm{e}^{\mathrm{i}kx}, \tag{2.3.5}$$

i.e., $\rho(0, x)$ and $g(0, x, v)$ are band-limited, then the latter solutions will remain in the same frequency range. In this case the solution is clearly low-rank.

This analysis is similar to the analysis conducted in [61], where it was shown that for the linearized Vlasov–Maxwell equation the solution remains low rank if it is initially in a form similar to (2.3.5). However, the present situation is different in the sense that if we have a

non-constant σ^S, as is commonly the case in practice, then even if the initial value is in that form additional Fourier modes are excited gradually with time. This is as far as we can go with such an argument.

However, it should not be taken to imply that performing a low-rank approximation is necessarily futile in such a situation. In fact, the dynamical low-rank integrator makes no assumptions that the space dependence has to take the form of a finite number of Fourier modes; this is purely an artifact of the analysis done here. Hence, just because we have an infinite number of Fourier modes does not necessarily imply the solution can not be captured by a low-rank scheme. In fact, from the numerical tests in the next section, we can see that the rank of the solution in the kinetic regime when σ^S is not constant can be rather intricate, but that often relatively small ranks are sufficient to obtain an accurate approximation to the dynamics of interest.

2.4 Numerical results

In this section, we present several numerical examples to demonstrate the accuracy and efficiency of the proposed low rank method. In all examples, we consider a two-dimensional square domain in physical space, i.e. $\mathbf{x} = (x, y) \in [a, b]^2$ and periodic boundary conditions. Note that in some of the examples (e.g., the line source problem), one has to choose a large number of points in the angular direction to obtain a reasonable solution (for both the full tensor and low rank methods). This is the well-known drawback of the discrete velocity or collocation method. If a Galerkin rather than collocation approach is adopted, one could potentially use fewer discretization points (or bases). As the focus of the paper is on the low rank method, we leave the comparison of different angular discretizations to a future study.

2.4.1 Accuracy test

We first examine the accuracy of our method (in time and space) using a manufactured solution. We choose

$$f(t, x, y, \xi, \eta, \gamma) = \exp(-t)\sin^2(2\pi x)\sin^2(2\pi y)\left(1 + \varepsilon\left(\frac{\eta + \eta^3}{3}\right)\right), \quad (x, y) \in [0, 1]^2. \quad (2.4.1)$$

The corresponding ρ and g are

$$\begin{aligned}\rho(t,x,y) &= \exp(-t)\sin^2(2\pi x)\sin^2(2\pi y),\\ g(t,x,y,\xi,\eta,\gamma) &= \exp(-t)\sin^2(2\pi x)\sin^2(2\pi y)\left(\frac{\eta+\eta^3}{3}\right).\end{aligned} \tag{2.4.2}$$

Let the scattering and absorption coefficients be $\sigma^S = 1$, $\sigma^A = 0$, then the source term G is given by

$$G(t,x,y,\xi,\eta,\gamma) = \partial_t f + \frac{1}{\varepsilon}\mathbf{v}\cdot\nabla_{\mathbf{x}} f + \frac{1}{\varepsilon} g. \tag{2.4.3}$$

We use this source term and the initial condition $\rho(t=0,x,y)$ and $g(t=0,x,y,\xi,\eta,\gamma)$ as input for our low rank method and compute the solution up to a certain time. Note that the source term here depends also on time and velocity, hence the scheme needs to be modified accordingly to take into account this dependency. We omit the details.

We consider both the first order scheme in Section 2.2.1 and the second order scheme in Section 2.2.4, coupled with the second order spatial discretization described in Section 3. We always take $N_{\mathbf{v}} = 590$ Lebedev quadrature points on the sphere $\mathbb{S}^2$ [62]. Since we know *a priori* the rank of the exact solution g is 1, we fix $r = 5$ in the low rank method which is certainly sufficient to obtain an accurate solution.

We vary the spatial size Δx and the value of ε, and evaluate the error at $t = 0.1$ as

$$\left(\Delta x^2 \sum_{k,l=1}^{N_x} \left(\rho_{\text{low rank}}(x_{k+\frac{1}{2}}, y_{l+\frac{1}{2}}) - \rho_{\text{exact}}(x_{k+\frac{1}{2}}, y_{l+\frac{1}{2}})\right)^2\right)^{\frac{1}{2}}. \tag{2.4.4}$$

Since the proposed schemes are AP, we expect them to be stable under a hyperbolic CFL condition when $\varepsilon \sim O(1)$ and a parabolic CFL condition when $\varepsilon \ll 1$. Specifically, we consider three kinds of CFL conditions: mixed CFL condition $\Delta t \sim c_1\Delta x^2 + c_2\varepsilon\Delta x$, hyperbolic CFL condition $\Delta t \sim \Delta x$, and parabolic CFL condition $\Delta t \sim \Delta x^2$.

The results of the first order (in time) scheme are shown in Figure 2.2. Under the mixed CFL condition, we expect to see first order convergence in the kinetic regime ($\varepsilon \sim O(1)$) and second order in the diffusive regime ($\varepsilon \ll 1$), which is clearly observed in Figure 2.2 (left).

Under the parabolic CFL condition, we always expect second order convergence, which is also clear in Figure 2.2 (right).

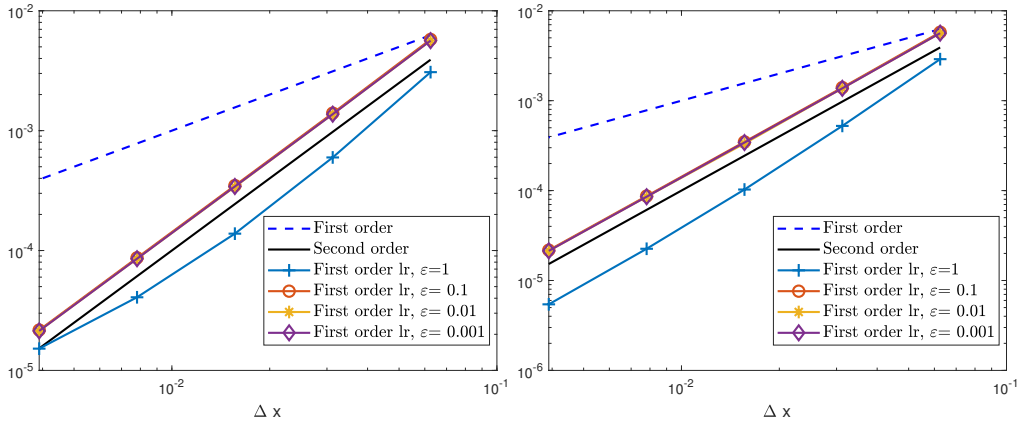

Figure 2.2. Section 2.4.1: convergence order (first order low rank scheme). l^2-error v.s. Δx. Left: mixed CFL condition $\Delta t = 0.18\Delta x^2 + 0.1\varepsilon\Delta x$. Right: parabolic CFL condition $\Delta t = 0.25\Delta x^2$. Blue dashed line and black line are reference slopes of 1 and 2, respectively.

For the second order (in time) scheme, we don't expect order higher than two in the diffusive regime since $\Delta t \sim \Delta x^2$ and the error behaves as $O(\Delta t^2 + \Delta x^2) = O(\Delta x^4 + \Delta x^2)$. Hence we only test its performance in the kinetic regime ($\varepsilon \sim O(1)$) under the hyperbolic CFL condition, where $\Delta t \sim \Delta x$ and the error is $O(\Delta t^2 + \Delta x^2) = O(\Delta x^2)$. The result is shown in Figure 2.3, where we can clearly see the uniform second order accuracy of the scheme (in contrast to the first order scheme).

2.4.2 Test with Gaussian initial value

In this test case, we consider a smooth Gaussian initial condition:

$$f(t=0,x,y,\xi,\eta,\gamma) = \frac{1}{4\pi\varsigma^2}\exp\left(-\frac{x^2+y^2}{4\varsigma^2}\right), \quad \varsigma^2 = 10^{-2}, \quad (x,y)\in[-1,1]^2, \qquad (2.4.5)$$

with zero absorption coefficient and source term $\sigma^A = G = 0$.

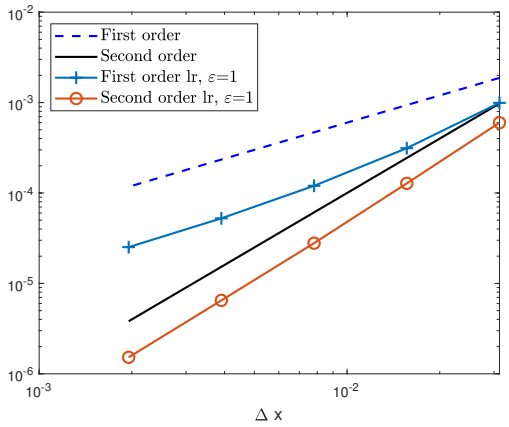

Figure 2.3. Section 2.4.1: convergence order (second order low rank scheme). l^2-error v.s. Δx. Hyperbolic CFL condition $\Delta t = 0.4\Delta x$ is used. Blue dashed line and black line are reference slopes of 1 and 2, respectively. Result of the first order scheme under the same CFL condition is plotted also for comparison.

Constant scattering coefficient σ^S

We first consider $\sigma^S \equiv 1$ and focus on the AP property of the proposed method. Therefore, we set $\varepsilon = 10^{-6}$ and compare our first order low rank method with the reference solution obtained by integrating (2.2.58), which solves the limiting diffusion equation directly. In the low rank method, we use $N_x = N_y = 128$, $N_{\mathbf{v}} = 590$ Lebedev quadrature points on $\mathbb{S}^2$, and time step $\Delta t = 0.1\Delta x^2 + 0.1\varepsilon\Delta x$, and fix the rank as $r = 5$. In solving the diffusion equation, we use $N_x = N_y = 512$ and time step $\Delta t = 0.75\Delta x^2$.

The solutions at $t = 0.1$ are shown in Figure 2.4, where they match very well. As the theory predicts, in the limiting diffusive regime, the solution g should be become rank-2. To confirm this, we track the singular values of the matrix S, see Figure 2.5. Clearly, the effective rank is 2 (two singular values are above the threshold of 10^{-5}, which is on the order of the spatial error).

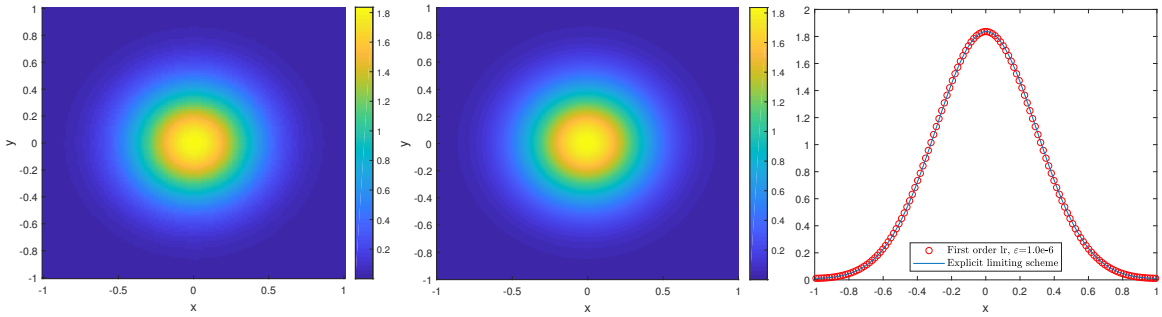

Figure 2.4. Section 2.4.2: constant scattering coefficient. Density profile of the low rank solution (left), reference solution to the limiting diffusion equation (middle), and comparison of two solutions with $y = 0$ (right).

Variable scattering coefficient σ^S

We then set $\varepsilon = 0.01$ (an intermediate regime) and consider a spatially dependent scattering coefficient

$$\sigma^S(x,y) = \begin{cases} 0.999c^4(c+\sqrt{2})^2(c-\sqrt{2})^2 + 0.001, & c = \sqrt{x^2+y^2} < 1, \\ 1, & \text{otherwise}, \end{cases} \tag{2.4.6}$$

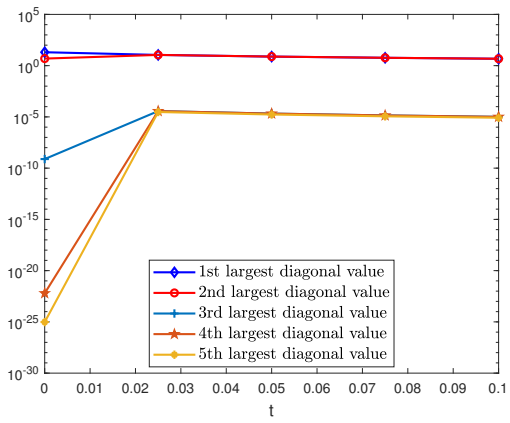

Figure 2.5. Section 2.4.2: constant scattering coefficient. Singular values of the matrix S for the low rank method.

whose profile is shown in Figure 2.6. This is a challenging test as $\frac{\sigma^S(x,y)}{\varepsilon}$ varies in a large range $[0.1, 100]$. Our aim here is to investigate the rank dependence of the low rank method and its performance compared with the full tensor method.

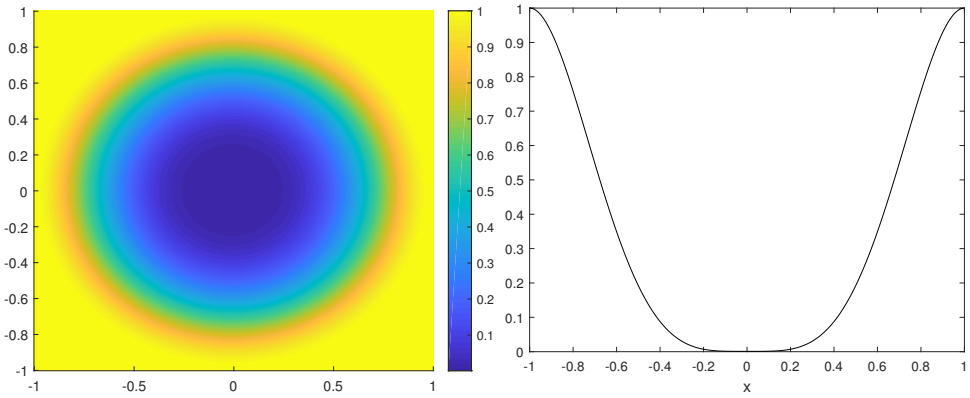

Figure 2.6. Section 2.4.2: variable scattering coefficient. Profile of σ^S (left) and a slice with $y = 0$ (right).

Specifically, we compare the first order low rank method with the first order IMEX method that solves the macro-micro decomposition of the linear transport equation directly [44] (referred to as the full tensor method in the following). We use the same spatial mesh, same CFL condition $\Delta t = 0.1 \min(\sigma^S)\Delta x^2 + 0.1\varepsilon\Delta x$, and same $N_{\mathbf{v}} = 2702$ Lebedev quadrature points on $\mathbb{S}^2$ for both methods. In the low rank method, we choose different ranks from 20 to 120.

The comparison of the low rank solution and the full tensor solution on a 256×256 mesh at different times is shown in Figure 2.7 (top). We can see that the low rank solution matches well with the full tensor solution except for rank $r = 20$. To quantitatively understand the rank dependence, we compute the difference of two solutions on the same mesh as follows

$$\left(\Delta x^2 \sum_{k,l=1}^{N_x} \left(\rho_{\text{low rank}}(x_{k+\frac{1}{2}}, y_{l+\frac{1}{2}}) - \rho_{\text{full tensor}}(x_{k+\frac{1}{2}}, y_{l+\frac{1}{2}}) \right)^2 \right)^{\frac{1}{2}}. \tag{2.4.7}$$

and track how this evolves in time under certain fixed ranks r ranging from 20 to 120. The results are shown in Figure 2.7 (bottom). The common trend is that once the rank is increased to a certain level, the difference saturates. This is because then the spatial error

dominants. Also it is clear that the rank of the solution in this problem increases gradually with time.

In addition, we record the computational time needed to compute the solution to $t = 0.012$ for both methods on an i7-8700k @3.70 GHz CPU in Figure 2.8. The speedup of the low rank method is significant, especially for a large number of spatial points N_x.

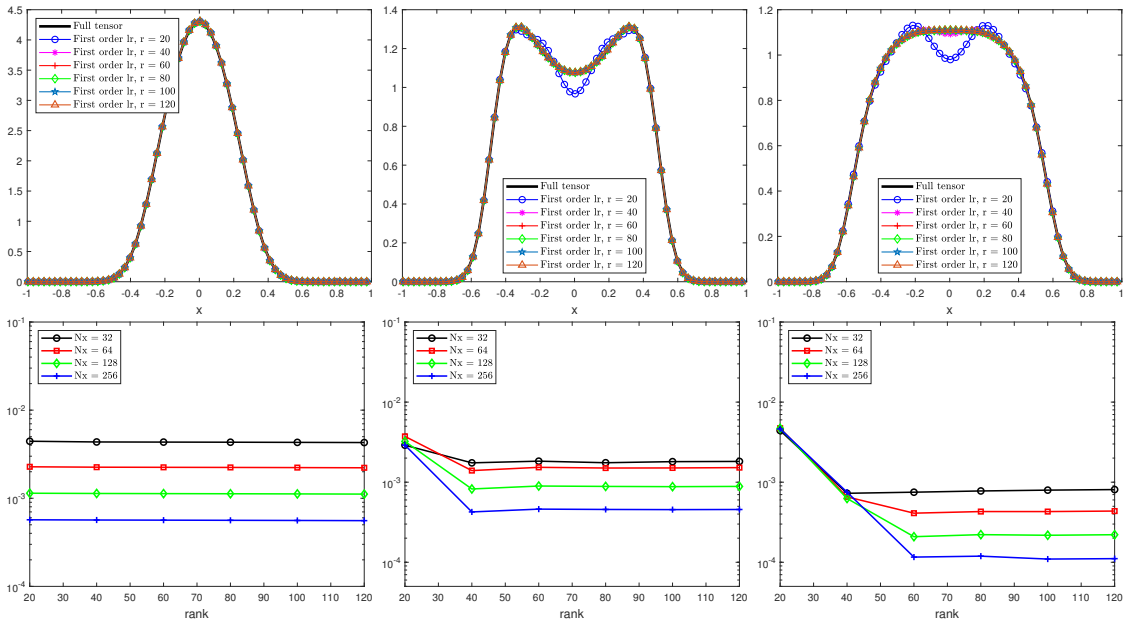

Figure 2.7. Section 2.4.2: variable scattering coefficient. Density profile with $y = 0$ of the low rank solution and full tensor solution on a 256×256 mesh at time $t = 0.002$ (top left), $t = 0.006$ (top middle), and $t = 0.010$ (top right). Difference (2.4.7) between the low rank solution and full tensor solution computed on different meshes and with different ranks at time $t = 0.002$ (bottom left), $t = 0.006$ (bottom middle), and $t = 0.010$ (bottom right).

2.4.3 Two-material test

The two-material test models a domain with different materials with discontinuities in material cross sections and source term. It is a slight modification of the lattice benchmark problem for linear transport equation. Here we choose the computational domain as $[0, 5]^2$

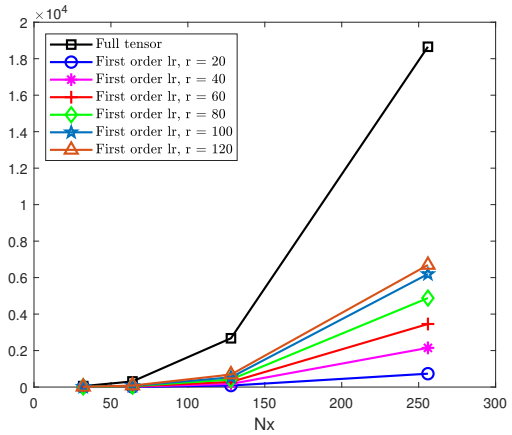

Figure 2.8. Section 2.4.2: variable scattering coefficient. Computational time (in seconds) needed for the low rank method and full tensor method to compute the solution at time $t = 0.012$.

with the absorption coefficient σ^A and scattering coefficient σ^S given as in Figure 2.9. The source term is given by

$$G(x,y) = \begin{cases} 1, & (x,y) \in [2,3]^2, \\ 0, & \text{otherwise.} \end{cases} \tag{2.4.8}$$

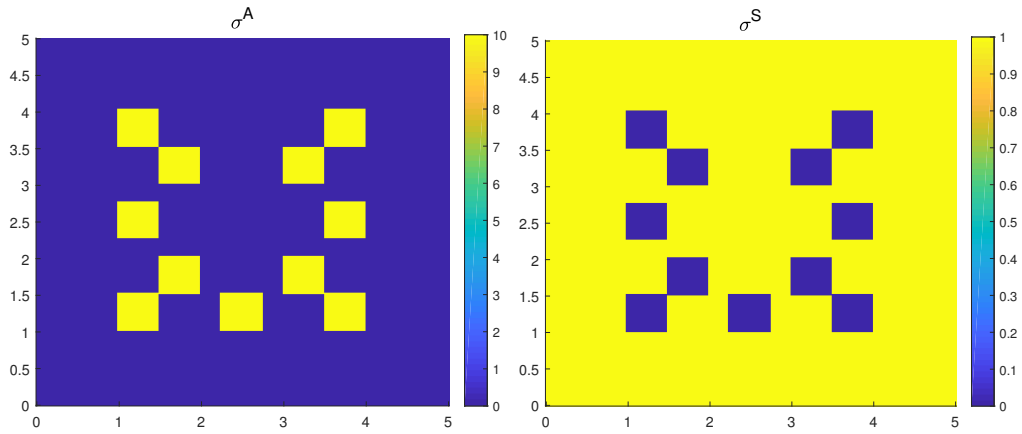

Figure 2.9. Section 2.4.3: two-material test. Profiles of absorption coefficient σ^A (left) and scattering coefficient σ^S (right). Each square block in the computational domain is a 0.5×0.5 square. In the left figure, yellow square blocks represent that $\sigma^A = 10$ and for the rest blue region $\sigma^A = 0$; in the right figure, blue square blocks represent that $\sigma^S = 0$ and for the rest yellow region $\sigma^S = 1$.

We set $\varepsilon = 1$ and compare the first order low rank method with the first order full tensor method. For both methods, we choose $N_x = N_y = 250$, $N_{\mathbf{v}} = 2702$ Lebedev quadrature points on $\mathbb{S}^2$, and same mixed CFL condition $\Delta t = 0.1 \min(\sigma^S)\Delta x^2 + 0.1\varepsilon\Delta x$. The initial condition is given by

$$f(t=0,x,y,\xi,\eta,\gamma) = \frac{1}{4\pi\varsigma^2} \exp\left(-\frac{(x-2.5)^2+(y-2.5)^2}{4\varsigma^2}\right), \quad \varsigma^2 = 10^{-2}, \quad (x,y) \in [0,5]^2. \tag{2.4.9}$$

We test different ranks from 40 to 300 in the low rank method and compare it with the full tensor solution. The error and computational time are reported in Figure 2.10. It is clear that at around rank $r = 150$, the spatial error dominates and increasing the rank further will have no gain in solution accuracy. Moreover, at $r = 150$, the efficiency of the low rank method is clearly better than the full tensor method. We then fix $r = 150$ and plot both the

low rank solution and full tensor solution at $t = 1.7$ in Figure 2.11, where a good match is obtained.

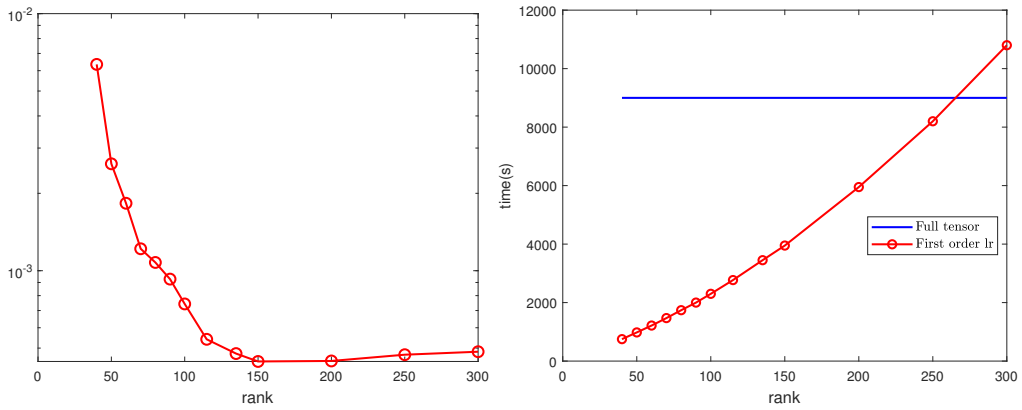

Figure 2.10. Section 2.4.3: two-material test ($\varepsilon = 1$). Difference (2.4.7) between the low rank solution with different ranks and full tensor solution at time $t = 1.7$ (left). Computational time (in seconds) needed for the low rank method with different ranks and full tensor method to compute the solution at $t = 1.7$ (right).

In addition, we consider another scenario with $\varepsilon = 0.1$. The same parameters are used as in the case of $\varepsilon = 1$, except we set the rank $r = 100$ in the low rank method (because we expect the rank of the solution to decrease as ε decreases). The solutions of the low rank method and full tensor method at time $t = 0.6$ are shown in Figure 2.12, where we again observe good agreement. An optimal (and possibly smaller) rank can be determined similarly as in Figure 2.10, we omit the result.

2.4.4 Line source test

We finally consider the line source test which is another important benchmark test for the linear transport equation. Here we approximate the initial delta function via (2.4.5) with a much smaller $\varsigma^2 = 4 \times 10^{-4}$. $\sigma^S = 1$ and $\sigma^A = G = 0$. We set $\varepsilon = 1$ and compare the first order low rank method with the full tensor method. For both methods, we choose the computational domain as $[-1.5, 1.5]^2$ with $N_x = N_y = 150$, $N_{\mathbf{v}} = 5810$ Lebedev quadrature points on $\mathbb{S}^2$, and the same mixed CFL condition $\Delta t = 0.025\Delta x^2 + 0.025\varepsilon\Delta x$. We fix the

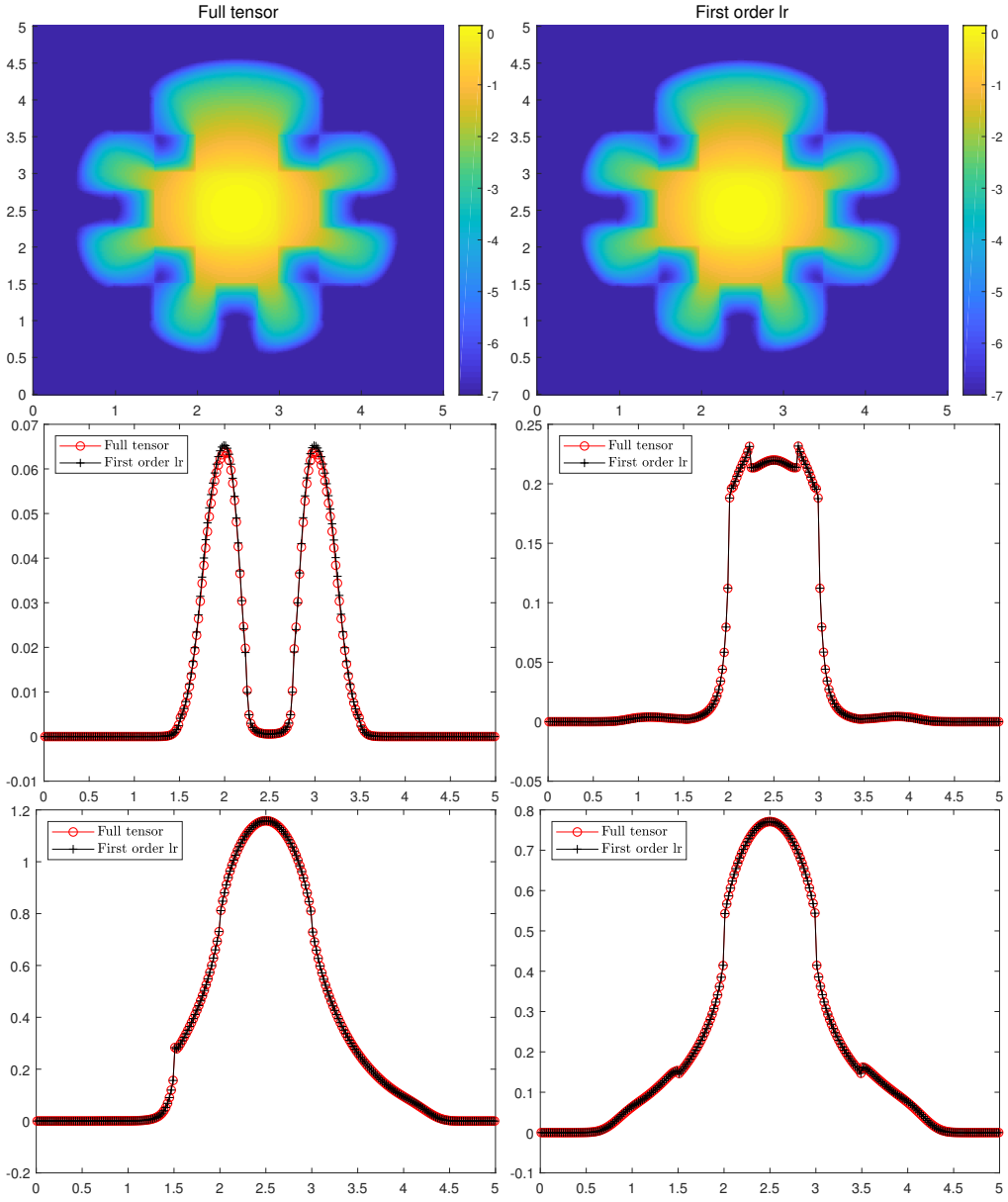

Figure 2.11. Section 2.4.3: two-material test ($\varepsilon = 1$). Contour plot of the log density at time $t = 1.7$ of the full tensor solution (top left) and low rank solution (top right) on a 250×250 mesh. Density slice of both solutions along $x = 1$ (middle left), $x = 1.5$ (middle right), $x = 2.5$ (bottom left), and $x = 3$ (bottom right). $r = 150$ in the low rank method.

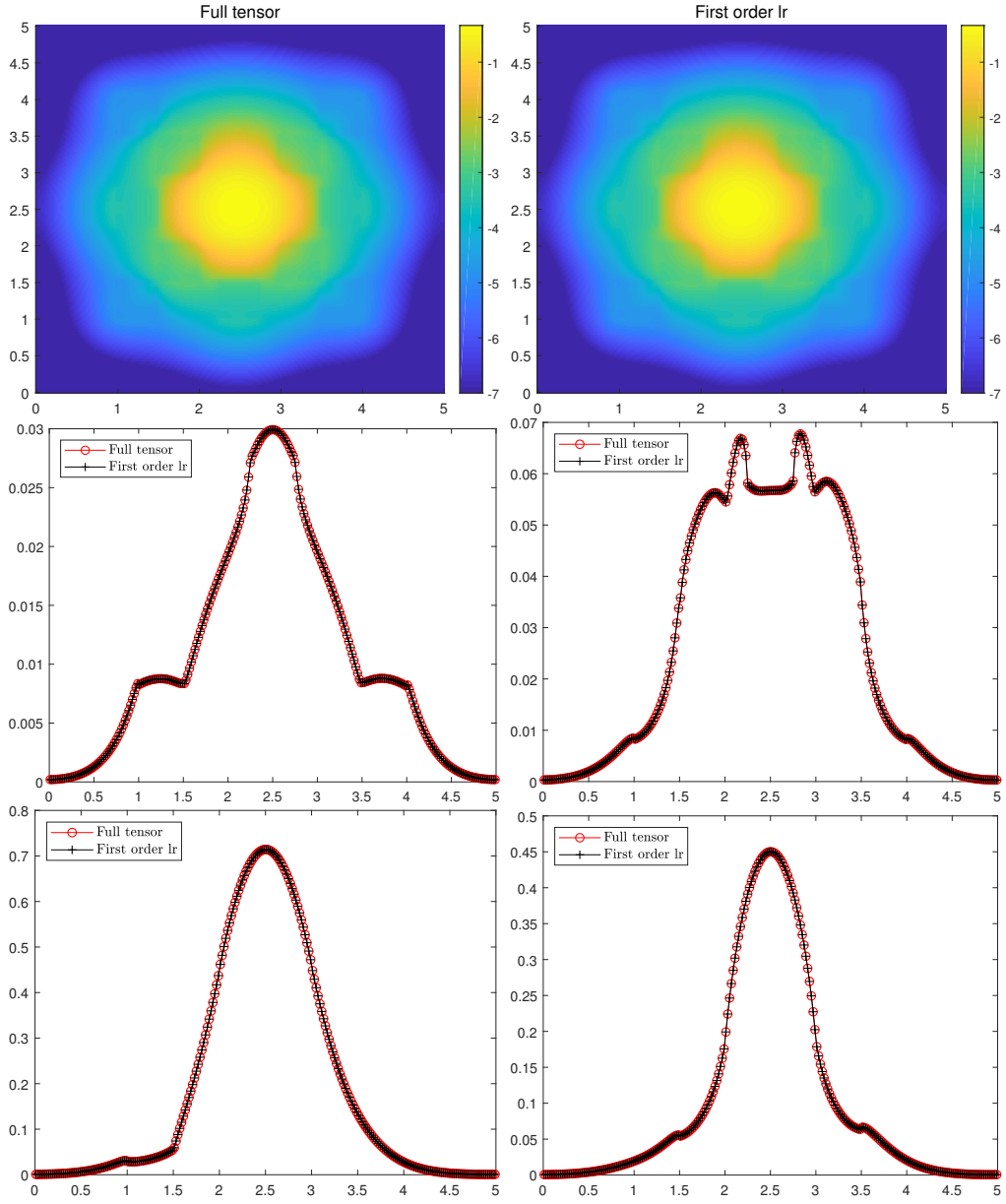

Figure 2.12. Section 2.4.3: two-material test ($\varepsilon = 0.1$). Contour plot of the log density at time $t = 0.6$ of the full tensor solution (top left) and low rank solution (top right) on a 250×250 mesh. Density slice of both solutions along $x = 1$ (middle left), $x = 1.5$ (middle right), $x = 2.5$ (bottom left), and $x = 3$ (bottom right). $r = 100$ in the low rank method.

rank as $r = 600$ in the low rank method. The density profiles of both methods at time $t = 0.7$ are shown in Figure 2.13. We can see that the solutions match well.

We would like to mention that this is a difficult problem compared to the cases considered previously. Many more points are need on the sphere to get a reasonable solution. Nevertheless, there are still oscillations in the solution (for both the full tensor and the low rank method). This is a well-known artifact in the S_N method. In addition, we found that a higher rank and a more stringent CFL condition is needed in the low rank method. We believe part of the reason are the numerical oscillations, which can be tempered by applying a proper filter or using a positivity-preserving scheme. We refer to [63], and references therein, for more details.

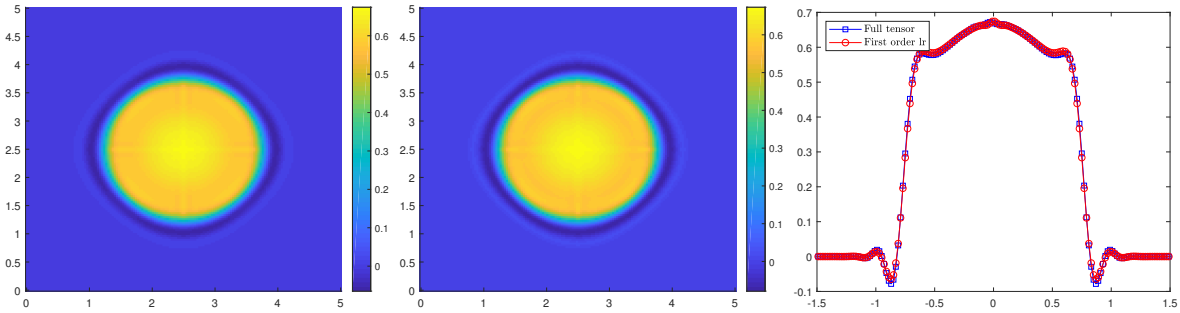

Figure 2.13. Section 2.4.4: line source test. Density profile of the full tensor solution (left) and low rank solution (middle) on a 150×150 mesh, and comparison of two solutions along $y = 0$ (right) at time $t = 0.7$. $r = 600$ in the low rank method.

2.5 Conclusions of this chapter

We have introduced a dynamical low-rank method for the multi-scale multi-dimensional linear transport equation. The method is based on a macro-micro decomposition of the equation and uses the low rank approximation only for the micro part of the solution. The key feature of the proposed scheme is that it is explicitly implementable, asymptotic-preserving in the diffusion limit, and maintains second order in both kinetic and diffusive regimes. A series of numerical examples in 2D including some well-known benchmark tests have been performed to validate the accuracy, efficiency, rank dependence, and AP property of the

proposed method. Some interesting ongoing and future work includes adaptive rank selection and the theoretical investigation of rank dependence of the solution in the kinetic regime.

3. ADAPTIVE DYNAMICAL LOW-RANK METHODS FOR FULL BOLTZMANN EQUATION

In this chapter, we will further study the rank dependence in kinetic regime ($\varepsilon \sim \mathcal{O}(1)$) and explore the adaptive low-rank methods especially for steady-state solutions of full Boltzmann equation (1.0.1). This chapter is structured as follows. In Section 3.1, we introduce the dynamic low rank method where the fully discrete schemes are discussed in detail. In Section 3.2, we presents the adaptive rank selection strategy for low-rank methods. In Section 3.3, we briefly analyze the low rank structure of normal shock wave problems. Section 3.4 presents several numerical tests using the adaptive low rank method in Boltzmann equation where we carefully examine the efficiency, accuracy and rank dependence. This chapter is concluded in Section 3.5. In this chapter, we set the Knudsen number $\varepsilon = 1$ to focus on the kinetic regime. Most of the results in this chapter come from a working paper with Dr. Jingwei Hu.

3.1 The dynamical low rank formulation and the fully discrete schemes

Similarly as in Section 2.2, we first rewrite (1.0.1) as

$$\partial_t f = -\mathbf{v} \cdot \nabla_{\mathbf{x}} f - \mathcal{Q}(f, f) := \text{RHS}, \tag{3.1.1}$$

and we constrain the probability density function $f(t, \mathbf{x}, \mathbf{v})$ to a low rank manifold $\mathbb{M}$ such that

$$f(t, \mathbf{x}, \mathbf{v}) = \sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}}(t, \mathbf{x}) S_{\mathrm{ij}}(t) V_{\mathrm{j}}(t, \mathbf{v}), \tag{3.1.2}$$

where r is the representation rank and the basis functions $\{X_{\mathrm{i}}\}_{1\le \mathrm{i}\le r} \subset L^2(\Omega_{\mathbf{x}})$ and $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r} \subset L^2(\Omega_{\mathbf{v}})$ are defined similarly as in (2.2.6). Note here we consider a finite velocity domain $\Omega_{\mathbf{v}}$ rather than the whole space $\mathbb{R}^{d_v}$ to avoid the complication in the infinite domain. This is a reasonable assumption because the majority of the numerical methods for kinetic equations need to first truncate the velocity domain and then perform the discretization. It can often be done without much loss of accuracy since f decays sufficiently fast as $\mathbf{v}$ goes to infinity.

We then project RHS onto the tangent space of $\mathbb{M}$ and use operator splitting to obtain the following three sub-flows.

$$\partial_t f = \sum_{\mathrm{j}=1}^{r} \langle V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{v}} V_{\mathrm{j}}, \tag{3.1.3}$$

$$\partial_t f = -\sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}} \langle X_{\mathrm{i}} V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{x},\mathbf{v}} V_{\mathrm{j}}, \tag{3.1.4}$$

$$\partial_t f = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}} \langle X_{\mathrm{i}}, \mathrm{RHS} \rangle_{\mathbf{x}}. \tag{3.1.5}$$

Using orthogonality condition and gauge condition, we can further simplify each sub-flow and proceed in the following three sub-steps:

- K-step: Define $K_{\mathrm{j}}(t,\mathbf{x}) = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}}(t,\mathbf{x}) S_{\mathrm{ij}}(t)$ then $f(t,\mathbf{x},\mathbf{v}) = \sum_{\mathrm{j}=1}^{r} K_{\mathrm{j}}(t,\mathbf{x}) V_{\mathrm{j}}(t,\mathbf{v})$. We can rewrite (3.1.3) as

$$\partial_t \left(\sum_{\mathrm{j}=1}^{r} K_{\mathrm{j}} V_{\mathrm{j}} \right) = \sum_{\mathrm{j}=1}^{r} \left(\partial_t K_{\mathrm{j}} V_{\mathrm{j}} + K_{\mathrm{j}} \partial_t V_{\mathrm{j}} \right) = \sum_{\mathrm{j}=1}^{r} \langle V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{v}} V_{\mathrm{j}}. \tag{3.1.6}$$

  Using the orthogonality of $\{V_{\mathrm{j}}\}_{1 \le \mathrm{j} \le r}$ and $\langle \partial_t V_{\mathrm{j}}, V_k \rangle_{\mathbf{v}} = 0$ for $1 \le \mathrm{j}, k \le r$, we have

$$\begin{aligned} \partial_t K_{\mathrm{j}} &= \langle V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{v}} \\ &= -\sum_{l=1}^{r} \langle \mathbf{v} V_{\mathrm{j}} V_l \rangle_{\mathbf{v}} \cdot \nabla_{\mathbf{x}} K_l + \sum_{m,n=1}^{r} \langle V_{\mathrm{j}} \mathcal{Q}(V_m, V_n) \rangle_{\mathbf{v}} K_m K_n, \quad \mathrm{j} = 1, \dots, r, \end{aligned} \tag{3.1.7}$$

  where the simplification of the last term relies crucially on the bilinearity of the collision operator (1.0.2) as well as the fact that collisions act locally in the physical space. It can be seen that (3.1.7) together with $\partial_t V_{\mathrm{j}} = 0$ solve (3.1.6). Since the solution to the sub-flow is unique, we thus know $\{V_{\mathrm{j}}\}_{1 \le \mathrm{j} \le r}$ remains unchanged during this sub-step.

- S-step: We can argue similarly to obtain that the sub-flow (3.1.4) is equivalent to

$$\begin{aligned} \partial_t S_{\mathrm{ij}} &= -\langle X_{\mathrm{i}} V_{\mathrm{j}}, \mathrm{RHS} \rangle_{\mathbf{x},\mathbf{v}} \\ &= \sum_{k,l=1}^{r} \langle \mathbf{v} V_{\mathrm{j}} V_l \rangle_{\mathbf{v}} \cdot \langle X_{\mathrm{i}} \nabla_{\mathbf{x}} X_k \rangle_{\mathbf{x}} S_{kl} - \sum_{k,l,m,n=1}^{r} \langle X_{\mathrm{i}} X_k X_l \rangle_{\mathbf{x}} \langle V_{\mathrm{j}} \mathcal{Q}(V_m, V_n) \rangle_{\mathbf{v}} S_{km} S_{ln}, \quad \mathrm{i,j} = 1, \dots, r. \end{aligned} \tag{3.1.8}$$

During this sub-step, both $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$ and $\{X_{\mathrm{i}}\}_{1\le \mathrm{i}\le r}$ remain unchanged.

- L-step: Define $L_{\mathrm{i}}(t,\mathbf{v}) = \sum_{\mathrm{j}=1}^{r} S_{\mathrm{ij}}(t)V_{\mathrm{j}}(t,\mathbf{v})$ then $f(\mathbf{x},\mathbf{v},t) = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}}(\mathbf{x},t)L_{\mathrm{i}}(\mathbf{v},t)$. By similar arguments, the sub-flow (3.1.5) is equivalent to

$$\begin{aligned}\partial_t L_{\mathrm{i}} &= \langle X_{\mathrm{i}}, \mathrm{RHS}\rangle_{\mathbf{x}},\\ &= -\sum_{l=1}^{r} \mathbf{v}\cdot\langle X_{\mathrm{i}}\nabla_{\mathbf{x}} X_l\rangle_{\mathbf{x}} L_l + \sum_{m,n=1}^{r} \mathcal{Q}(L_m,L_n)\langle X_{\mathrm{i}} X_m X_n\rangle_{\mathbf{x}}, \quad \mathrm{i}=1,\dots,r.\end{aligned} \tag{3.1.9}$$

During this sub-step, $\{X_{\mathrm{i}}\}_{1\le \mathrm{i}\le r}$ remains unchanged.

Therefore, we have obtained a set of low rank equations (3.1.7)-(3.1.9) in the continuous setting. The task remains is to apply the proper discretization to these equations in the velocity space, physical space, and time, which we will detail in the following subsections.

3.1.1 Velocity space discretization

Examining the equations (3.1.7)-(3.1.9), we can see that all terms pertaining to the collision operator have the form of $\mathcal{Q}(h_1,h_2)$, where h_1 and h_2 are some functions of $\mathbf{v}$. Luckily this isn't much change from the original collision operator in (1.0.1) and we can apply the well-developed fast Fourier spectral methods.

Specifically, for 2D Maxwell molecules ($d_v = 2$ and $B = \mathrm{const}$) and 3D hard spheres ($d_v = 3$ and $B = \mathrm{const}|\mathbf{v}-\mathbf{v}_*|$), we can use the algorithm proposed in [30] with complexity $\mathcal{O}(M_{FF} N_{\mathbf{v}}^{d_v}\log N_{\mathbf{v}})$, where $N_{\mathbf{v}}$ is the number of points in each dimension of the velocity space and M_{FF} is the number of points used on the sphere S^{d_v-1}; for general collision kernels, we can use the algorithm proposed in [64] with complexity $\mathcal{O}(M_{FF} N_{\mathbf{v}}^{d_v+1}\log N_{\mathbf{v}})$. Both algorithms can be implemented as a discrete velocity method: one chooses an appropriate velocity domain $[-L_{\mathbf{v}}, L_{\mathbf{v}}]^{d_v}$ and uniform grid points $\{\mathbf{v}_q\}$; the collision solver takes discrete values $\{h_1(\mathbf{v}_q)\}$ and $\{h_2(\mathbf{v}_q)\}$ and outputs $\{\mathcal{Q}(h_1,h_2)(\mathbf{v}_q)\}$ on the same set of grid points. For more details, the readers can refer to [30], [64].

3.1.2 Physical space discretization

There are various ways to discretize the equations (3.1.7)-(3.1.9) in the physical space, for example, one can apply the Fourier spectral method [25] or the high resolution finite difference scheme [27] directly to these equations. Generally speaking, the conventional scheme used for the original equation needs to be tailored when solving the equations resulted from the low rank projection. The issue also becomes a bit tricky when the boundary condition is not periodic.

Here we adopt a "first discretize, then project" strategy, which is simpler because it follows directly from the scheme for the original equation. We mention that this idea is similar to the so-called kinetic flux vector splitting (KFVS) scheme [65], a well-known method for solving the compressible Euler equations derived from the kinetic equation. For simplicity, we focus on the first order upwind scheme in this work. To extend it to high order, similar strategy for the KFVS scheme [66] can be considered.

We use the one-dimensional case ($d = 1$) to illustrate the idea. Extension to high dimension with rectangular grid is straightforward as implemented in our numerical examples. Assume $\Omega_{\mathbf{x}} = [-L_x, L_x]$ with uniform grid points chosen as $x_p = -L_x + (p - \frac{1}{2})\Delta x$, $p = 1, \ldots, N_{\mathbf{x}}$, $\Delta x = \frac{2L_x}{N_{\mathbf{x}}}$. Since the transport term in the Boltzmann equation (1.0.1) is linear, it is very easy to apply the upwind scheme:

$$\begin{aligned} \partial_t f(t,x,v) = &- \frac{v+|v|}{2}\frac{f(t,x,v) - f(x-\Delta x,v,t)}{\Delta x} \\ &- \frac{v-|v|}{2}\frac{f(x+\Delta x,v,t) - f(t,x,v)}{\Delta x} + \mathcal{Q}(f(t,x,v), f(t,x,v)) \\ := &- v^+ D_+ f(t,x,v) - v^- D_- f(x,v,t) + \mathcal{Q}(f(t,x,v), f(t,x,v)), \end{aligned} \tag{3.1.10}$$

where $v^{\pm} = \frac{v \pm |v|}{2}$, and $D_{\pm}$ are first order upwind operators. $f(t,x,v)$ is evaluated at spatial uniform grid points $\{x_p\}_{p=1,\ldots,N_{\mathbf{x}}}$.

For (3.1.10), we can apply the same projection process as we did previously to (3.1.1) to obtain (i.e., the analogs of (3.1.7)-(3.1.9)):

- K-step:

$$\begin{aligned}\partial_t K_{\mathrm{j}}(t,x) = &-\sum_{l=1}^{r}\langle v^+ V_{\mathrm{j}}(t,v)V_l(t,v)\rangle_v D_+ K_l(t,x) - \sum_{l=1}^{r}\langle v^- V_{\mathrm{j}}(t,v)V_l(t,v)\rangle_v D_- K_l(t,x)\\ &+\sum_{m,n=1}^{r}\langle V_{\mathrm{j}}(t,v)\mathcal{Q}(V_m(t,v),V_n(t,v))\rangle_v K_m(t,x)K_n(t,x).\end{aligned} \tag{3.1.11}$$

- S-step:

$$\begin{aligned}\partial_t S_{\mathrm{ij}}(t) = &\sum_{k,l=1}^{r}\langle v^+ V_{\mathrm{j}}(t,v)V_l(t,v)\rangle_v\langle X_{\mathrm{i}}(t,x)D_+X_k(t,x)\rangle_x S_{kl}\\ &+\sum_{k,l=1}^{r}\langle v^- V_{\mathrm{j}}(t,v)V_l(t,v)\rangle_v\langle X_{\mathrm{i}}(t,x)D_-X_k(t,x)\rangle_x S_{kl}\\ &-\sum_{k,l,m,n=1}^{r}\langle X_{\mathrm{i}}(t,x)X_k(t,x)X_l(t,x)\rangle_x\langle V_{\mathrm{j}}(t,v)\mathcal{Q}(V_m(t,v),V_n(t,v))\rangle_v S_{km}S_{ln}.\end{aligned} \tag{3.1.12}$$

- L-step:

$$\begin{aligned}\partial_t L_{\mathrm{i}}(t,v) = &-\sum_{l=1}^{r} v^+\langle X_{\mathrm{i}}(t,x)D_+X_l(t,x)\rangle_x L_l(t,v) - \sum_{l=1}^{r} v^-\langle X_{\mathrm{i}}(t,x)D_-X_l(t,x)\rangle_x L_l(t,v)\\ &+\sum_{m,n=1}^{r}\mathcal{Q}(L_m(t,v),L_n(t,v))\langle X_{\mathrm{i}}(t,x)X_m(t,x)X_n(t,x)\rangle_x.\end{aligned} \tag{3.1.13}$$

3.1.3 Treatment of the boundary condition

In the low rank framework, boundary condition for $f(t,\mathbf{x},\mathbf{v})$ need to be transformed to the boundary condition of $\{K_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$. In fact, this transformation has a non-trivial impact on the fully discrete scheme which we shall describe in the next subsection.

For a boundary point $\mathbf{x}\in\partial\Omega_{\mathbf{x}}$ with outward pointing normal $\mathbf{n}(\mathbf{x})$ and boundary velocity $\mathbf{u}_w(\mathbf{x})$, general boundary conditions for Boltzmann equation (1.0.1) are defined through the inflow direction:

$$f(t,\mathbf{x},\mathbf{v}) = f_{bdy}(t,\mathbf{x},\mathbf{v}),\quad (\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x}) < 0. \tag{3.1.14}$$

However, there are no prescribed inflow direction defined for $\{K_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$ and we need to reconstruct the full boundary value by approximating the outflow values using values inside the domain (extrapolation):

$$f^b(t,\mathbf{x},\mathbf{v}) := \begin{cases} f_{bdy}(t,\mathbf{x},\mathbf{v}) & (\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x}) < 0 \\ f_{extrap}(t,\mathbf{x},\mathbf{v}) & (\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x}) \ge 0. \end{cases} \tag{3.1.15}$$

Accordingly, we can project the full boundary value $f^b(t,\mathbf{x},\mathbf{v})$ to the space spanned by $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$ to obtain boundary values for $\{K_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$:

$$\begin{aligned} K_{\mathrm{j}}(t,\mathbf{x}) =& \langle f^b(t,\mathbf{x},\mathbf{v}), V_{\mathrm{j}}(t,\mathbf{v})\rangle_{\mathbf{v}} \\ =& \langle f_{bdy}(t,\mathbf{x},\mathbf{v})\mathbb{1}_{(\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x})<0}, V_{\mathrm{j}}(t,\mathbf{v})\rangle_{\mathbf{v}} + \langle f_{extrap}(t,\mathbf{x},\mathbf{v})\mathbb{1}_{(\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x})\ge 0}, V_{\mathrm{j}}(t,\mathbf{v})\rangle_{\mathbf{v}} \\ =& \langle f_{bdy}(t,\mathbf{x},\mathbf{v})\mathbb{1}_{(\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x})<0}, V_{\mathrm{j}}(t,\mathbf{v})\rangle_{\mathbf{v}} + \sum_{l=1}^{r} K_l(t,\mathbf{x})\langle \mathbb{1}_{(\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x})\ge 0} V_l(t,\mathbf{v}) V_{\mathrm{j}}(t,\mathbf{v})\rangle_{\mathbf{v}}, \end{aligned} \tag{3.1.16}$$

where the K_l term appearing on the right hand side of (3.1.16) can be approximated by extrapolation since the term results from the outflow.

Two typical boundary conditions used when solving the Boltzmann equation (1.0.1) are inflow boundary and Maxwell diffusive boundary. For inflow boundary, we assume $\mathbf{u}_w(\mathbf{x}) = \mathbf{0}$ and the typical inflow boundary is given by:

$$f_{bdy}(t,\mathbf{x},\mathbf{v}) = \frac{\rho_{\mathrm{i}n}(t,\mathbf{x})}{(2\pi T_{\mathrm{i}n}(t,\mathbf{x}))^{d/2}} \exp\left(-\frac{|\mathbf{v}-\mathbf{u}_{\mathrm{i}n}(t,\mathbf{x})|^2}{2T_{\mathrm{i}n}(t,\mathbf{x})}\right), \quad \mathbf{v}\cdot\mathbf{n}(\mathbf{x}) < 0, \tag{3.1.17}$$

where $\rho_{\mathrm{i}n}$, $\mathbf{u}_{\mathrm{i}n}$ and $T_{\mathrm{i}n}$ are the density, bulk velocity and temperature of the prescribed inflow.

The Maxwell diffusive boundary condition is given by:

$$f_{bdy}(t,\mathbf{x},\mathbf{v}) = \rho_w(t,\mathbf{x}) \exp\left(-\frac{|\mathbf{v}-\mathbf{u}_w(t,\mathbf{x})|^2}{2T_w(t,\mathbf{x})}\right), \quad (\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x}) < 0, \tag{3.1.18}$$

where T_w is the wall temperature, ρ_w is determined by conservation of mass through the wall:

$$\rho_w(t,\mathbf{x}) = -\frac{\int_{(\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x})\ge 0} (\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x}) f(t,\mathbf{x},\mathbf{v})\,\mathrm{d}\mathbf{v}}{\int_{(\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x})<0} (\mathbf{v}-\mathbf{u}_w(\mathbf{x}))\cdot\mathbf{n}(\mathbf{x}) \exp\left(-\frac{|\mathbf{v}-\mathbf{u}_w(t,\mathbf{x})|^2}{2T_w(t,\mathbf{x})}\right)\,\mathrm{d}\mathbf{v}}. \tag{3.1.19}$$

3.1.4 Time discretization and the fully discrete scheme

We now add the time discretization to (3.1.7)-(3.1.9) to obtain a fully discrete scheme. Since most of the examples we are interested in this paper concern the stationary Boltzmann equation, the first order time discretization suffices. For high order method in time, the readers can refer to [53] and references therein.

Given the initial condition $f(0,\mathbf{x},\mathbf{v}) = f^0(\mathbf{x},\mathbf{v})$, we first perform the singular value decomposition $f^0(\mathbf{x},\mathbf{v}) = \sum_{\mathrm{i,j}=1}^{r} X_{\mathrm{i}}^0(\mathbf{x}) S_{\mathrm{ij}}^0 V_{\mathrm{j}}^0(\mathbf{v})$ to obtain $(X_{\mathrm{i}}^0, S_{\mathrm{ij}}^0, V_{\mathrm{j}}^0)$, where a fixed, reasonable rank r is chosen and used in the following computation.

Suppose at time step t^n, $(X_{\mathrm{i}}^n, S_{\mathrm{ij}}^n, V_{\mathrm{j}}^n)$ are available. In order to obtain $(X_{\mathrm{i}}^{n+1}, S_{\mathrm{ij}}^{n+1}, V_{\mathrm{j}}^{n+1})$ at t^{n+1}, we proceed as follows:

1. K**-step**.

   (a) Construct $K_{\mathrm{j}}^n = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}}^n S_{\mathrm{ij}}^n$.

   (b) Perform the forward Euler step in (3.1.7) to obtain K_{j}^{n+1}:

$$K_{\mathrm{j}}^{n+1} = K_{\mathrm{j}}^n - \Delta t \sum_{l=1}^{r} \left\langle \mathbf{v} V_{\mathrm{j}}^n V_l^n \right\rangle_{\mathbf{v}} \cdot \nabla_{\mathbf{x}} K_l^n + \Delta t \sum_{m,n=1}^{r} \left\langle V_{\mathrm{j}}^n \mathcal{Q}\left(V_m^n, V_n^n\right) \right\rangle_{\mathbf{v}} K_m^n K_n^n, \quad \mathrm{j} = 1,\ldots,r. \tag{3.1.20}$$

   (c) Compute the QR decomposition of $K_{\mathrm{j}}^{n+1} = \sum_{\mathrm{i}=1}^{r} X_{\mathrm{i}}^{n+1} S_{\mathrm{ij}}^{(1)}$ to obtain updated X_{i}^{n+1} and $S_{\mathrm{ij}}^{(1)}$.

   The overall arithmetic complexity of this step is $\mathcal{O}\left(r^2(rN_{\mathbf{v}}^{d_v} + rN_{\mathbf{x}}^{d_x} + MN_{\mathbf{v}}^{d_v}\log N_{\mathbf{v}})\right)$ (suppose the algorithm in [30] is used).

2. S**-step**.

   (a) Perform the forward Euler step in (3.1.8) to obtain $S_{\mathrm{ij}}^{(2)}$:

$$\begin{aligned} S_{\mathrm{ij}}^{(2)} = {} & S_{\mathrm{ij}}^{(1)} + \Delta t \sum_{l=1}^{r} \langle \mathbf{v} V_{\mathrm{j}}^n V_l^n \rangle_{\mathbf{v}} \cdot \langle X_{\mathrm{i}}^{n+1} \nabla_{\mathbf{x}} K_l^{n+1} \rangle_{\mathbf{x}} \\ & - \Delta t \sum_{m,n=1}^{r} \langle V_{\mathrm{j}}^n \mathcal{Q}(V_m^n, V_n^n) \rangle_{\mathbf{v}} \sum_{l=1}^{r} \left(\sum_{k=1}^{r} \left(\langle X_{\mathrm{i}}^{n+1} X_k^{n+1} X_l^{n+1} \rangle_{\mathbf{x}} S_{km}^{(1)} \right) S_{ln}^{(1)} \right), \quad \mathrm{i,j} = 1,\ldots,r. \end{aligned} \tag{3.1.21}$$

Since some of the quantities have been computed in the K-step, they can be reused in this step, for example, the term $\mathcal{Q}(V_m^n, V_n^n)$. Note that we changed the second term on the right hand side such that it uses $\nabla_{\mathbf{x}} K_{\mathrm{j}}^{n+1}$ rather than $\nabla_{\mathbf{x}} X_{\mathrm{j}}^{n+1}$. This is crucial because we have only available the boundary condition expressed in terms of K_{j}^{n+1} as seen in Section 3.1.3.

The overall arithmetic complexity of this step is $\mathcal{O}\left(r^2(rN_{\mathbf{x}}^{d_x} + r^2)\right)$.

3. L**-step**.

   (a) Construct $L_{\mathrm{i}}^n = \sum_{\mathrm{j}=1}^r S_{\mathrm{ij}}^{(2)} V_{\mathrm{j}}^n$ and $\tilde{K}_{\mathrm{j}}^{n+1} = \sum_{\mathrm{i}=1}^r X_{\mathrm{i}}^{n+1} S_{\mathrm{ij}}^{(2)}$.

   (b) Perform the forward Euler step in (3.1.9) to obtain L_{i}^{n+1}:

$$L_{\mathrm{i}}^{n+1} = L_{\mathrm{i}}^n - \Delta t \sum_{l=1}^r \mathbf{v} \cdot \langle X_{\mathrm{i}}^{n+1} \nabla_{\mathbf{x}} \tilde{K}_l^{n+1} \rangle_{\mathbf{x}} V_l^n + \Delta t \sum_{p,q=1}^r \mathcal{Q}(V_p^n, V_q^n) \sum_{n=1}^r \left(\sum_{m=1}^r \left(\langle X_{\mathrm{i}}^{n+1} X_m^{n+1} X_n^{n+1} \rangle_{\mathbf{x}} S_{mp}^{(2)} \right) S_{nq}^{(2)} \right), \quad \mathrm{i} = 1, \ldots, r. \tag{3.1.22}$$

   The term involving the collision operator is rearranged so that the previously computed values can be reused. For the same reason as in the S-step, $\nabla_{\mathbf{x}} \tilde{K}_{\mathrm{j}}^{n+1}$ is introduced to avoid using $\nabla_{\mathbf{x}} X_l^{n+1}$.

   (c) Compute the QR decomposition of $L_{\mathrm{i}}^{n+1} = \sum_{\mathrm{j}=1}^r S_{\mathrm{ij}}^{n+1} V_{\mathrm{j}}^{n+1}$ to obtain updated V_{j}^{n+1} and S_{ij}^{n+1}.

   The overall arithmetic complexity of this step is $\mathcal{O}\left(r^2(N_{\mathbf{x}}^{d_x} + rN_{\mathbf{v}}^{d_v} + r^2)\right)$.

To simplify the notation, we treat $\mathbf{x}$, $\mathbf{v}$ as the continuous variables in the above presentation. The discretization in $\mathbf{x}$ and $\mathbf{v}$ can be added straightforwardly following the discussion in Section 3.1.1 and Section 3.1.2. The inner products $\langle \cdot \rangle_{\mathbf{v}}$, $\langle \cdot \rangle_{\mathbf{x}}$ are evaluated using the midpoint rule at the discrete velocity and spatial grid points.

If r is small, the computational complexity of the above algorithm will be dominated by the evaluation of the collision operator $\mathcal{O}(r^2 M_{FF} N_{\mathbf{v}}^{d_v} \log N_{\mathbf{v}})$, which can be much more efficient than the full tensor method whose complexity is $\mathcal{O}(N_{\mathbf{x}}^{d_x} M_{FF} N_{\mathbf{v}}^{d_v} \log N_{\mathbf{v}})$.

3.2 An adaptive dynamical low rank method

The dynamical low rank method introduced in the last section uses a fixed rank r throughout the entire time evolution. This turns out to be a bad strategy when solving the stationary Boltzmann equation subject to inflow or Maxwell diffusive boundary conditions. The reason is two-fold: 1) The boundary keeps sending new information to the interior of the domain so that the basis X_{i}, S_{ij}, V_{j} initialized according to the initial condition is not sufficient to capture the solution at later time. Thus new basis needs to be injected to the solution over time. 2) For many benchmark tests of the Boltzmann equation, the steady state solutions are often low rank (see Section 3.3 for a partial justification). Therefore, keeping adding basis without dropping anything would unnecessarily increase the computational cost. In this section, we provide an adaptive strategy to add and delete basis during the time evolution of a dynamical low rank method.

3.2.1 Adding basis from the boundary

Assume that the full boundary values (3.1.15) are given by

$$f(t,\mathbf{x},\mathbf{v}) = f^b(t,\mathbf{x},\mathbf{v}), \quad \mathbf{x}\in\partial\Omega_{\mathbf{x}}. \tag{3.2.1}$$

Since the function $f^b(t,\mathbf{x},\mathbf{v})$ does not necessarily belong to the space spanned by $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$, using a fixed set of basis will result in information loss.

We can fix this problem by explicitly adding boundary conditions as basis into $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$. For example, at the beginning of time step t^n, in the fully discrete scheme, suppose there are N_{bx} spatial points on the boundary $\partial\Omega_{\mathbf{x}}$, $N_{\mathbf{v}}^{d_v}$ velocity points over the velocity space $\Omega_{\mathbf{v}}$ and $N_{\mathbf{x}}^{d_x}$ spatial points over the physical space $\Omega_{\mathbf{x}}$. We can represent the full boundary values $f^b(\mathbf{x},\mathbf{v},t^n)$ using a matrix $F_b\in\mathbb{R}^{N_{bx}\times N_{\mathbf{v}}^{d_v}}$. We then proceed as follows:

1. Compute SVD of F_b to obtain $F_b = U_b\Sigma_b Q_b^T$ where U_b, Q_b are orthogonal and Σ_b is diagonal with descending singular values.

2. Drop singular values in Σ_b that are smaller than 1.0e$-$10. Suppose there are r_b singular values remaining, set $\bar{Q}_b = Q_b(:, 1 : r_b) \in \mathbb{R}^{N_{\mathbf{v}}^{d_v} \times r_b}$

3. Concatenate a random matrix $X_h \in \mathbb{R}^{N_{\mathbf{x}}^{d_x} \times r_b}$ to X^n, $\bar{Q}_b$ to V^n and extend S^n with zero padding:

$$\widehat{X} = [X^n, X_h] \in \mathbb{R}^{N_{\mathbf{x}}^{d_x} \times (r+r_b)}, \quad \widehat{S} = \begin{bmatrix} S^n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{(r+r_b)\times(r+r_b)}, \quad \widehat{V} = [V^n, \bar{Q}_b] \in \mathbb{R}^{N_{\mathbf{v}}^{d_v} \times (r+r_b)}, \tag{3.2.2}$$

4. Perform QR decomposition to $\widehat{X}$ and $\widehat{V}$ to orthonormalize new basis as $\widehat{X} = X_q S_x$ and $\widehat{V} = V_q S_v$. Set $S_q = S_x \widehat{S} S_v^T$.

Then (X_q, S_q, V_q) are the new basis and we proceed as in Section 3.1.4. Using SVD to get representative basis $\bar{Q}_b$ is crucial to increase computational efficiency by reducing number of basis added r_b. If $f^b(t, \mathbf{x}, \mathbf{v}) = f^b(\mathbf{v}, t)$ is spatially homogeneous, then we can directly start at step 3 and concatenate F_b to V^n.

3.2.2 Dropping basis adaptively

To avoid the rank accumulation from the above procedure, we can decrease the rank r by dropping some small singular values of matrix $(S_{\mathrm{ij}})_{1\le \mathrm{i,j} \le r}$.

At the end of time step t^n as described in Section 3.1.4, we proceed as follows to adjust the rank:

1. Compute the SVD of $S^{n+1} = (S_{\mathrm{ij}}^{n+1})_{1\le \mathrm{i,j} \le r}$ to obtain $S^{n+1} = U\Sigma Q^T$, where $U, Q \in \mathbb{R}^{r\times r}$ are orthonormal and $\Sigma \in \mathbb{R}^{r\times r}$ is diagonal with descending singular values.

2. Drop singular values in Σ that are less than some tolerance **drop_tol**. Suppose there are r' singular values remaining, we set $\bar{U} = U(:, 1 : r')$, $\bar{\Sigma} = \Sigma(1 : r', 1 : r')$ and $\bar{Q} = Q(:, 1 : r')$. Define $\bar{S}^{n+1} = \bar{\Sigma} \approx S^{n+1}$.

3. Update the basis as $[\bar{X}_1^{n+1}, \bar{X}_2^{n+1}, \ldots, \bar{X}_{r'}^{n+1}] = [X_1^{n+1}, X_2^{n+1}, \ldots, X_r^{n+1}]\bar{U}$ and $[\bar{V}_1^{n+1}, \bar{V}_2^{n+1}, \ldots, \bar{V}_{r'}^{n+1}] = [V_1^{n+1}, V_2^{n+1}, \ldots, V_r^{n+1}]\bar{Q}$ where $\{\bar{X}_{\mathrm{i}}^{n+1}\}_{\mathrm{i}=1,\ldots,r'}$ and $\{\bar{V}_{\mathrm{i}}^{n+1}\}_{\mathrm{i}=1,\ldots,r'}$ are the updated spatial and velocity basis functions respectively.

drop_tol plays an important role in overall computational efficiency and accuracy. Large **drop_tol** causes low accuracy for some high-rank solutions and small **drop_tol** suffers from heavy computation by large computational rank. We dynamically choose **drop_tol** according to the accuracy of the current solution. More details are given in Section 3.4.1.

3.3 Normal shock problem and low rank property of the solution

Generally speaking, it is hard to predict or analyze the rank of the solution to the Boltzmann equation due to its highly nonlinear structure. As such, the dynamical low rank method introduced above is really like a black box solver since one cannot tell in advance the rank of the solution until the actual simulation is run. If the rank turns out to be high, the method becomes slow and might not be competitive to the full tensor method. Nevertheless, in this section we identify a class of problems whose solutions are indeed low rank so that we have confidence about the efficiency of the low rank method.

The normal shock problem [2] is a classical benchmark test in rarefied gas dynamics and has been used to validate all kinds of numerical methods for the nonlinear Boltzmann equation. Consider a plane shock wave perpendicular to a flow. The flow is in the x_1 direction. The gas is uniform at upstream infinity ($x_1 \to -\infty$) and downstream infinity ($x_1 \to +\infty$) and the whole flow is stationary. We are interested in the shock profile developed in this setup with various Mach numbers.

The governing equation is the following 1D stationary Boltzmann equation:

$$v_1 \partial_{x_1} f = \mathcal{Q}(f, f), \tag{3.3.1}$$

with boundary condition

$$\begin{aligned}
\lim_{x_1 \to -\infty} f(x_1, \mathbf{v}) = f_L(\mathbf{v}) = \mathcal{M}[\rho_L, \mathbf{u}_L, T_L](\mathbf{v}) = \frac{\rho_L}{(2\pi T_L)^{d/2}} \exp\left(-\frac{(v_1 - u_L)^2 + v_2^2 + ... + v_d^2}{2T_L}\right), \\
\lim_{x_1 \to +\infty} f(x_1, \mathbf{v}) = f_R(\mathbf{v}) = \mathcal{M}[\rho_R, \mathbf{u}_R, T_R](\mathbf{v}) = \frac{\rho_R}{(2\pi T_R)^{d/2}} \exp\left(-\frac{(v_1 - u_R)^2 + v_2^2 + ... + v_d^2}{2T_R}\right),
\end{aligned} \tag{3.3.2}$$

where $\mathcal{M}[\rho, \mathbf{u}, T]$ is the Maxwellian distribution; $(\rho_L, \mathbf{u}_L, T_L)$ and $(\rho_R, \mathbf{u}_R, T_R)$ are the density, bulk velocity and temperature of the upstream and downstream flows; and R is the gas constant.

The net flow of mass, momentum and energy into the shock must be equal to the ones out of the shock:

$$\int v_1 f_L(\mathbf{v}) \begin{bmatrix} 1 \\ v_1 \\ \mathbf{v}^2 \end{bmatrix} \mathrm{d}\mathbf{v} = \int v_1 f_R(\mathbf{v}) \begin{bmatrix} 1 \\ v_1 \\ \mathbf{v}^2 \end{bmatrix} \mathrm{d}\mathbf{v}. \tag{3.3.3}$$

Rewriting (3.3.3) in terms of macroscopic quantities $\rho_{L,R}$, $u_{L,R}$ and $T_{L,R}$, we have the following Rankine-Hugoniot relations

$$\begin{aligned} &\rho_L u_L = \rho_R u_R, \\ &\rho_L u_L^2 + \rho_L T_L = \rho_R u_R^2 + \rho_R T_R, \\ &\rho_L u_L \left(u_L^2 + (d_v + 2) T_L\right) = \rho_R u_R \left(u_R^2 + (d_v + 2) T_R\right). \end{aligned} \tag{3.3.4}$$

Given the upstream quantities (ρ_L, u_L, T_L) and using the upstream flow Mach number defined by

$$M_L = \frac{u_L}{(\gamma T_L)^{\frac{1}{2}}}, \quad \gamma = \frac{d_v + 2}{d_v}, \tag{3.3.5}$$

we can solve (3.3.4) to obtain

$$\rho_R = \rho_L \frac{(d_v + 1) M_L^2}{M_L^2 + d_v}, \quad u_R = u_L \frac{M_L^2 + d_v}{(d_v + 1) M_L^2}, \quad T_R = T_L \frac{((d_v + 2) M_L^2 - 1)(M_L^2 + d_v)}{(d_v + 1)^2 M_L^2}. \tag{3.3.6}$$

In the following, we consider two scenarios where one can obtain some low rank approximation to the solutions of (3.3.1)-(3.3.2).

3.3.1 Weak shock wave: $M_L = \mathcal{O}(1)$

When $M_L = 1$, it is clear from (3.3.6) that there will be no jump hence no shock. When $M_L = \mathcal{O}(1)$ but bigger than 1, a weak shock will be developed. We assume

$$M_L = 1 + \varepsilon, \tag{3.3.7}$$

where ε is a small parameter. In fact, ε is on the same order of the mean free path [67]. We then rescale x_1 according to $\tilde{x}_1 = \varepsilon x_1$. The (3.3.1) thus becomes

$$v_1 \partial_{\tilde{x}_1} f = \frac{1}{\varepsilon} \mathcal{Q}(f, f). \tag{3.3.8}$$

On the other hand, we can see from (3.3.6) that the macroscopic quantities of upstream flow and downstream flow are very close:

$$\begin{aligned} \frac{\rho_R}{\rho_L} &= 1 + \frac{d_v(M_L^2 - 1)}{M_L^2 + d_v} = 1 + \mathcal{O}(\varepsilon), \\ \frac{u_R}{u_L} &= 1 - \frac{d_v(M_L^2 - 1)}{(d_v + 1)M_L^2} = 1 + \mathcal{O}(\varepsilon), \\ \frac{T_R}{T_L} &= 1 + \frac{(d_v + 1)(M_L^4 - 1) + (M_L^2 - 1)^2}{(d_v + 1)^2 M_L^2} = 1 + \mathcal{O}(\varepsilon). \end{aligned} \tag{3.3.9}$$

Hence

$$\frac{f_R}{f_L} = 1 + \mathcal{O}(\varepsilon). \tag{3.3.10}$$

Therefore, it is reasonable to assume

$$f(\tilde{x}_1, \mathbf{v}) = f_L(\mathbf{v}) + \varepsilon f_1(\tilde{x}_1, \mathbf{v}) + \mathcal{O}(\varepsilon^2), \tag{3.3.11}$$

where $f_1(\tilde{x}_1, \mathbf{v})$ is yet to be determined.

The rest of the analysis is similar to the Hilbert expansion. Substituting (3.3.11) into (3.3.8) and matching orders, we obtain at order $\mathcal{O}(\varepsilon)$:

$$\mathcal{Q}(f_1, f_L) + \mathcal{Q}(f_L, f_1) = v_1 \partial_{\tilde{x}_1} f_L(\mathbf{v}) \equiv 0. \tag{3.3.12}$$

Using the linearized Boltzmann collision operator [6] defined by

$$L_{\mathcal{M}}(f) := \frac{1}{\mathcal{M}} \left(\mathcal{Q}(\mathcal{M}, \mathcal{M}f) + \mathcal{Q}(\mathcal{M}f, \mathcal{M}) \right), \quad \mathcal{M} \text{ is a Maxwellian,} \tag{3.3.13}$$

we can write (3.3.12) as

$$L_{f_L} \left(\frac{f_1}{f_L} \right) (\tilde{x}_1, \mathbf{v}) = 0. \tag{3.3.14}$$

The kernel property of $L_{\mathcal{M}}$ implies that $\frac{f_1}{f_L}$ must be a linear combination of collision invariants 1, $\mathbf{v}$, $|\mathbf{v}|^2$ and we may write

$$f_1(\tilde{x}_1, \mathbf{v}) = f_L(\mathbf{v}) \left(a(\tilde{x}_1) + \mathbf{b}(\tilde{x}_1) \cdot \mathbf{v} + c(\tilde{x}_1)|\mathbf{v}|^2 \right), \tag{3.3.15}$$

where a, $\mathbf{b}$ and c are functions of x_1 only. Together with (3.3.11), we have

$$f(\tilde{x}_1, \mathbf{v}) = f_L(\mathbf{v})(1 + \varepsilon a(\tilde{x}_1) + \varepsilon \mathbf{b}(\tilde{x}_1) \cdot \mathbf{v} + \varepsilon c(\tilde{x}_1)|\mathbf{v}|^2) + \mathcal{O}(\varepsilon^2). \tag{3.3.16}$$

Therefore, up to order $O(\varepsilon)$, the solution $f(\tilde{x}_1, \mathbf{v})$ is a low rank separated function in $\tilde{x}_1$ and $\mathbf{v}$.

We mention that the derivation of $\mathcal{O}(\varepsilon)$ term does not require specific properties of the collision kernel B. One can continue this process to derive $\mathcal{O}(\varepsilon^2)$ term, which is a low rank function as well and depends on the kernel B, see [67] for details.

3.3.2 Strong shock wave: $M_L \to \infty$

When M_L is very large, a strong shock wave will develop and one cannot hope for any asymptotic expansion as in the previous subsection. Over the years, people have tried to find various approximations to the solution in this regime and it turns out many heuristic solutions match well with the experiments, yet are low rank [2], [68]. Here we present one such approximation due to Mott-Smith, who obtained the first solution of Boltzmann's equation for the shock structure problem in 1951. More sophisticated approximations exist but they more or less follow a similar idea as Mott-Smith.

The starting point is a bimodal distribution (and low rank) approximation of f as

$$f(x_1, \mathbf{v}) = a_1(x_1) f_L(\mathbf{v}) + a_2(x_1) f_R(\mathbf{v}). \tag{3.3.17}$$

To satisfy the Rankine-Hugoniot equations, we must have $a_1(x_1) + a_2(x_1) \equiv 1$. We thus write $a(x_1) = a_1(x_1)$ and $a_2(x_1) = 1 - a(x_1)$. In order to determine $a(x_1)$, one additional condition is needed. The simplest way is to enforce the moment equation by multiplying (3.3.1) by $\int \cdot \, v_1^2 \, \mathrm{d}\mathbf{v}$:

$$\int v_1^3 \partial_{x_1} f \, \mathrm{d}\mathbf{v} = \int v_1^2 \mathcal{Q}(f, f) \, \mathrm{d}\mathbf{v}, \tag{3.3.18}$$

which reduces to

$$a'(x_1) \left(\rho_L u_L (u_L^2 + 3T_L) - \rho_R u_R (u_R^2 + 3T_R) \right) = \alpha a(x_1)(1 - a(x_1)), \tag{3.3.19}$$

with

$$\alpha = \int v_1^2 \left(\mathcal{Q}(f_L, f_R) + \mathcal{Q}(f_R, f_L) \right) \mathrm{d}\mathbf{v}. \tag{3.3.20}$$

Using (3.3.4), (3.3.19) can be further simplified to

$$(d-1) \rho_L u_L (T_L - T_R) a'(x_1) = -\alpha a(x_1)(1 - a(x_1)). \tag{3.3.21}$$

This equation easily integrates to

$$a(x_1) = \frac{1}{\exp(\beta x_1) + 1}, \quad \beta = \frac{\alpha}{(d_v - 1) \rho_L u_L (T_L - T_R)}. \tag{3.3.22}$$

Therefore, we have found a closed form solution in the form of (3.3.17). Note that to evaluate α, we need to make use of specific properties of the collision kernel B. Accordingly, we can see that the spatial change in ρ across the shock wave increases with increasing Mach number M_L of the upstream:

$$\frac{\rho(x_1)}{\rho_L} = \frac{1 + \frac{(d_v+1)M_L^2}{M_L^2 + d_v} \exp(\beta x_1)}{1 + \exp(\beta x_1)}. \tag{3.3.23}$$

3.4 Numerical examples

In this section, we evaluate the accuracy and efficiency of the proposed dynamical low rank method by several classical benchmark tests in rarefied gas dynamics, including normal shock wave (1D), Fourier flow (1D), lid driven cavity flow (2D), and thermally driven cavity flow (2D). All these examples concern the steady-state solution of the Boltzmann equation and we use the first order method in both time and space as described in Section 3.1, and Fourier spectral method for 2D Maxwell molecules [30] to evaluate the collision operator. The results are compared with full tensor method using the same discretization.

3.4.1 Convergence criterion

Unlike time dependent problems, we need a proper stopping criterion for solving the steady state solutions.

For the full tensor method, we define the error as

$$\mathrm{err}_{\text{full tensor}} = \|f^{n+1}_{\text{full tensor}} - f^{n}_{\text{full tensor}}\|_{L^2} = \left\langle f^{n+1}_{\text{full tensor}} - f^{n}_{\text{full tensor}}, f^{n+1}_{\text{full tensor}} - f^{n}_{\text{full tensor}} \right\rangle^{\frac{1}{2}}_{\mathbf{x},\mathbf{v}}. \tag{3.4.1}$$

For the low rank method, we define the error similarly as

$$\mathrm{err}_{\text{low rank}} = \|f^{n+1}_{\text{low rank}} - f^{n}_{\text{low rank}}\|_{L^2} = \left\langle f^{n+1}_{\text{low rank}} - f^{n}_{\text{low rank}}, f^{n+1}_{\text{low rank}} - f^{n}_{\text{low rank}} \right\rangle^{\frac{1}{2}}_{\mathbf{x},\mathbf{v}}, \tag{3.4.2}$$

where $f^{n}_{\text{low rank}} = \sum_{\mathrm{i,j}=1}^{r} X^{n}_{\mathrm{i}} S^{n}_{\mathrm{ij}} V^{n}_{\mathrm{j}}$. Rather than reconstructing $f^{n}_{\text{low rank}}$, the above error term can be broke into three pieces:

$$\begin{aligned} f^{n+1}_{\text{low rank}} - f^{n}_{\text{low rank}} &= \sum_{\mathrm{i,j}=1}^{r} X^{n+1}_{\mathrm{i}} S^{n+1}_{\mathrm{ij}} V^{n+1}_{\mathrm{j}} - \sum_{\mathrm{i,j}=1}^{r} X^{n}_{\mathrm{i}} S^{n}_{\mathrm{ij}} V^{n}_{\mathrm{j}} \\ &= \sum_{\mathrm{j}=1}^{r} \left(K^{n+1}_{\mathrm{j}} - K^{n}_{\mathrm{j}}\right) V^{n}_{\mathrm{j}} + \sum_{\mathrm{i,j}=1}^{r} X^{n+1}_{\mathrm{i}} \left(S^{(2)}_{\mathrm{ij}} - S^{(1)}_{\mathrm{ij}}\right) V^{n}_{\mathrm{j}} + \sum_{\mathrm{i}=1}^{r} X^{n+1}_{\mathrm{i}} \left(L^{n+1}_{\mathrm{i}} - L^{n}_{\mathrm{i}}\right) \\ &:= \sum_{\mathrm{j}=1}^{r} \Delta K_{\mathrm{j}} V^{n}_{\mathrm{j}} + \sum_{\mathrm{i,j}=1}^{r} X^{n+1}_{\mathrm{i}} \Delta S_{\mathrm{ij}} V^{n}_{\mathrm{j}} + \sum_{\mathrm{i}=1}^{r} X^{n+1}_{\mathrm{i}} \Delta L_{\mathrm{i}} \end{aligned} \tag{3.4.3}$$

where the notation follows Section 3.1.4. By orthogonality of $\{X_{\mathrm{i}}\}_{1\le \mathrm{i}\le r}$ and $\{V_{\mathrm{j}}\}_{1\le \mathrm{j}\le r}$, (3.4.2) can be simplified as

$$\begin{aligned}\mathrm{err}^2_{\text{low rank}} &= \left\langle f^{n+1}_{\text{low rank}} - f^{n}_{\text{low rank}}, f^{n+1}_{\text{low rank}} - f^{n}_{\text{low rank}} \right\rangle_{\mathbf{x},\mathbf{v}} \\ &= \sum_{\mathrm{j}=1}^{r} \langle \Delta K_{\mathrm{j}}, \Delta K_{\mathrm{j}} \rangle_{\mathbf{x}} + \sum_{\mathrm{i},\mathrm{j}=1}^{r} \Delta S^2_{\mathrm{ij}} + \sum_{\mathrm{i}=1}^{r} \langle \Delta L_{\mathrm{i}}, \Delta L_{\mathrm{i}} \rangle_{\mathbf{v}} + \mathrm{I} + \mathrm{II} + \mathrm{III},\end{aligned} \tag{3.4.4}$$

where I, II and III are cross terms:

$$\begin{aligned}\mathrm{I} &= 2 \sum_{\mathrm{i},\mathrm{j}=1}^{r} \left\langle \Delta K_{\mathrm{j}}, X^{n+1}_{\mathrm{i}} \right\rangle_{\mathbf{x}} \Delta S_{\mathrm{ij}}. \\ \mathrm{II} &= 2 \sum_{\mathrm{i},\mathrm{j}=1}^{r} \left\langle \Delta L_{\mathrm{i}}, V^{n}_{\mathrm{j}} \right\rangle_{\mathbf{v}} \Delta S_{\mathrm{ij}}. \\ \mathrm{III} &= 2 \sum_{\mathrm{i},\mathrm{j}=1}^{r} \left\langle \Delta K_{\mathrm{j}}, X^{n+1}_{\mathrm{i}} \right\rangle_{\mathbf{x}} \cdot \left\langle \Delta L_{\mathrm{i}}, V^{n}_{\mathrm{j}} \right\rangle_{\mathbf{v}}.\end{aligned} \tag{3.4.5}$$

We emphasize that it is crucial to evaluate $\mathrm{err}_{\text{low rank}}$ using (3.4.4)-(3.4.5), since the cost of reconstructing $f^n_{\text{low rank}}$ is $\mathcal{O}(r^2 N_{\mathbf{x}}^{d_x} N_{\mathbf{v}}^{d_v})$ which is comparable to a full tensor method.

In general, we set a fixed convergence tolerance **res_tol** and terminate the time iteration whenever $\mathrm{err}_{\text{low rank}},\ \mathrm{err}_{\text{full tensor}} \le$ **res_tol** for both the full tensor method and low rank method.

For the adaptive low rank method discussed in Section 3.2, we have

$$|\mathrm{err}_{\text{low rank}} - \mathrm{err}^{\text{ada}}_{\text{low rank}}| \le \|\bar{f}^{n+1}_{\text{low rank}} - f^{n+1}_{\text{low rank}}\|_{L^2} \le (r - r')^{\frac{1}{2}} \cdot \textbf{drop\_tol}, \tag{3.4.6}$$

where $\mathrm{err}^{\text{ada}}_{\text{low rank}} = \|\bar{f}^{n+1}_{\text{low rank}} - f^{n}_{\text{low rank}}\|_{L^2}$, $\bar{f}^{n+1}_{\text{low rank}}$ is the solution at the end of time step t^n after adding and removing basis. We dynamically set **drop_tol** $= c \cdot \mathrm{err}^{\text{ada}}_{\text{low rank}}$ and control $\mathrm{err}^{\text{ada}}_{\text{low rank}}$ through

$$\frac{1}{1 + c(r - r')^{\frac{1}{2}}} \mathrm{err}_{\text{low rank}} \le \mathrm{err}^{\text{ada}}_{\text{low rank}} \le \frac{1}{1 - c(r - r')^{\frac{1}{2}}} \mathrm{err}_{\text{low rank}}. \tag{3.4.7}$$

In the following tests, we set $c = 0.2$ and apply the adaptive dynamical low rank method with convergence criterion $\mathrm{err}^{\text{ada}}_{\text{low rank}} \le$ **res_tol**.

3.4.2 Normal shock wave

We first consider the normal shock problem (3.3.1)-(3.3.2) with several different Mach numbers.

The spatial domain is chosen as $x_1 \in [-30, 30]$ with $N_{\mathbf{x}} = 1000$. The velocity domain is $(v_1, v_2) \in [-L_{\mathbf{v}}, L_{\mathbf{v}}]^2$.

For different Mach number M_L, the initial condition is chosen as

$$\begin{aligned}
&\alpha = 0.5, \quad \beta = 0 \\
&\rho_0(x) = \frac{\tanh(\alpha(x-\beta))+1}{2(\rho_R - \rho_L)} + \rho_L, \\
&T_0(x) = \frac{\tanh(\alpha(x-\beta))+1}{2(T_R - T_L)} + T_L, \\
&\mathbf{u}_0(x) = \left(\frac{\tanh(\alpha(x-\beta))+1}{2(u_R - u_L)} + u_L, 0 \right),
\end{aligned}$$

and

$$\begin{aligned}
&\gamma = 2, \\
&(\rho_L,\ \rho_R) = (1,\ \frac{(\gamma+1)M_L^2}{(\gamma-1)M_L^2+2}), \\
&(u_L,\ u_R) = (\sqrt{\gamma}M_L,\ \frac{\rho_L u_L}{\rho_R}), \\
&(T_L,\ T_R) = (1,\ \frac{2\gamma M_L^2 - (\gamma-1)}{(\gamma+1)\rho_R}).
\end{aligned}$$

When showing the numerical results, we are mainly interested in the macroscopic quantities density $\rho(x)$, bulk velocity $v(x)$ (in first dimension) and temperature $T(x)$. Their normalized values will be presented, which are defined by

$$\widehat{\rho}(x) = \frac{\rho(x) - \rho_L}{\rho_R - \rho_L}, \quad \widehat{v}(x) = \frac{v(x) - u_R}{u_L - u_R}, \quad \widehat{T}(x) = \frac{T(x) - T_L}{T_R - T_L}.$$

Weak shock wave: Mach 1.4

In this subsection we consider Mach number to be $M_L = 1.4$ and set $N_{\mathbf{v}} = 32$, $L_v = 13.11$. We set the reference solution f_{ref} as the solution from full grid method with convergence

criterion **res_tol** $= 4.0\text{e}-10$. To compare the efficiency and accuracy of the adaptive low rank method, we vary convergence criterion **res_tol** and compare the L^2 error with reference solution as $\|f_{\text{ref}} - f_{\text{test}}\|_{L^2}$ where f_{test} is the solution from either the full grid method or the low rank method. At the same time, we record the computational time needed for both methods to reach the same convergence criterion as well.

From Figure 3.1, we can see that low rank method can achieve the same accuracy much more efficiently compared to the full grid method. In Figure 3.2, we tracked the rank evolution profile and normalized macroscopic quantities under convergence criterion **res_tol** $= 3.0\text{e}-7$. For low rank method, the numerical rank grows slowly as time evolves and is stabilized to 16 before reaching convergence criterion. Furthermore, both method match well with the reference solution.

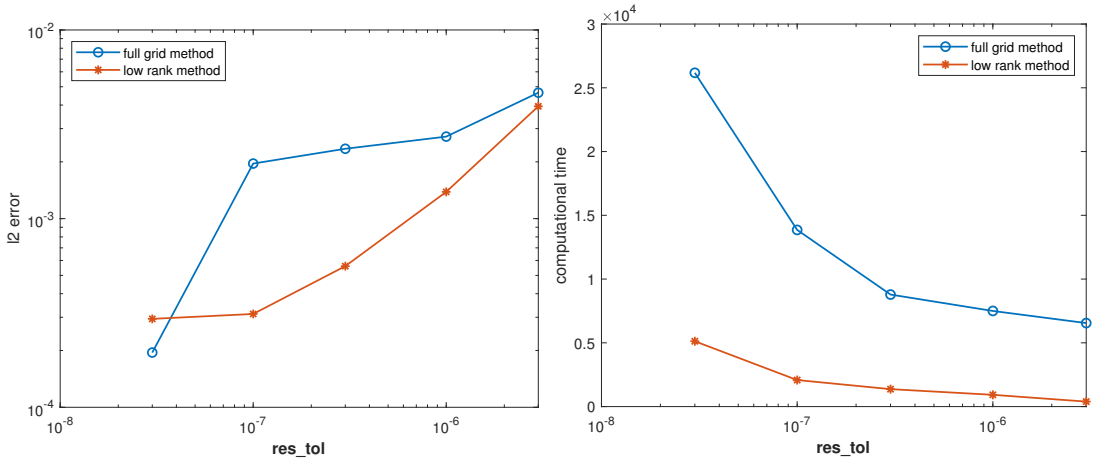

Figure 3.1. Normal shock wave (Mach 1.4): L^2 with reference solution f_{ref} (Left) and computational time in seconds (Right) for both full grid and low rank method under the same convergence criterion **res_tol**.

Strong shock wave: Mach 3.8 & Mach 6.5

In this subsection we consider the strong shock wave problems with two different large Mach numbers. For both cases, we consider the same convergence criterion **res_tol** $= 4.6\text{e}-7$.

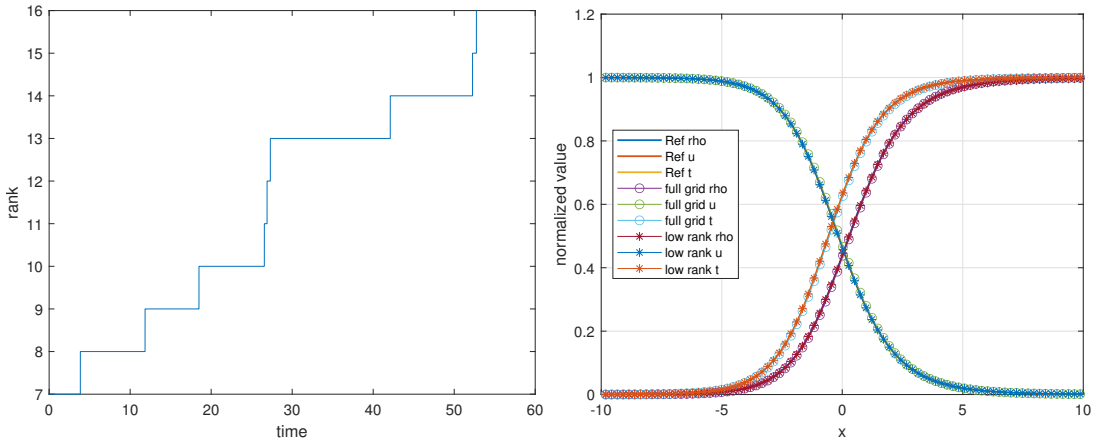

Figure 3.2. Normal shock wave (Mach 1.4): Rank evolution profile of low rank method (Left); Normalized density, bulk velocity and temperature (Right) profile of reference (**res_tol** $= 4.0\text{e}-10$), full grid and low rank method (**res_tol** $= 3.0\text{e}-7$).

For case $M_L = 3.8$, we set $N_{\mathbf{v}} = 32$ and , $L_v = 20.97$. We need computational time 18540 seconds for full grid method and 7556 seconds for low rank methods to reach convergence criterion. For case $M_L = 6.5$, we set $N_{\mathbf{v}} = 48$ and $L_v = 34.08$. We need computational time 44379 seconds for full grid method and 16157 seconds for low rank methods to reach convergence criterion. In Figure 3.3, we track the rank evolution profile as well as the normalized macroscopic quantities. The numerical rank for high Mach number shock wave behaves similarly as the weak shock wave case, except that the growing is more rapidly. A similar observation can be obtained that in both cases that the numerical ranks are stabilized before reaching convergence criterion. We also obtain good matches in the normalized macroscopic quantities profile plot.

3.4.3 Fourier flow

In the following, we will study the performance of our method with diffusive Maxwell boundary condition. We consider a Fourier heat transfer problem and the spatial domain is 1D where $x \in [0, 2]$ and velocity domain is 2D where $(v_1, v_2) \in [-L_v, L_v]^2$ where $L_v = 7.86$. We set $N_{\mathbf{x}} = 200$ and $N_{\mathbf{v}} = 32$ in each velocity dimension. For initial condition, we consider a spatially homogeneous Maxwellian with $\rho_0(x) = 1$, $\mathbf{u}_0(x) = (0, 0)$ and $T_0(x) = 1$. The diffusive Maxwell boundary condition is assumed at $x = 0$ and $x = 2$ with wall quantities $\mathbf{u}_w = (0, 0)$, $T_w = 1$ at $x = 0$ and $\mathbf{u}_w = (0, 0)$, $T_w = 1.2$ at $x = 2$. The convergence criterion is **res_tol** $= 2.0\mathrm{e} - 7$ for both full grid and low rank methods. For full grid method, we need 925 seconds to reach convergence criterion while for low rank method, we only need 509 seconds. From the rank evolution profile in Figure 3.4, the numerical rank is stabilized to 11 in a short time. We also plot the temperature profile as in Figure 3.4 and we can see that both methods match well.

3.4.4 Lid driven cavity flow

We next consider the 2D lid driven cavity flow problem to study the performance of our adaptive low rank method and the rank dependency. The spatial domain is rectangular $(x, y) \in [0, 0.5]^2$ and velocity domain is $(v_1, v_2) \in [-L_v, L_v]^2$ where $L_v = 7.86$. We set

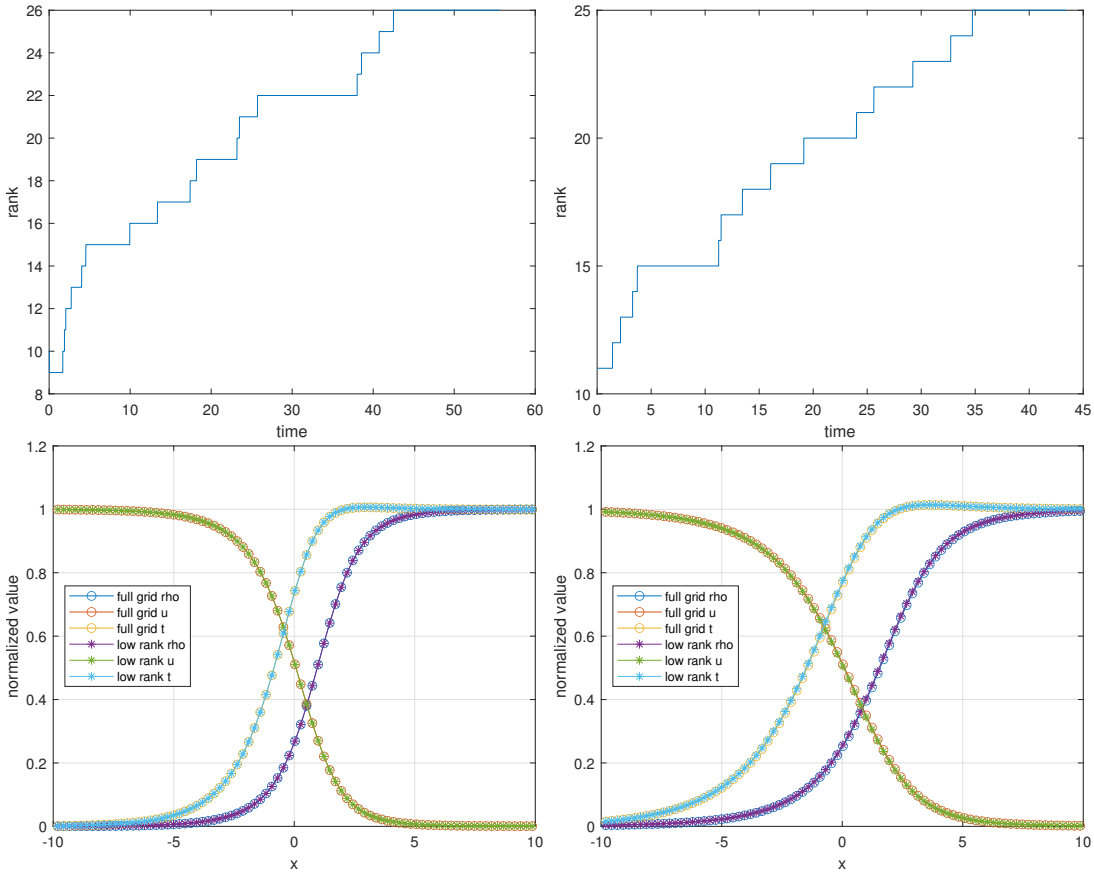

Figure 3.3. Normal shock wave (Mach 3.8 & Mach 6.5) Rank evolution profile of low rank method with Mach 3.8 (Top Left) and Mach 6.5 (Top Right); Normalized density, bulk velocity and temperature profile of full grid and low rank method ($\mathbf{res\_tol} = 4.6\text{e}-7$) with Mach 3.8 (Bottom Left) and Mach 6.5 (Bottom Right).

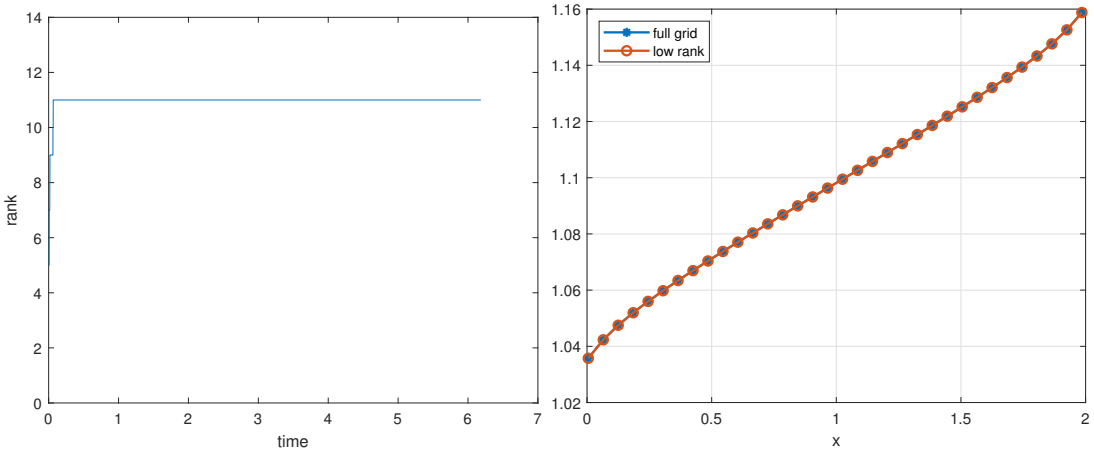

Figure 3.4. Fourier flow: Rank evolution profile (Left); Temperature profile (Right) of full grid and low rank method with convergence criterion **res_tol** = 2.0e − 7.

$N_{\mathbf{x}} = 100$ in each spatial dimension and $N_{\mathbf{v}} = 32$ in each velocity dimension. For initial condition, we consider a spatially homogeneous Maxwellian with $\rho_0(x) = 1$, $\mathbf{u}_0(x) = (1, 1)$ and $T_0(x) = 1$. The diffusive Maxwell boundary condition is assumed at boundaries of $[0, 0.5]^2$. We set the wall temperatures at $y = 0$ with $\mathbf{u}_w = (1, 0)$, $T_w = 1$ and while at $x = 0$, $x = 0.5$ and $y = 0$, we all set $\mathbf{u}_w = (0, 0)$, $T_{w,b} = 1$. The convergence criterion is **res_tol** $= 2.0\text{e} - 7$. For full grid method, we need 29043 seconds to reach convergence criterion and for low rank methods, we only need 8323 seconds. We compare the temperature and velocity profile in Figure 3.5 and a well match is obtained. We track the rank of low rank method and compare the error decaying behavior of both methods as in Figure 3.6. We can see for low rank method, as $\text{err}^{\text{ada}}_{\text{low rank}}$ is decreasing, the rank needed is increasing more rapidly than previous tests. Rank stabilization is not observed here. When time $t = 1.5$, rank needed for low rank method is as many as 100. However, low rank method can efficiently evolve distribution to a relatively low accuracy solution efficiently. From the error decaying behavior, we can see both error behaves similarly.

3.4.5 Thermally driven cavity flow

We now consider the effect of flow induced due to thermal gradients in 2D with the thermally driven cavity flow problem. The spatial domain is rectangular $(x, y) \in [0, 2]^2$ and velocity domain is $(v_1, v_2) \in [-L_v, L_v]^2$ where $L_v = 6.55$. We set $N_{\mathbf{x}} = 100$ in each spatial dimension and $N_{\mathbf{v}} = 32$ in each velocity dimension. For initial condition, we consider a spatially homogeneous Maxwellian with $\rho_0(x) = 1$, $\mathbf{u}_0(x) = (0, 0)$ and $T_0(x) = 1$. The diffusive Maxwell boundary condition is assumed at boundaries of $[0, 2]^2$. We set the wall temperatures at $y = 0$, $y = 2$ with $\mathbf{u}_w = (0, 0)$ and that T_w follows a linear function ranging from 1 to 1.2 as in Figure 3.7. At $x = 0$ and $x = 2$, we set $\mathbf{u}_w = (0, 0)$, $T_w = 1$. The convergence criterion is **res_tol** $= 2.0\text{e} - 7$. For full grid method, we need 19011 seconds to reach convergence criterion and for low rank methods, we only need 7112 seconds. We plot the temperature and velocity profile for both methods as in Figure 3.8 where we can see a good match. We track the rank trajectory of low rank method and the error decay profile as in Figure 3.9. Different from the lid driven cavity flow problem, the rank increases more

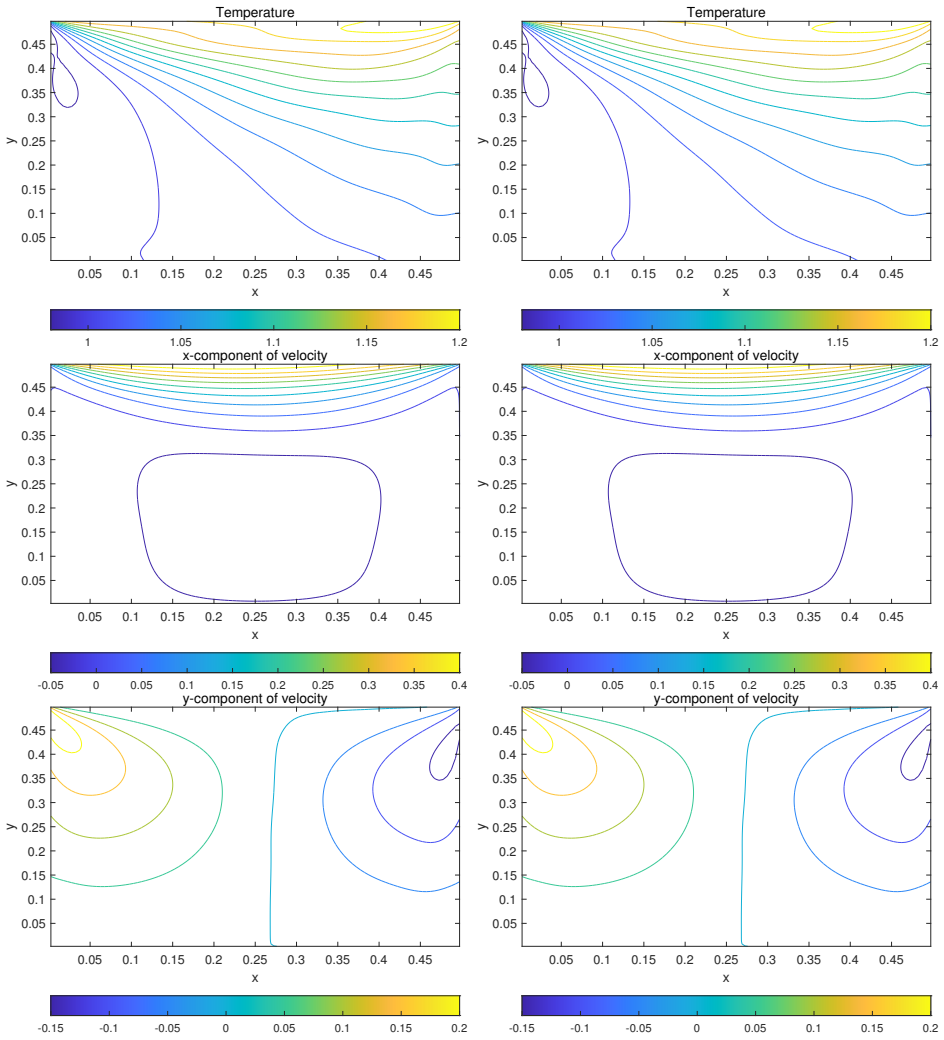

Figure 3.5. Lid driven cavity flow: Temperature profile of full grid (Top Left) and low rank (Top Right); x-component velocity of full grid (Middle Left) and low rank (Middle Right); y-component velocity of full grid (Bottom Left) and low rank (Bottom Right) method. Convergence criterion is $\mathbf{res\_tol} = 2.0\mathrm{e}-7$ for both methods

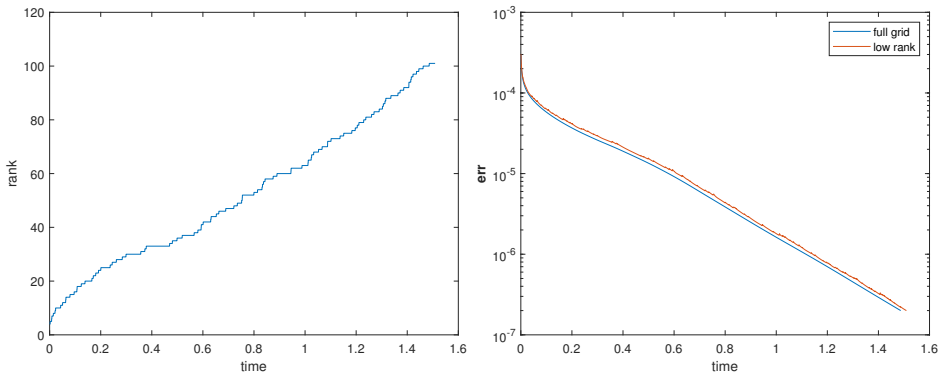

Figure 3.6. Lid driven cavity flow: Rank evolution profile of low rank method (Left); Error decaying behaviors of full grid and low rank method (Right)

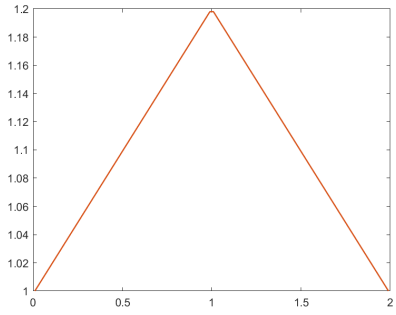

Figure 3.7. Thermally driven cavity flow: Wall temperature profile at $y = 0$ and $y = 2$

rapidly and when reaching convergence criterion, rank is increasing to as many as 120. We emphasize that for this problem, the current convergence criterion **res_tol** may not be small enough to get a highly accurate solution because of the small-scale induced velocity. A more efficient way would to be first use the low rank method to get an approximated solution and then turn to full grid method to further increase the accuracy of the solution.

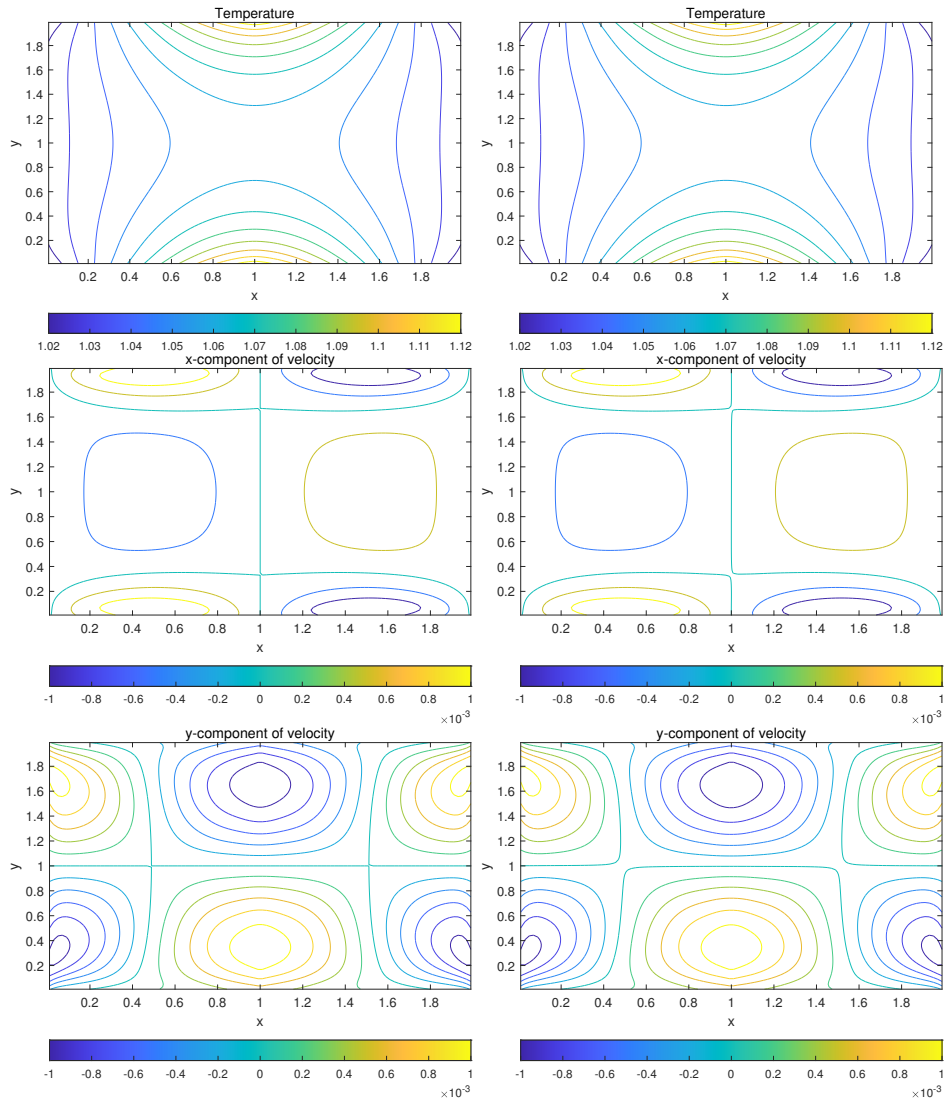

Figure 3.8. Thermally driven cavity flow: Temperature profile of full grid (Top Left) and low rank (Top Right) method; x-component velocity of full grid (Middle Left) and low rank (Middle Right); y-component velocity of full grid (Bottom Left) and low rank (Bottom Right). Convergence criterion is **res_tol** $= 2.0\text{e} - 7$ for both methods.

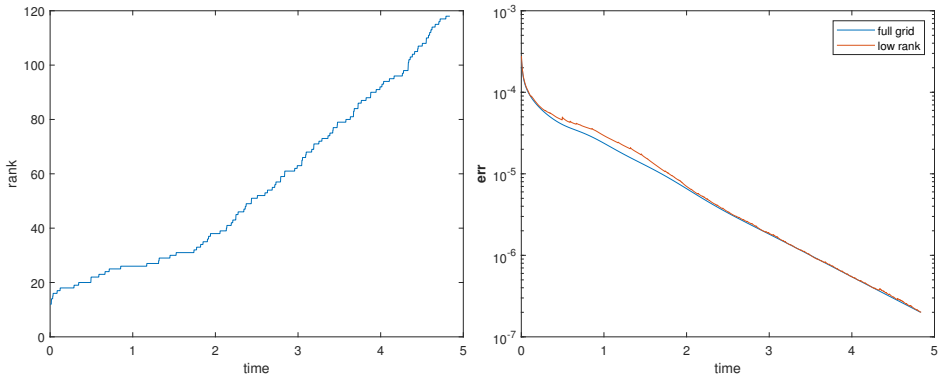

Figure 3.9. Thermally driven cavity flow: Rank evolution profile of low rank method (Left); error decaying behaviors of full grid and low rank method (Right)

3.5 Conclusions of this chapter

We have introduced an adaptive low rank method in Boltzmann equations for computation of steady state solutions. This method employed the fast Fourier method and the dynamic low rank method to obtain computational efficiency. Unlike the fixed-rank low rank method, this method can adaptively control the computational rank by monitoring residual errors of current solutions. A series of benchmark tests are conducted to show the efficiency and accuracy of the proposed method. Some interesting future work would be to analyze the theoretical rank dependency of general Boltzmann equations.

4. UNCERTAINTY QUANTIFICATION FOR KINETIC BGK EQUATION USING VARIANCE REDUCED MONTE CARLO METHODS

It remains to discuss the kinetic equations with uncertainties. We focus on the BGK model (1.0.6) of the Boltzmann equation with uncertainties. We propose an efficient control variate multilevel Monte Carlo method that can effectively reduce the variance of approximation. The rest of this chapter is structured as follows. In the next section, we introduce the BGK equation with random inputs and establish the well-posedness of the equation. The Monte Carlo methods and analysis are presented in Section 4.2, whereas in Section 4.3 we discuss their multilevel extension in a standard and control variate setting. In Section 4.4 we show the numerical results obtained with standard MC, MLMC and control variate MLMC methods. Finally some conclusions are drawn in Section 4.5. In a separate Section 4.A we report the details of the dimension reduction method and the numerical scheme adopted to solve the deterministic BGK equation. Most of the results in this chapter are extracted from [69].

4.1 The BGK equation with random inputs

In this section we formulate systematically the BGK equation with random inputs and establish the well-posedness of the equation by extending the results in [50], [70].

4.1.1 Setup of the problem

In the BGK equation, due to uncertain initial- or boundary- conditions, the resulting solution f would be a random variable taking values in the functional space, where the solution of the BGK equation (1.0.6) lies in. In most circumstances, it is the physical observables or macroscopic quantities (such as ρ, $\boldsymbol{U}$, T) at certain time that are of interest, hence we will mainly consider random variables taking values in $L^1(\Omega_{\mathbf{x}})$, where $\Omega_{\mathbf{x}}$ is the physical domain. Following the discussion in [37], we first present some basic concepts from probability theory and functional analysis.

Let $(\Omega, \mathscr{F}, \mathbb{P})$ be a probability space with Ω being the set of elementary events, $\mathscr{F}$ the corresponding σ-algebra, and $\mathbb{P}$ the probability measure mapping Ω into $[0,1]$ such that $\mathbb{P}(\Omega) = 1$. A random variable taking values in $L^1(\Omega_{\mathbf{x}})$, a separable Banach space, is defined to be a mapping $X\colon\ \Omega \to L^1(\Omega_{\mathbf{x}})$ such that for any $A \in \mathscr{G}$, the preimage $X^{-1}(A) \in \mathscr{F}$, where $X^{-1}(A) = \{w \in \Omega : X(w) \in A\}$ and $(L^1(\Omega_{\mathbf{x}}), \mathscr{G})$ is a measurable space.

To define the expectation and variance of random variables in $L^1(\Omega_{\mathbf{x}})$, we need the concept of Bochner integral by extending the Lebesgue integral theory. The strong measurable mapping $X : \Omega \to L^1(\Omega_{\mathbf{x}})$ is *Bochner integrable* if, for any probability measure $\mathbb{P}$ on the measurable space $(\Omega, \mathscr{F})$,

$$\int_{\Omega} \|X(w)\|_{L^1(\Omega_{\mathbf{x}})} \, \mathrm{d}\mathbb{P}(w) < \infty. \tag{4.1.1}$$

Moreover, any Bochner integrable random variable $X : \Omega \to L^1(\Omega_{\mathbf{x}})$ can be approximated by a sequence of simple random variables $\{X_n\}_{n\in\mathbb{N}}$ defined as follows,

$$X_n = \sum_{\mathrm{i}=1}^{N} x_{n,\mathrm{i}} \chi_{A_{n,\mathrm{i}}}, \quad A_{n,\mathrm{i}} \in \mathscr{F},\ x_{n,\mathrm{i}} \in L^1(\Omega_{\mathbf{x}}),\ N < \infty. \tag{4.1.2}$$

To get moments like expectation or central moments like variance, similar as the derivation of the Lebesgue integral, the Bochner integral is defined by taking the limit of sequences of simple random variables $\{X_n(w)\}$, for example, the k-th order moments is defined as

$$\mathbb{E}[X^k] := \int_{\Omega} X^k(w) \, \mathrm{d}\mathbb{P}(w) = \lim_{n\to\infty} \int_{\Omega} X_n^k(w) \, \mathrm{d}\mathbb{P}(w), \tag{4.1.3}$$

and the variance is defined as

$$\mathbb{V}[X] := \mathbb{E}[(X - \mathbb{E}[X])^2] = \int_{\Omega} (X(w) - \mathbb{E}[X])^2 \, \mathrm{d}\mathbb{P}(w) = \mathbb{E}[X^2] - (\mathbb{E}[X])^2. \tag{4.1.4}$$

For the error analysis, we need to introduce the Banach space $L^p(\Omega, \mathscr{F}, \mathbb{P}; L^1(\Omega_{\mathbf{x}}))$ with the norm

$$\|X\|_{L^p(\Omega;L^1(\Omega_{\mathbf{x}}))} := (\mathbb{E}[\|X\|^p_{L^1(\Omega_{\mathbf{x}})}])^{\frac{1}{p}} < \infty,\ 1 \le p < \infty; \tag{4.1.5}$$

and $L^\infty(\Omega, \mathscr{F}, \mathbb{P}; L^1(\Omega_{\mathbf{x}}))$ with the norm

$$\|X\|_{L^\infty(\Omega;L^1(\Omega_{\mathbf{x}}))} := \operatorname{ess\,sup}_{w\in\Omega}\|X\|_{L^1(\Omega_{\mathbf{x}})}. \tag{4.1.6}$$

We consider the real physical case where $d_x = d_v = 1$ and the BGK equation with random inputs hence reads

$$\begin{aligned}\partial_t f(w;t,\mathbf{x},\mathbf{v}) + \mathbf{v}\cdot\nabla_{\mathbf{x}} f(w;t,\mathbf{x},\mathbf{v}) = \frac{1}{\varepsilon}(\mathcal{M}[\rho,\boldsymbol{U},T](w;t,\mathbf{x},\mathbf{v}) - f(w;t,\mathbf{x},\mathbf{v})),\\ w\in\Omega,\ \mathbf{x}\in\Omega_{\mathbf{x}}\subset\mathbb{R}^3,\ \mathbf{v}\in\mathbb{R}^3,\ t>0,\end{aligned} \tag{4.1.7}$$

where

$$\mathcal{M}[\rho,\boldsymbol{U},T](w;t,\mathbf{x},\mathbf{v}) = \frac{\rho(w;t,\mathbf{x})}{(2\pi T(w;t,\mathbf{x}))^{\frac{3}{2}}}\exp\left(-\frac{|\mathbf{v}-\boldsymbol{U}(w;t,\mathbf{x})|^2}{2T(w;t,\mathbf{x})}\right), \tag{4.1.8}$$

with

$$\begin{aligned}&\rho(w;t,\mathbf{x}) = \int_{\mathbb{R}^3} f(w;t,\mathbf{x},\mathbf{v})\ \mathrm{d}\mathbf{v}, \qquad \boldsymbol{U}(w;t,\mathbf{x}) = \frac{1}{\rho(w;t,\mathbf{x})}\int_{\mathbb{R}^3}\mathbf{v}f(w;t,\mathbf{x},\mathbf{v})\ \mathrm{d}\mathbf{v},\\ &T(w;t,\mathbf{x}) = \frac{1}{3\rho(w;t,\mathbf{x})}\int_{\mathbb{R}^3}|\mathbf{v}-\boldsymbol{U}(w;t,\mathbf{x})|^2 f(w;t,\mathbf{x},\mathbf{v})\ \mathrm{d}\mathbf{v}.\end{aligned} \tag{4.1.9}$$

The initial condition is given as

$$f(w;0,\mathbf{x},\mathbf{v}) = f_0(w;\mathbf{x},\mathbf{v}), \quad w\in\Omega,\ \mathbf{x}\in\Omega_{\mathbf{x}}\subset\mathbb{R}^3,\ \mathbf{v}\in\mathbb{R}^3. \tag{4.1.10}$$

For the boundary condition, we consider one of the following:

- periodic boundary: $f(w;t,\mathbf{x}+\mathbf{a},\mathbf{v}) = f(w;t,\mathbf{x},\mathbf{v})$ for $\mathbf{x}\in\partial\Omega_{\mathbf{x}}$ and some $\mathbf{a}\in\mathbb{R}^3$;
- Dirichlet boundary: $f(w;t,\mathbf{x},\mathbf{v}) = g(w;t,\mathbf{x},\mathbf{v})$ for $\mathbf{x}\in\partial\Omega_{\mathbf{x}}$;
- purely diffusive Maxwell boundary: for $\mathbf{x}\in\partial\Omega_{\mathbf{x}}$,

$$f(w;t,\mathbf{x},\mathbf{v}) = \mathcal{M}[\rho_w,0,T_w](w;t,\mathbf{x},\mathbf{v}) = \mathcal{M}_w(w;t,\mathbf{x},\mathbf{v}),\ \mathbf{v}\cdot\mathbf{n}<0, \tag{4.1.11}$$

where $\mathbf{n}$ is the outward normal of $\partial\Omega_{\mathbf{x}}$ and $\mathcal{M}_w$ is given by

$$\mathcal{M}_w(w;t,\mathbf{x},\mathbf{v}) = \frac{\rho_w(w;t,\mathbf{x})}{(2\pi T_w(w;t,\mathbf{x}))^{\frac{3}{2}}} \exp\left(-\frac{|\mathbf{v}|^2}{2T_w(w;t,\mathbf{x})}\right), \tag{4.1.12}$$

where $T_w(w;t,\mathbf{x})$ is the wall temperature and $\rho_w(w;t,\mathbf{x})$ is chosen such that

$$\int_{\mathbf{v}\cdot\mathbf{n}>0} \mathbf{v}\cdot\mathbf{n}\, f(w;t,\mathbf{x},\mathbf{v})\, \mathrm{d}\mathbf{v} = -\int_{\mathbf{v}\cdot\mathbf{n}<0} \mathbf{v}\cdot\mathbf{n}\, \mathcal{M}_w(w;t,\mathbf{x},\mathbf{v})\, \mathrm{d}\mathbf{v}. \tag{4.1.13}$$

4.1.2 Well-posedness of the equation and some estimates of the macroscopic quantities

In the following, we establish the well-posedness of the BGK equation (4.1.7) with random inputs. We also obtain some estimates for the macroscopic quantities ρ, $\boldsymbol{U}$ and T. For simplicity, we assume the periodic boundary condition and consider the uncertainty only arising in the initial condition f_0.

First of all, some general estimates on the macroscopic quantities can be obtained point-wise in w following [50] for the deterministic BGK equation.

Proposition 4.1.1 ([50]). *Suppose that $f(w;t,\mathbf{x},\mathbf{v}) \geq 0$. Define $\rho(w;t,\mathbf{x})$, $\boldsymbol{U}(w;t,\mathbf{x})$, $T(w;t,\mathbf{x})$ according to (4.1.9). Moreover, set*

$$N_q(f)(w;t) := \sup_{\mathbf{x}} \sup_{\mathbf{v}} f(w;t,\mathbf{x},\mathbf{v})|\mathbf{v}|^q, \quad q \geq 0. \tag{4.1.14}$$

Then the following estimates hold:

$$\frac{\rho(w;t,\mathbf{x})}{T(w;t,\mathbf{x})^{\frac{3}{2}}} \leq C_0 N_0(f), \tag{4.1.15}$$

$$\rho(w;t,\mathbf{x})(3T(w;t,\mathbf{x}) + |\boldsymbol{U}(w;t,\mathbf{x})|^2)^{\frac{q-3}{2}} \leq C_q N_q(f), \quad \text{for } q > 5, \tag{4.1.16}$$

where C_0, C_q are some positive constants.

Based on the above estimates, one can obtain the existence and uniqueness of the solution to (4.1.7) also following [50] in a point-wise manner in w.

Theorem 4.1.1 ([50]). *Set*

$$\mathbb{N}_q(f)(w;t) := \sup_{\mathbf{x}} \sup_{\mathbf{v}} f(w;t,\mathbf{x},\mathbf{v})(1+|\mathbf{v}|^q), \tag{4.1.17}$$

then, by definition, $N_q(f) \le \mathbb{N}_q(f)$. *Suppose that the initial condition* $f_0(w;\mathbf{x},\mathbf{v}) \ge 0$ *and that for some* $q > 5$,

$$\begin{aligned} &\mathbb{N}_q(f_0)(w) = \sup_{\mathbf{x}} \sup_{\mathbf{v}} f_0(w;\mathbf{x},\mathbf{v})(1+|\mathbf{v}|^q), \\ &\sup_{w} \mathbb{N}_q(f_0)(w) \le A_0 < \infty, \end{aligned} \tag{4.1.18}$$

and

$$\begin{aligned} &\gamma(w;t,\mathbf{x}) := \int_{\mathbb{R}^3} f_0(w;\mathbf{x}-\mathbf{v}t,\mathbf{v})\ \mathrm{d}\mathbf{v}, \\ &\inf_{w} \inf_{\mathbf{x}} \inf_{t} \gamma(w;t,\mathbf{x}) \ge A_1 > 0, \end{aligned} \tag{4.1.19}$$

then, for fixed Knudsen number $\varepsilon > 0$, *there exists a unique mild solution of the initial-value problem* (4.1.7)-(4.1.10) *with periodic boundary condition.*

Moreover, for all $t > 0$, *the following bounds hold:*

$$\mathbb{N}_0(f)(w;t) \le A_0 \exp\left(\frac{C_0}{\varepsilon}t\right), \quad \mathbb{N}_q(f)(w;t) \le A_0 \exp\left(\frac{C_q}{\varepsilon}t\right), \tag{4.1.20}$$

$$\inf_{\mathbf{x}} \rho(w;t,\mathbf{x}) \ge A_1 \exp\left(-\frac{t}{\varepsilon}\right), \tag{4.1.21}$$

where C_0 *and* C_q *are the same constants appearing in Proposition 4.1.1.*

As a direct consequence of Proposition 4.1.1 and Theorem 4.1.1, we have the following corollary on the upper bounds of the macroscopic quantities.

Corollary 4.1.2. *Suppose that the conditions in Theorem 4.1.1 hold. We also assume the Knudsen number* $\varepsilon \ge \varepsilon_0 > 0$. *Then for all* $t > 0$, *the following bounds hold:*

$$\sup_{w} \sup_{\mathbf{x}} \{\rho(w;t,\mathbf{x}),\ |\boldsymbol{U}(w;t,\mathbf{x})|,\ T(w;t,\mathbf{x})\} \le C_1 \exp\left(\frac{C_2}{\varepsilon_0}t\right), \tag{4.1.22}$$

where C_1 and C_2 are positive constants depending only on A_0, A_1, C_0 and C_q.

Proof. By (4.1.16), (4.1.20) and (4.1.21), we have

$$(3T(w;t,\mathbf{x})+|\boldsymbol{U}(w;t,\mathbf{x})|^2)^{\frac{q-3}{2}}\leq\frac{C_qN_q(f)}{\rho(w;t,\mathbf{x})}\leq\frac{C_qA_0}{A_1}\exp\left(\frac{C_q+1}{\varepsilon}t\right). \tag{4.1.23}$$

Hence

$$T(w;t,\mathbf{x})\leq\frac{1}{3}\left(\frac{C_qA_0}{A_1}\right)^{\frac{2}{q-3}}\exp\left(\frac{2(C_q+1)}{(q-3)\varepsilon}t\right),\quad|\boldsymbol{U}(w,t,\mathbf{x})|\leq\left(\frac{C_qA_0}{A_1}\right)^{\frac{1}{q-3}}\exp\left(\frac{C_q+1}{(q-3)\varepsilon}t\right). \tag{4.1.24}$$

By (4.1.15) and (4.1.24), we have

$$\begin{aligned}\rho(w;t,\mathbf{x})&\leq C_0N_0(f)T(w;t,\mathbf{x})^{\frac{3}{2}}\\&\leq 3^{-\frac{3}{2}}C_0C_q^{\frac{3}{q-3}}A_0^{\frac{q}{q-3}}A_1^{-\frac{3}{q-3}}\exp\left(\frac{3(C_q+1)+(q-3)C_0}{(q-3)\varepsilon}t\right).\end{aligned} \tag{4.1.25}$$

□

4.2 Standard Monte Carlo method

In this section, we describe the basic Monte Carlo sampling method to solve the BGK equation (4.1.7) and establish some error estimates. For simplicity, we will consider that the uncertainty only comes from the initial condition. The case for the random boundary condition is similar.

4.2.1 Monte Carlo method

Suppose we generate M independent and identically distributed (i.i.d.) random samples f_0^{i}, $\mathrm{i}=1,\ldots,M$, according to the random initial condition $f_0(w;\mathbf{x},\mathbf{v})$. Then each $f_0^{\mathrm{i}}(w;\mathbf{x},\mathbf{v})$

will yield a unique analytical solution to (4.1.7) at time t, denoted by $f^{\mathrm{i}}(w;t,\mathbf{x},\mathbf{v})$. From $f^{\mathrm{i}}(w;t,\mathbf{x},\mathbf{v})$, we can easily compute

$$\begin{aligned}
&\rho^{\mathrm{i}}(w;t,\mathbf{x}) = \int_{\mathbb{R}^3} f^{\mathrm{i}}(w;t,\mathbf{x},\mathbf{v})\,\mathrm{d}\mathbf{v}, \quad \mathbf{m}^{\mathrm{i}}(w;t,\mathbf{x}) = \int_{\mathbb{R}^3} \mathbf{v} f^{\mathrm{i}}(w;t,\mathbf{x},\mathbf{v})\,\mathrm{d}\mathbf{v},\\
&E^{\mathrm{i}}(w;t,\mathbf{x}) = \int_{\mathbb{R}^3} \frac{|\mathbf{v}|^2}{2} f^{\mathrm{i}}(w;t,\mathbf{x},\mathbf{v})\,\mathrm{d}\mathbf{v},
\end{aligned} \tag{4.2.1}$$

then $\boldsymbol{U}^{\mathrm{i}}$ and T^{i} are given by

$$\boldsymbol{U}^{\mathrm{i}}(w;t,\mathbf{x}) = \frac{\mathbf{m}^{\mathrm{i}}(w;t,\mathbf{x})}{\rho^{\mathrm{i}}(w;t,\mathbf{x})}, \quad T^{\mathrm{i}}(w;t,\mathbf{x}) = \frac{2\rho^{\mathrm{i}}(w;t,\mathbf{x})E^{\mathrm{i}}(w;t,\mathbf{x}) - |\mathbf{m}^{\mathrm{i}}(w;t,\mathbf{x})|^2}{3(\rho^{\mathrm{i}}(w;t,\mathbf{x}))^2}. \tag{4.2.2}$$

Since it is the macroscopic quantities we are interested in, in the following, without further notice we will use a single variable q to denote ρ, $|\boldsymbol{U}|$ or T.

Given the samples q^{i}, $\mathrm{i} = 1,\ldots,M$, the MC estimate of the expectation $\mathbb{E}[q(w;t,\mathbf{x})]$ is given by

$$\mathbb{E}[q(w;t,\mathbf{x})] \approx E_M[q(w;t,\mathbf{x})] := \frac{1}{M}\sum_{\mathrm{i}=1}^{M} q^{\mathrm{i}}(w;t,\mathbf{x}). \tag{4.2.3}$$

To estimate the error between $\mathbb{E}[q(w;t,\mathbf{x})]$ and $E_M[q(w;t,\mathbf{x})]$, we need the following lemma.

Lemma 4.2.1. *For every finite sequence $\{Y_{\mathrm{j}}\}_{\mathrm{j}=1}^{M}$ of independent random variables with zero mean in $L^2(\Omega;L^2(\Omega_{\mathbf{x}}))$,*

$$\Big\|\sum_{\mathrm{j}=1}^{M} Y_{\mathrm{j}}\Big\|^2_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} = \sum_{\mathrm{j}=1}^{M} \|Y_{\mathrm{j}}\|^2_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))}. \tag{4.2.4}$$

Proof. From independence of $\{Y_{\mathrm{j}}\}_{\mathrm{j}=1}^{M}$ and that $\mathbb{E}[Y_{\mathrm{j}}] = 0$,

$$\begin{aligned}
\Big\|\sum_{\mathrm{j}=1}^{M} Y_{\mathrm{j}}\Big\|^2_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} &= \int_{\Omega_{\mathbf{x}}} \mathbb{E}[(\sum_{\mathrm{j}=1}^{M} Y_{\mathrm{j}})^2]\,\mathrm{d}\mathbf{x} = \int_D \mathbb{V}[\sum_{\mathrm{j}=1}^{M} Y_{\mathrm{j}}]\,\mathrm{d}\mathbf{x}\\
&= \int_D \sum_{\mathrm{j}=1}^{M} \mathbb{V}[Y_{\mathrm{j}}]\,\mathrm{d}\mathbf{x} = \sum_{\mathrm{j}=1}^{M}\int_D \mathbb{E}[Y_{\mathrm{j}}^2]\,\mathrm{d}\mathbf{x} = \sum_{\mathrm{j}=1}^{M}\|Y_{\mathrm{j}}\|^2_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))}.
\end{aligned} \tag{4.2.5}$$

□

We have the following consistency theorem.

Theorem 4.2.2. *For any $M \in \mathbb{N}^+$, at time $t = t_1$,*

$$\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E_M[q(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_\mathbf{x}))} \leq M^{-\frac{1}{2}}|\Omega_\mathbf{x}|^{\frac{1}{2}}\|\mathbb{V}[q(w;t_1,\mathbf{x})]\|^{\frac{1}{2}}_{L^1(\Omega_\mathbf{x})}. \tag{4.2.6}$$

Proof. We interpret the M samples $\{f_0^{\mathrm{i}}\}_{\mathrm{i}=1}^M$ as unique realizations of M independent samples of f_0 in the probability space $(\Omega, \mathscr{F}, \mathbb{P})$. In other words, $\{f_0^{\mathrm{i}}\}_{\mathrm{i}=1}^M$ are i.i.d. copies of $f_0 \in L^1(\Omega_\mathbf{x} \times \mathbb{R}^3)$. As a result, the corresponding copies of macroscopic quantities $\{q^{\mathrm{i}}(w;t_1,\mathbf{x})\}_{\mathrm{i}=1}^M$ derived from the initial data $\{f_0^{\mathrm{i}}\}_{\mathrm{i}=1}^M$ are also independent in $L^2(\Omega; L^1(\Omega_\mathbf{x}))$.

Denote $\mathbb{E}[q(w;t_1,\mathbf{x})] - q^{\mathrm{i}}(w;t_1,\mathbf{x})$ by $\Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})$, then

$$\mathbb{E}[\Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})] = 0, \tag{4.2.7}$$

and

$$\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E_M[q(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_\mathbf{x}))} = M^{-1}\Big\|\sum_{\mathrm{i}=1}^M \Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})\Big\|_{L^2(\Omega;L^1(\Omega_\mathbf{x}))}. \tag{4.2.8}$$

Using the boundedness of domain $\Omega_\mathbf{x}$,

$$\Big\|\sum_{\mathrm{i}=1}^M \Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})\Big\|^2_{L^1(\Omega_\mathbf{x})} \leq |\Omega_\mathbf{x}|\Big\|\sum_{\mathrm{i}=1}^M \Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})\Big\|^2_{L^2(\Omega_\mathbf{x})}. \tag{4.2.9}$$

Taking the expectation, noting that Δq^{i} are independent and using Lemma 4.2.1, we have

$$\begin{aligned}
&\Big\|\sum_{\mathrm{i}=1}^M \Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})\Big\|_{L^2(\Omega;L^1(\Omega_\mathbf{x}))} \leq |\Omega_\mathbf{x}|^{\frac{1}{2}}\Big\|\sum_{\mathrm{i}=1}^M \Delta q^{\mathrm{i}}(w,t,x_1)\Big\|_{L^2(\Omega;L^2(\Omega_\mathbf{x}))} \\
=&|\Omega_\mathbf{x}|^{\frac{1}{2}}\sqrt{\sum_{\mathrm{i}=1}^M \|\Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})\|^2_{L^2(\Omega;L^2(\Omega_\mathbf{x}))}} = |\Omega_\mathbf{x}|^{\frac{1}{2}}M^{\frac{1}{2}}\|\Delta q^{\mathrm{i}}(w,t_1,\mathbf{x})\|_{L^2(\Omega;L^2(\Omega_\mathbf{x}))} \\
=&|\Omega_\mathbf{x}|^{\frac{1}{2}}M^{\frac{1}{2}}\|\mathbb{V}[q(w;t_1,\mathbf{x})]\|^{\frac{1}{2}}_{L^1(\Omega_\mathbf{x})}.
\end{aligned} \tag{4.2.10}$$

□

As a direct result of Theorem 4.2.2 and Corollary 4.1.2, we have the following convergence theorem.

Theorem 4.2.3. *Under assumptions of Theorem 4.1.1 and Corollary 4.1.2, for* $0 < t_1 < \infty$, *as* $M \to \infty$, *the MC estimate* $E_M[q(w; t_1, \mathbf{x})]$ *converges in* $L^2(\Omega; L^1(\Omega_{\mathbf{x}}))$ *to* $\mathbb{E}[q(w; t_1, \mathbf{x})]$. *Furthermore, for any* $M \in \mathbb{N}^+$, *there holds the error bound*

$$\|\mathbb{E}[q(w; t_1, \mathbf{x})] - E_M[q(w; t_1, \mathbf{x})]\|_{L^2(\Omega; L^1(\Omega_{\mathbf{x}}))} \leq C_1 |\Omega_{\mathbf{x}}| \exp\left(\frac{C_2}{\varepsilon_0} t_1\right) M^{-\frac{1}{2}}. \tag{4.2.11}$$

Proof. It only needs to note that

$$\|\mathbb{V}[q(w; t_1, \mathbf{x})]\|^{\frac{1}{2}}_{L^1(\Omega_{\mathbf{x}})} \leq \|\mathbb{E}[q^2(w; t_1, \mathbf{x})]\|^{\frac{1}{2}}_{L^1(\Omega_{\mathbf{x}})} \leq |\Omega_{\mathbf{x}}|^{\frac{1}{2}} C_1 \exp\left(\frac{C_2}{\varepsilon_0} t_1\right). \tag{4.2.12}$$

□

4.2.2 Monte Carlo method with fully discrete scheme

To complete the error analysis, we need to consider the Monte Carlo method coupled with the fully discrete scheme for the BGK equation, which includes discretization in time, physical space and velocity space. The details are given in the 4.A. Simply speaking, we are using Gauss quadrature in the velocity space, second order IMEX-RK scheme for time discretization, and second order MUSCL finite volume scheme for spatial discretization (under the hyperbolic CFL condition $\Delta t \leq C\Delta x$). Overall, this leads to a second order positivity-preserving and asymptotic-preserving scheme for the deterministic BGK equation. In the following, we assume that the velocity discretization is accurate enough and ignore the work and error in velocity space. It is then reasonable to assume the numerical solution $q_{\Delta x, \Delta t}(w; t_1, \mathbf{x})$, computed with mesh size Δx and time step Δt corresponding to initial data $f_0(w; \mathbf{x}, \mathbf{v})$ up to time t_1, satisfies the following error estimate point-wise in w:

Assumption 4.2.1. *For fixed time* $t_1 > 0$, *under the hyperbolic CFL condition* $\Delta t \leq C\Delta x$, *we have*

$$\|q(w; t_1, \mathbf{x}) - q_{\Delta x, \Delta t}(w; t_1, \mathbf{x})\|_{L^1(\Omega_{\mathbf{x}})} \leq C(w)\left((\Delta x)^2 + (\Delta t)^2\right) \leq C_w (\Delta x)^2, \tag{4.2.13}$$

where C_w *is some positive constant.*

The MC estimate of the expectation $\mathbb{E}[q(w;t,\mathbf{x})]$ is now given by

$$\mathbb{E}[q(w;t,\mathbf{x})] \approx E_M[q_{\Delta x,\Delta t}(w;t,\mathbf{x})] := \frac{1}{M}\sum_{\mathrm{i}=1}^{M} q^{\mathrm{i}}_{\Delta x,\Delta t}(w;t,\mathbf{x}). \tag{4.2.14}$$

We have

Theorem 4.2.4. *For any $M \in \mathbb{N}^+$, at time $t = t_1$,*

$$\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E_M[q_{\Delta x,\Delta t}(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}\Omega_{\mathbf{x}}))} \leq M^{-\frac{1}{2}}|\Omega_{\mathbf{x}}|^{\frac{1}{2}}\|\mathbb{V}[q(w;t_1,\mathbf{x})]\|^{\frac{1}{2}}_{L^1(\Omega_{\mathbf{x}})} + C_w(\Delta x)^2. \tag{4.2.15}$$

Proof.

$$\begin{aligned}\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E_M[q_{\Delta x,\Delta t}(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} &\leq \|\mathbb{E}[q] - E_M[q]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} \\ &\quad + \|E_M[q] - E_M[q_{\Delta x,\Delta t}]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))}.\end{aligned} \tag{4.2.16}$$

It is enough to apply Theorem 4.2.2 and Assumption 4.2.1. □

The following corollary is a direct result of Theorem 4.2.4.

Corollary 4.2.5. *Under assumptions of Theorem 4.1.1 and Corollary 4.1.2, for $0 < t_1 < \infty$, as $M \to \infty$ and $\Delta x,\ \Delta t \to 0$, the MC estimate $E_M[q_{\Delta x,\Delta t}(w;t_1,\mathbf{x})]$ converges in $L^2(\Omega;L^1(\Omega_{\mathbf{x}}))$ to $\mathbb{E}[q(w;t_1,\mathbf{x})]$. Furthermore, for any $M \in \mathbb{N}^+$, there holds the error bound:*

$$\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E_M[q_{\Delta x,\Delta t}(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} \leq C_1|\Omega_{\mathbf{x}}|\exp\left(\frac{C_2}{\varepsilon_0}t_1\right)M^{-\frac{1}{2}} + C_w(\Delta x)^2. \tag{4.2.17}$$

4.3 Control variate multilevel Monte Carlo method

In this section we first introduce the multilevel Monte Carlo method and then following [71] we discuss the use of control variate techniques to optimize its variance reduction properties locally using two subsequent levels or globally among all levels.

4.3.1 Multilevel Monte Carlo method

The MLMC method is defined as a multilevel discretization in $\mathbf{x}$ and t with a level l dependent number of samples M_l. Suppose we have a nested triangulation $\{\mathcal{T}_l\}_{l=1}^{L}$ of the spatial domain $\Omega_{\mathbf{x}}$ ($L \in \mathbb{N}^+$ is the number of levels) such that the mesh size Δx_l at level l satisfies

$$\Delta x_l = \sup\{diam(K) : K \in \mathcal{T}_l\} \searrow \text{ as } l \nearrow . \tag{4.3.1}$$

Set $q^{\mathrm{i}}_{\Delta x_0, \Delta t_0}(w;t,\mathbf{x}) := 0$, then given a target level L of spatial resolution, the MLMC estimate of the expectation $\mathbb{E}[q(w;t,\mathbf{x})]$ is given as follows

$$\begin{aligned}
\mathbb{E}[q(w;t,\mathbf{x})] &\approx E^L[q_{\Delta x_L, \Delta t_L}(w;t,\mathbf{x})] \\
&:= \sum_{l=1}^{L} E_{M_l}\left[q_{\Delta x_l, \Delta t_l}(w;t,\mathbf{x}) - q_{\Delta x_{l-1}, \Delta t_{l-1}}(w;t,\mathbf{x})\right] \\
&= \sum_{l=1}^{L}\sum_{\mathrm{i}=1}^{M_l} \frac{1}{M_l}\left[q^{\mathrm{i}}_{\Delta x_l, \Delta t_l}(w;t,\mathbf{x}) - q^{\mathrm{i}}_{\Delta x_{l-1}, \Delta t_{l-1}}(w;t,\mathbf{x})\right].
\end{aligned} \tag{4.3.2}$$

Hence what we really sample is the difference of solutions at two consecutive levels. At each level l, we separately generate M_l i.i.d. samples f_0^{i}, $\mathrm{i} = 1, \ldots, M_l$, of the initial data f_0 on meshes Δx_l and Δx_{l-1} respectively, and then use the fully discrete scheme for the BGK equation (4.1.7) to advance solutions $q^{\mathrm{i}}_{\Delta x_l, \Delta t_l}$ and $q^{\mathrm{i}}_{\Delta x_{l-1}, \Delta t_{l-1}}$ to a certain time t.

To simplify the notation, we set $q_{\Delta x_0, \Delta t_0}(w;t,\mathbf{x}) := 0$ and define the random variable $Y_l := q_{\Delta x_l, \Delta t_l}(w;t,\mathbf{x}) - q_{\Delta x_{l-1}, \Delta t_{l-1}}(w;t,\mathbf{x})$, and the specific samples $Y_l^{\mathrm{i}} := q^{\mathrm{i}}_{\Delta x_l, \Delta t_l}(w;t,\mathbf{x}) - q^{\mathrm{i}}_{\Delta x_{l-1}, \Delta t_{l-1}}(w;t,\mathbf{x})$. We have the following consistency and convergence results for the estimator (4.3.2).

Theorem 4.3.1. *For any $M_l \in \mathbb{N}^+$, $l = 1, \ldots, L$, at time $t = t_1$,*

$$\begin{aligned}
\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E^L[q_{\Delta x_L, \Delta t_L}(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} &\le C_w(\Delta x_L)^2 \\
&\quad + |\Omega_{\mathbf{x}}|^{\frac{1}{2}} \sum_{l=1}^{L} M_l^{-\frac{1}{2}} \|\mathbb{V}[Y_l]\|^{\frac{1}{2}}_{L^1(\Omega_{\mathbf{x}})}.
\end{aligned} \tag{4.3.3}$$

Proof.

$$
\begin{aligned}
&\|\mathbb{E}[q] - E^L[q_{\Delta x_L,\Delta t_L}]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} = \|\mathbb{E}[q] - \sum_{l=1}^{L} E_{M_l}[Y_l]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} \\
&\leq \|\mathbb{E}[q] - \sum_{l=1}^{L} \mathbb{E}[Y_l]\|_{L^2(\Omega,L^1(\Omega_{\mathbf{x}}))} + \|\sum_{l=1}^{L} E_{M_l}[Y_l] - \sum_{l=1}^{L} \mathbb{E}[Y_l]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} \\
&\leq \|\mathbb{E}[q] - \mathbb{E}[q_{\Delta x_L,\Delta t_L}]\|_{L^1(\Omega_{\mathbf{x}})} + |\Omega_{\mathbf{x}}|^{\frac{1}{2}} \sum_{l=1}^{L} \|E_{M_l}[Y_l] - \mathbb{E}[Y_l]\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} \\
&= I + II.
\end{aligned}
\tag{4.3.4}
$$

For part I, Assumption 4.2.1 yields

$$
I = \|q(w;t_1,\mathbf{x}) - q_{\Delta x_L,\Delta t_L}(w;t_1,\mathbf{x})\|_{L^1(\Omega;L^1(\Omega_{\mathbf{x}}))} \leq C_w(\Delta x_L)^2. \tag{4.3.5}
$$

For part II, using Lemma 4.2.1,

$$
II = |\Omega_{\mathbf{x}}|^{\frac{1}{2}} \sum_{l=1}^{L} M_l{}^{-\frac{1}{2}} \|Y_l^{\mathrm{i}} - \mathbb{E}[Y_l]\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} = |\Omega_{\mathbf{x}}|^{\frac{1}{2}} \sum_{l=1}^{L} M_l{}^{-\frac{1}{2}} \|\mathbb{V}[Y_l]\|_{L^1(\Omega_{\mathbf{x}})}^{\frac{1}{2}}. \tag{4.3.6}
$$

□

Theorem 4.3.2. *Under the assumptions of Theorem 4.1.1 and Corollary 4.1.2, for* $0 < t_1 < \infty$, *as* $M_l \to \infty$ *and* $\Delta x,\ \Delta t \to 0$, *the MLMC estimate* $E^L[q_{\Delta x_L,\Delta t_L}(w;t_1,\mathbf{x})]$ *converges in* $L^2(\Omega;L^1(\Omega_{\mathbf{x}}))$ *to* $\mathbb{E}[q(w;t_1,\mathbf{x})]$. *Furthermore, there holds the error bound:*

$$
\begin{aligned}
&\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E^L[q_{\Delta x_L,\Delta t_L}(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} \\
&\leq C_w(\Delta x_L)^2 + \left(C_w|\Omega_{\mathbf{x}}|^{\frac{1}{2}}(\Delta x_1)^2 + C_1|\Omega_{\mathbf{x}}|\exp\left(\frac{C_2}{\varepsilon_0}t_1\right)\right) M_1^{-\frac{1}{2}} \\
&+ \sum_{l=2}^{L} C_w|\Omega_{\mathbf{x}}|^{\frac{1}{2}} \left((\Delta x_l)^2 + (\Delta x_{l-1})^2\right) M_l^{-\frac{1}{2}}.
\end{aligned}
\tag{4.3.7}
$$

Proof. From (4.1.4), we can see that $\mathbb{V}[X] \leq \mathbb{E}[X^2]$, then from Theorem 4.3.1 for $l = 1$,

$$\begin{aligned}
\|Y_1^{\mathrm{i}} - \mathbb{E}[Y_1]\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} &= \|q^{\mathrm{i}}_{\Delta x_1,\Delta t_1} - \mathbb{E}[q^{\mathrm{i}}_{\Delta x_1,\Delta t_1}]\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} \\
&\leq \|q^{\mathrm{i}}_{\Delta x_1,\Delta t_1}\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} \\
&\leq \|q^{\mathrm{i}}_{\Delta x_1,\Delta t_1} - q^{\mathrm{i}}\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} + \|q^{\mathrm{i}}\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} \\
&\leq C_w(\Delta x_1)^2 + |\Omega_{\mathbf{x}}|^{\frac{1}{2}} C_1 \exp\left(\frac{C_2}{\varepsilon_0} t_1\right),
\end{aligned} \tag{4.3.8}$$

and similarly for $l \geq 2$,

$$\begin{aligned}
\|Y_l^{\mathrm{i}} - \mathbb{E}[Y_l]\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} &\leq \|Y_l^{\mathrm{i}}\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} \\
&= \|q^{\mathrm{i}}_{\Delta x_l,\Delta t_l} - q^{\mathrm{i}}_{\Delta x_{l-1},\Delta t_{l-1}}\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} \\
&\leq \|q^{\mathrm{i}}_{\Delta x_l,\Delta t_l} - q^{\mathrm{i}}\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} + \|q^{\mathrm{i}} - q^{\mathrm{i}}_{\Delta x_{l-1},\Delta t_{l-1}}\|_{L^2(\Omega;L^2(\Omega_{\mathbf{x}}))} \\
&\leq C_w((\Delta x_l)^2 + (\Delta x_{l-1})^2).
\end{aligned} \tag{4.3.9}$$

□

Remark 4.3.3. *The summation term on the right hand side of (4.3.7) implies that, if the mesh is refined by a factor of 2 as the level increases, then, to balance the errors in different levels, the sample ratio across levels should be chosen as* $2^4 = 16$. *Furthermore, it should be noted that the above error estimate highly depends on the regularity of the solution and is valid when the solution is smooth (typical when the BGK equation is in the kinetic regime). When the solution contains discontinuities/shocks (typical when the BGK equation is close to the fluid regime), it is well-known that the numerical scheme will not maintain its original order. A second order scheme as we considered here will generally degenerate to first order or even worse [72]. Therefore, to balance the errors in different levels, the sample ratio can be chosen smaller. Our numerical results in Section 4.4.2 (smooth solutions) and Sections 4.4.3-4.4.4 (discontinuous solutions) indeed confirmed this prediction (the general trend follows though the actual ratio value chosen may not be exactly the above predicted number due to the non-negligible first two terms on the right hand side of (4.3.7)).*

4.3.2 Quasi-optimal and optimal multilevel Monte Carlo method

In this section we generalize the previous MLMC method following [71]. To start with, take the 2 level MLMC method for example. Suppose we have a low fidelity (coarse mesh) approximation q_1 and a high fidelity (fine mesh) approximation q_2, then the 2 level MLMC method with control variate reads as follows

$$\mathbb{E}[q] \approx E_{M_1}[\lambda q_1] + E_{M_2}[q_2 - \lambda q_1], \tag{4.3.10}$$

where the multiplier λ has to be determined in order to minimize the overall variance $\mathbb{V}[q] = \lambda^2\mathbb{V}[q_1] + \mathbb{V}[q_2 - \lambda q_1]$. It can be shown that for independent samples the optimal value of λ is given by

$$\lambda = \frac{\mathrm{Cov}[q_1, q_2]}{2\mathbb{V}[q_1]}. \tag{4.3.11}$$

When $M_2 \ll M_1$, the contribution from $\mathbb{V}[\lambda q_1]$ is negligible compared to $\mathbb{V}[q_2 - \lambda q_1]$. We therefore can only focus on the minimization of the variance $\mathbb{V}[q_1 - \lambda q_2]$. In this case, the optimal value of λ is given by

$$\lambda = \frac{\mathrm{Cov}[q_1, q_2]}{\mathbb{V}[q_1]} \approx \frac{\sum_{\mathrm{i}=1}^{M_2}(q_1^{\mathrm{i}} - \bar{q}_1)(q_2^{\mathrm{i}} - \bar{q}_2)}{\sum_{\mathrm{i}=1}^{M_2}(q_1^{\mathrm{i}} - \bar{q}_1)^2}, \tag{4.3.12}$$

where $\bar{q}_1 = E_{M_2}[q_1]$, $\bar{q}_2 = E_{M_2}[q_2]$ and in the above expression the covariance and variance are estimated directly from the Monte Carlo samples.

Generally, suppose we have L levels of solutions $\{q_{\Delta x_{\mathrm{i}}, \Delta t_{\mathrm{i}}}\}_{\mathrm{i}=1,\dots,L}$, from coarsest level $q_{\Delta x_1, \Delta t_1}$ to finest level $q_{\Delta x_L, \Delta t_L}$. Then the MLMC method with control variates is given by

$$\begin{aligned}\mathbb{E}[q(w;t,\mathbf{x})] &\approx E^L_{CV}[q_{\Delta x_L, \Delta t_L}] \\ &:= \prod_{\mathrm{i}=1}^{L} \lambda_{\mathrm{i}} E_{M_1}[q_{\Delta x_1, \Delta t_1}] + \sum_{l=2}^{L}\prod_{\mathrm{i}=l}^{L} \lambda_{\mathrm{i}} E_{M_l}[q_{\Delta x_l, \Delta t_l} - \lambda_{l-1} q_{\Delta x_{l-1}, \Delta t_{l-1}}].\end{aligned} \tag{4.3.13}$$

Note that $\{\lambda_l\}_{l=1}^{L}$ here are the coefficients to be determined and $\lambda_L = 1$. If we only consider the variance reduction for each pair of consecutive levels, then we can easily get the analogy of (4.3.11) to estimate $\{\lambda_l\}$, which we refer to as the *quasi-optimal MLMC method*:

$$\lambda_{l-1} = \frac{\mathrm{Cov}[q_{\Delta x_l,\Delta t_l}, q_{\Delta x_{l-1},\Delta t_{l-1}}]}{\mathbb{V}[q_{\Delta x_{l-1},\Delta t_{l-1}}]} \approx \frac{\sum_{\mathrm{i}=1}^{M_l}(q^{\mathrm{i}}_{\Delta x_l,\Delta t_l} - \bar{q}_{\Delta x_l,\Delta t_l})(q^{\mathrm{i}}_{\Delta x_{l-1},\Delta t_{l-1}} - \bar{q}_{\Delta x_{l-1},\Delta t_{l-1}})}{\sum_{\mathrm{i}=1}^{M_l}(q^{\mathrm{i}}_{\Delta x_{l-1},\Delta t_{l-1}} - \bar{q}_{\Delta x_{l-1},\Delta t_{l-1}})^2}, \tag{4.3.14}$$

where $\bar{q}_{\Delta x_l,\Delta t_l} = E_{M_l}[q_{\Delta x_l,\Delta t_l}]$.

However, if we focus on minimizing the overall variance of the estimator 4.3.13 and assume that the levels are independent, then denoting

$$\hat{\lambda}_l = \prod_{\mathrm{i}=l}^{L} \lambda_{\mathrm{i}}, \quad l = 1, \ldots, L, \tag{4.3.15}$$

the optimality conditions yield a tridiagonal system for $\hat{\lambda}_l$:

$$\begin{aligned} &\hat{\lambda}_l \mathbb{V}[q_{\Delta x_l,\Delta t_l}] - \hat{\lambda}_{l+1}\frac{M_l}{M_l + M_{l+1}}\mathrm{Cov}[q_{\Delta x_{l+1},\Delta t_{l+1}}, q_{\Delta x_l,\Delta t_l}] \\ &- \hat{\lambda}_{l-1}\frac{M_{l+1}}{M_l + M_{l+1}}\mathrm{Cov}[q_{\Delta x_{l-1},\Delta t_{l-1}}, q_{\Delta x_l,\Delta t_l}] = 0, \quad l = 1, \ldots, L-1, \end{aligned} \tag{4.3.16}$$

where we assumed $\hat{\lambda}_0 = 0$, $\hat{\lambda}_L = 1$ and $q_{\Delta x_0,\Delta t_0} = 0$. A practical way to solve the above tridiagonal system is to rewrite (4.3.16) in terms of original λ_{i}. For simplicity, we denote $\mathbb{V}[q_{\Delta x_l,\Delta t_l}]$ by $\mathbb{V}_l$ and $\mathrm{Cov}[q_{\Delta x_{l+1},\Delta t_{l+1}}, q_{\Delta x_l,\Delta t_l}]$ by Cov_l to get

$$\begin{aligned} &\lambda_1\mathbb{V}_1 - \frac{M_1}{M_1 + M_2}\mathrm{Cov}_1 = 0, \\ &\lambda_2\mathbb{V}_2 - \frac{M_2}{M_2 + M_3}\mathrm{Cov}_2 - \lambda_1\lambda_2\frac{M_3}{M_2 + M_3}\mathrm{Cov}_1 = 0, \\ &\lambda_3\mathbb{V}_3 - \frac{M_3}{M_3 + M_4}\mathrm{Cov}_3 - \lambda_2\lambda_3\frac{M_4}{M_3 + M_4}\mathrm{Cov}_2 = 0, \\ &\ldots \\ &\lambda_{L-1}\mathbb{V}_{L-1} - \frac{M_{L-1}}{M_{L-1} + M_L}\mathrm{Cov}_{L-2} - \lambda_{L-2}\lambda_{L-1}\frac{M_L}{M_{L-1} + M_L}\mathrm{Cov}_{L-2} = 0, \end{aligned} \tag{4.3.17}$$

which can be easily solved by recursive substitution. This is what we refer to as the *optimal MLMC method*.

Denote the correlation coefficient of $q_{\Delta x_l,\Delta t_l}$ and $q_{\Delta x_{l+1},\Delta t_{l+1}}$ by

$$r_l = \frac{\mathrm{Cov}[q_{\Delta x_l,\Delta t_l}, q_{\Delta x_{l+1},\Delta t_{l+1}}]}{\left(\mathbb{V}[q_{\Delta x_{l+1},\Delta t_{l+1}}]\mathbb{V}[q_{\Delta x_l,\Delta t_l}]\right)^{\frac{1}{2}}}, \tag{4.3.18}$$

we can prove the following consistency and convergence results for the estimator (4.3.13):

Theorem 4.3.4. *For any $M_l \in \mathbb{N}^+$, $l = 1, \ldots, L$, if $\{\lambda_l\}$ are quasi-optimal and exact, i.e.,*

$$\lambda_l = \frac{Cov[q_{\Delta x_l,\Delta t_l}, q_{\Delta x_{l+1},\Delta t_{l+1}}]}{\mathbb{V}[q_{\Delta x_l,\Delta t_l}]}, \tag{4.3.19}$$

then at time $t = t_1$,

$$\begin{aligned} &\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E^L_{CV}[q_{\Delta x_L,\Delta t_L}(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_\mathbf{x}))} \\ &\leq C_w(\Delta x_L)^2 + |\Omega_\mathbf{x}|^{\frac{1}{2}} {M_1}^{-\frac{1}{2}} \hat{\lambda}_1 \|\mathbb{V}[q_{\Delta x_1,\Delta t_1}]\|^{\frac{1}{2}}_{L^1(\Omega_\mathbf{x})} \\ &+ |\Omega_\mathbf{x}|^{\frac{1}{2}} \sum_{l=2}^{L} {M_l}^{-\frac{1}{2}} \hat{\lambda}_l (1 - r_{l-1}^2)^{\frac{1}{2}} \|\mathbb{V}[q_{\Delta x_l,\Delta t_l}]\|^{\frac{1}{2}}_{L^1(\Omega_\mathbf{x})}. \end{aligned} \tag{4.3.20}$$

Proof. The proof is similar to Theorem 4.3.1. All we need is to note that when λ is quasi-optimal, we have for $l \geq 2$,

$$\begin{aligned} \mathbb{V}[q_{\Delta x_l,\Delta t_l} - \lambda_{l-1} q_{\Delta x_{l-1},\Delta t_{l-1}}] &= \mathbb{V}[q_{\Delta x_l,\Delta t_l}] + \lambda_{l-1}^2 \mathbb{V}[q_{\Delta x_{l-1},\Delta t_{l-1}}] \\ &\quad - 2\lambda_{l-1}\mathrm{Cov}[q_{\Delta x_l,\Delta t_l}, q_{\Delta x_{l-1},\Delta t_{l-1}}] \\ &= (1 - r_{l-1}^2)\mathbb{V}[q_{\Delta x_l,\Delta t_l}]. \end{aligned} \tag{4.3.21}$$

□

Theorem 4.3.5. *Under the assumptions of Theorem 4.1.1 and Corollary 4.1.2, and if $\{\lambda_l\}$ are quasi-optimal and exact, we have for $0 < t_1 < \infty$, as $M_l \to \infty$ and Δx, $\Delta t \to 0$, the quasi-*

optimal MLMC estimate $E^L_{CV}[q_{\Delta x_L,\Delta t_L}(w;t_1,\mathbf{x})]$ *converges in* $L^2(\Omega;L^1(\Omega_{\mathbf{x}}))$ *to* $\mathbb{E}[q(w;t_1,\mathbf{x})]$ *with the error bound*

$$\begin{aligned}
&\|\mathbb{E}[q(w;t_1,\mathbf{x})] - E^L_{CV}[q_{\Delta x_L,\Delta t_L}(w;t_1,\mathbf{x})]\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}}))} \\
&\leq C_w(\Delta x_L)^2 + \sum_{l=2}^{L} C_w|\Omega_{\mathbf{x}}|^{\frac{1}{2}}\hat{\lambda}_l M_l^{-\frac{1}{2}}(1-r_{l-1}^2)^{\frac{1}{2}}(\Delta x_l)^2 \\
&+ \left(C_w|\Omega_{\mathbf{x}}|^{\frac{1}{2}}(\Delta x_1)^2 + C_1|\Omega_{\mathbf{x}}|\exp\left(\frac{C_2}{\varepsilon_0}t_1\right)\right) M_1^{-\frac{1}{2}}\hat{\lambda}_1.
\end{aligned} \tag{4.3.22}$$

Remark 4.3.6. *Note that the computational cost for quasi-optimal and optimal MLMC is the same as the standard MLMC method. One can use the data from MLMC to estimate* λ_l *using* (4.3.14) *or* (4.3.17)*. Finally, we emphasize that in [73] one of the estimators, W-RDiff estimator, in fact coincides with our optimal MLMC strategy (see also [71]). However, the method has never been analyzed in the case of kinetic equations and additionally, the quasi-MLMC method does not appear in the previous literature. Without solving a tridiagonal system which may suffer from ill-conditioning, the quasi-MLMC method offers an efficient and robust alternative to the optimal MLMC method.*

4.4 Numerical results

In this section, we present several numerical examples for the BGK equation (4.1.7) with random initial condition or random boundary condition. The details of the deterministic solver are provided in Section 4.A. Simply speaking, we are solving a reduced system (4.A.6) and (4.A.7), which is equivalent to the full BGK equation in one spatial dimension. We use the IMEX-RK scheme for time discretization and finite volume scheme for spatial discretization so that the overall method is second order in both time and space. We choose $x \in [0,1]$ and $v \in [-5,5]$, where 40 Legendre-Gauss quadrature points are used in the velocity space to ensure that the error in velocity is negligible. The CFL condition is fixed as $\Delta t = 0.1\Delta x$.

4.4.1 Error evaluation

In the following, we assume the uncertainties come from either the initial condition or boundary condition. Since the solution is a random field, the numerical error is a random quantity as well. For error analysis, we therefore compute a statistical estimator by averaging numerical errors from several independent experiments.

More precisely, for each method we perform $K = 40$ experiments, and get the corresponding approximations $\{q^{(\mathrm{j})}(t,x)\}$, $\mathrm{j} = 1, \ldots, K$, where q can be ρ, U or T. We approximate the overall error in norm $\|\cdot\|_{L^2(\Omega;L^1(\Omega_{\mathbf{x}})))}$ via

$$E(t) = \sqrt{\frac{1}{K}\sum_{\mathrm{j}=1}^{K}\|q^{(\mathrm{j})}(t,\cdot) - q_{\mathrm{ref}}(t,\cdot)\|^2_{L^1(\Omega_{\mathbf{x}})}}, \tag{4.4.1}$$

where $q_{\mathrm{ref}}(t,x)$ is the reference solution obtained using the stochastic collocation method [74] with 120 Legendre-Gauss collocation points and $N_x = 1280$ spatial points. We are also interested in the error at each spatial point:

$$E_{\Delta x}(t,x) = \sqrt{\frac{1}{K}\sum_{\mathrm{j}=1}^{K}(q^{(\mathrm{j})}(t,x) - q_{\mathrm{ref}}(t,x))^2}. \tag{4.4.2}$$

Sometimes to better evaluate the error from the random domain, we would like to ignore the error induced by spatial discretization. To achieve so, we consider another kind of reference solution, $q_{\mathrm{rel}}(t,x)$, obtained again using the stochastic collocation with 120 collocation points, while in the spatial domain we use the same finest mesh Δx_L as that in the corresponding MLMC method to obtain $q^{(\mathrm{j})}(t,x)$. Therefore, we can assess the error as

$$E_{\mathrm{rel}\Delta x}(t,x) = \sqrt{\frac{1}{K}\sum_{\mathrm{j}=1}^{K}(q^{(\mathrm{j})}(t,x) - q_{\mathrm{rel}}(t,x))^2}. \tag{4.4.3}$$

In each of the following tests, we perform two stages of computations. The experimental stage is to determine the optimal sample allocation parameters (there are some guidance from the theoretical estimates, see Remark 4.3.3, but we still choose to do a careful testing

just as a way to verify the theory). The simulation stage is to perform various methods to estimate the physical quantities of interest.

4.4.2 Test 1: Smooth random initial condition

We first consider the BGK equation subject to random initial condition:

$$f^0(\mathbf{x}, \mathbf{v}, z) = 0.5\mathcal{M}[\rho, \boldsymbol{U}, T] + 0.5\mathcal{M}[\rho, -\boldsymbol{U}, T], \tag{4.4.4}$$

with

$$\mathcal{M}[\rho, \boldsymbol{U}, T](\mathbf{x}, \mathbf{v}, z) = \frac{\rho(\mathbf{x}, z)}{(2\pi T(\mathbf{x}, z))^{\frac{3}{2}}} \exp\left(-\frac{|\mathbf{v} - \boldsymbol{U}(\mathbf{x}, z)|^2}{2T(\mathbf{x}, z)}\right), \tag{4.4.5}$$

where

$$\begin{aligned} \rho(\mathbf{x}, z) &= \frac{2 + \sin(2\pi x) + \frac{1}{2}\sin(4\pi x)z}{3}, \quad \boldsymbol{U}(\mathbf{x}) = (0.2, 0, 0), \\ T(\mathbf{x}, z) &= \frac{3 + \cos(2\pi x) + \frac{1}{2}\cos(4\pi x)z}{4}, \end{aligned} \tag{4.4.6}$$

and the random variable z obeys the uniform distribution on $[-1, 1]$. The periodic boundary condition is used and the Knudsen number $\varepsilon = 1$.

To determine the number of samples needed in MC and MLMC methods as well as the sample ratio across levels in MLMC methods, we proceed as follows.

In the MC method, we consider a series of spatial discretizations: $N = 10$, 20, 30, 40, and for each case, we vary the sample size as $M = 5$, 10, 15, ... The results are shown in Figure 4.1 (left), where we plot the error (4.4.1). It can be observed that when the number of samples is few, the statistical error dominates and when there are enough number of samples, the spatial error dominates. Therefore, we can roughly determine the minimum number of samples needed so that the statistical error $O(M^{-\frac{1}{2}})$ balances with the spatial/temporal error $O(\Delta x^2)$:

- $N = 10$, $M \approx 40$.
- $N = 20$, $M \approx 640$.
- $N = 30$, $M \approx 3300$.

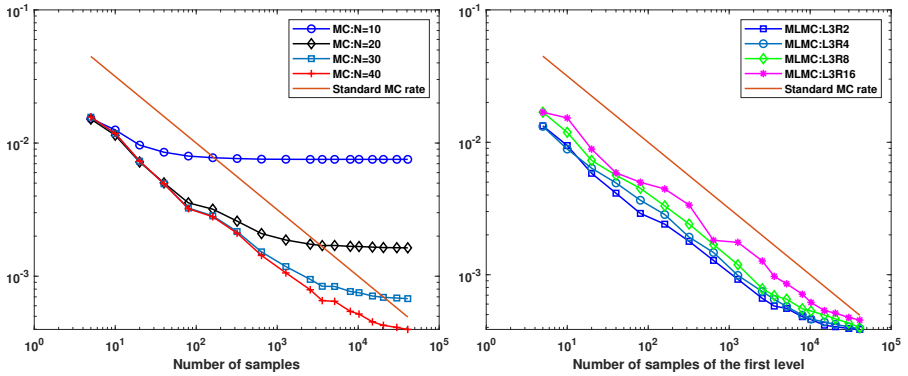

Figure 4.1. Test 1: Error (4.4.1) (density ρ) of MC method (left) and MLMC method (right) v.s. number of samples (for MLMC, it is the number of samples in the first level).

- $N = 40$, $M \approx 10240$.

In the MLMC method, we consider three levels of spatial discretizations: $N_1 = 10$, $N_2 = 20$, $N_3 = 40$ and the corresponding number of samples at each level are chosen as M_1, $M_2 = \frac{M_1}{a}$ and $M_3 = \frac{M_1}{a^2}$, where we test different ratios $a = 2, 4, 8, 16$. We then vary the starting sample size as $M_1 = 16$, 32, 48, ... The results are shown in Figure 4.1 (right), where we can see that regardless of ratios, the statistical error and spatial/temporal error are roughly balanced when $M_1 \approx 10240$ (the error saturates when the sample size further increases).

In Figure 4.2 we combine all the previous MC and MLMC results under the scale of workload. Since we are essentially solving 1D BGK problem, the workload for one deterministic run up to certain time with N spatial points is $O(N^2)$. Then for the MC method with M samples, the total work is $O(MN^2)$. For the MLMC method with ratio a, the amount of work is $O(M_1N_1^2 + M_2(N_1^2 + N_2^2) + M_3(N_2^2 + N_3^2) = \frac{a^2+5a+20}{a^2}M_1N_1^2)$. As we can see clearly from Figure 4.2, with the same workload, the MLMC methods can achieve better accuracy compared to various MC. Among MLMC methods with different ratios, there is no significant difference except for ratio $a = 2$. Therefore, we empirically set $a = 4$ for this smooth random initial condition test.

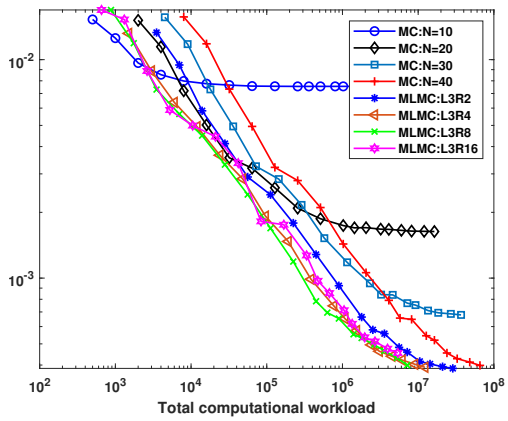

Figure 4.2. Test 1: Error (4.4.1) (density ρ) of MC and MLMC methods v.s. computational workload.

Now we fix the mesh sizes $N_1 = 10$, $N_2 = 20$, $N_3 = 40$, and sample sizes $M_1 = 10240$, $M_2 = 2560$, $M_3 = 640$ in the MLMC method. We then find the number of samples in the MC method such that they have the same workload. This means

- $N = 10$, $M = 30720$.
- $N = 20$, $M = 7680$.
- $N = 30$, $M = 3413$.
- $N = 40$, $M = 1920$.

Note that comparing with the numbers we found earlier, for $N = 10$ and 20, the number of samples are far beyond the minimum number of samples needed, while for $N = 30$, M is around the minimum number of samples needed. Finally for $N = 40$, the number of samples here is not enough to balance the statistical error and numerical error in the MC method. Using the above parameters, we compare the errors of the standard MC method and three MLMC methods, namely, the standard MLMC, the quasi-optimal MLMC, and optimal MLMC. The results are shown in Figure 4.3, from which we clearly see the better accuracy of MLMC methods compared to standard MC for fixed workload. On the other hand, the difference of three MLMC methods are not obvious in this example.

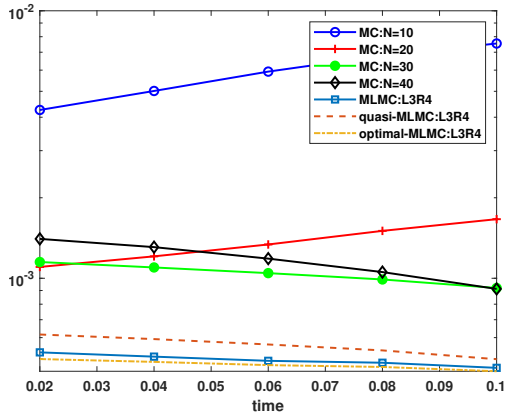

Figure 4.3. Test 1: Time evolution of the errors (4.4.1) (density ρ) using MC and various MLMC methods.

Next we examine the errors of the three MLMC methods as defined in (4.4.2), (4.4.3). The results are gathered in Figure 4.4. We can see that the three MLMC methods perform equally well in this test (the differences of the three methods are not significant though the optimal MLMC has the smallest error overall), largely because the solution is smooth.

To better understand this, we plot the values of λ_1 and λ_2 in the quasi-optimal and optimal MLMC methods in Figure 4.5. We can see that almost all values are not far from 1, which means the methods are not far from the standard MLMC.

4.4.3 Test 2: Shock tube problem

In this test, we consider two kinds of shock tube problems with random initial condition. The first one has uncertainty in the interface location:

$$I: \begin{cases} \rho_l = 1, & \boldsymbol{U}_l = (0,0,0), \quad T_l = 1, & f_0 = \mathcal{M}[\rho_l, \boldsymbol{U}_l, T_l] & x \le 0.5 + 0.05z, \\ \rho_r = 0.125, & \boldsymbol{U}_r = (0,0,0), \quad T_r = 0.25, & f_0 = \mathcal{M}[\rho_r, \boldsymbol{U}_r, T_r] & x > 0.5 + 0.05z. \end{cases} \tag{4.4.7}$$

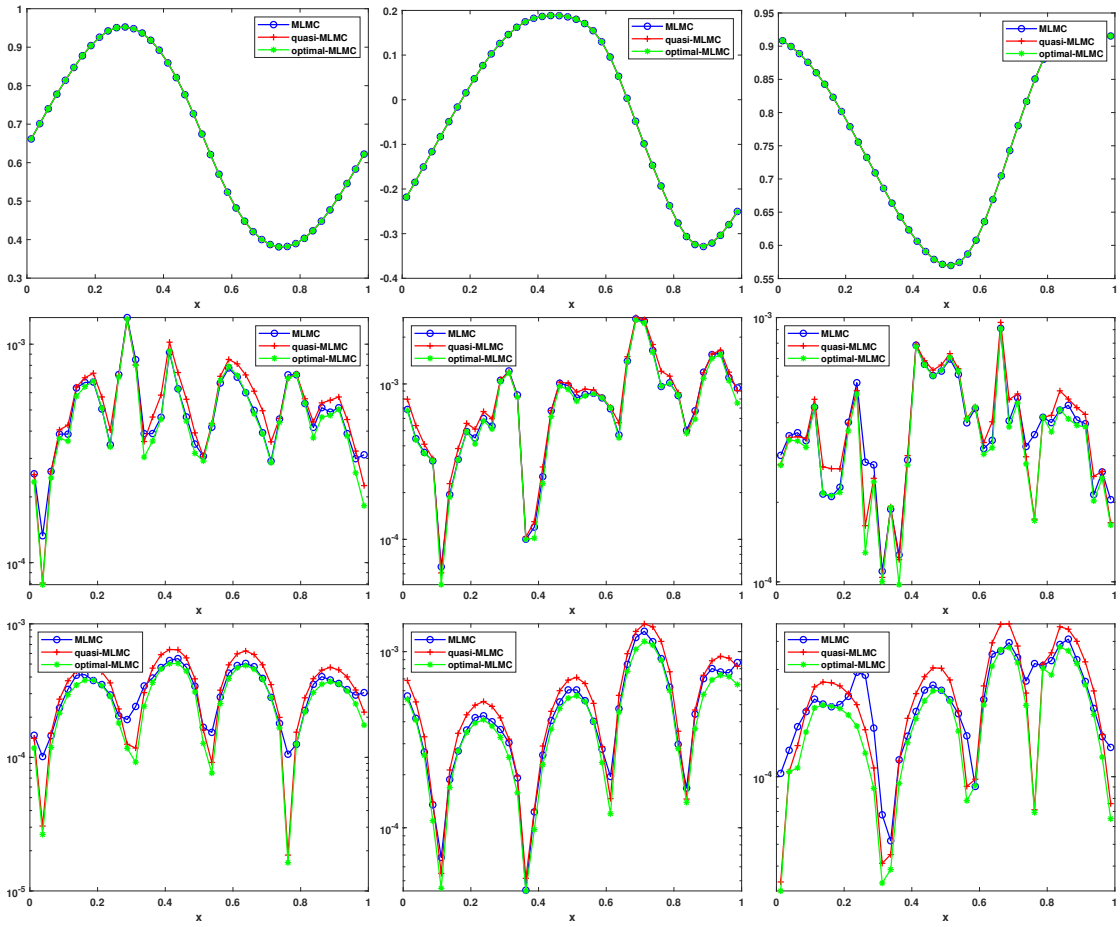

Figure 4.4. Test 1: Approximated expectation of density $\mathbb{E}[\rho]$ (left), velocity $\mathbb{E}[U]$ (middle) and temperature $\mathbb{E}[T]$ (right) using MLMC, quasi-optimal MLMC and optimal MLMC methods at time $t = 0.1$ (top row). Error (4.4.2) of expectation of density (left), velocity (middle) and temperature (right) using three MLMC methods (middle row). Relative error (4.4.3) of expectation of density (left), velocity (middle) and temperature (right) using three MLMC methods (bottom row).

The second one has uncertainty in the state variables:

$$II: \begin{cases} \rho_l = 1 + 0.1(z+1), & \boldsymbol{U}_l = (0,0,0), & T_l = 1, & f_0 = \mathcal{M}[\rho_l, \boldsymbol{U}_l, T_l] & x \le 0.5, \\ \rho_r = 0.125, & \boldsymbol{U}_r = (0,0,0), & T_r = 0.25, & f_0 = \mathcal{M}[\rho_r, \boldsymbol{U}_r, T_r] & x > 0.5. \end{cases} \tag{4.4.8}$$

The random variable z obeys the uniform distribution on $[-1, 1]$. We set the Knudsen number $\varepsilon = 10^{-6}$ so that the problem is close to the fluid regime.

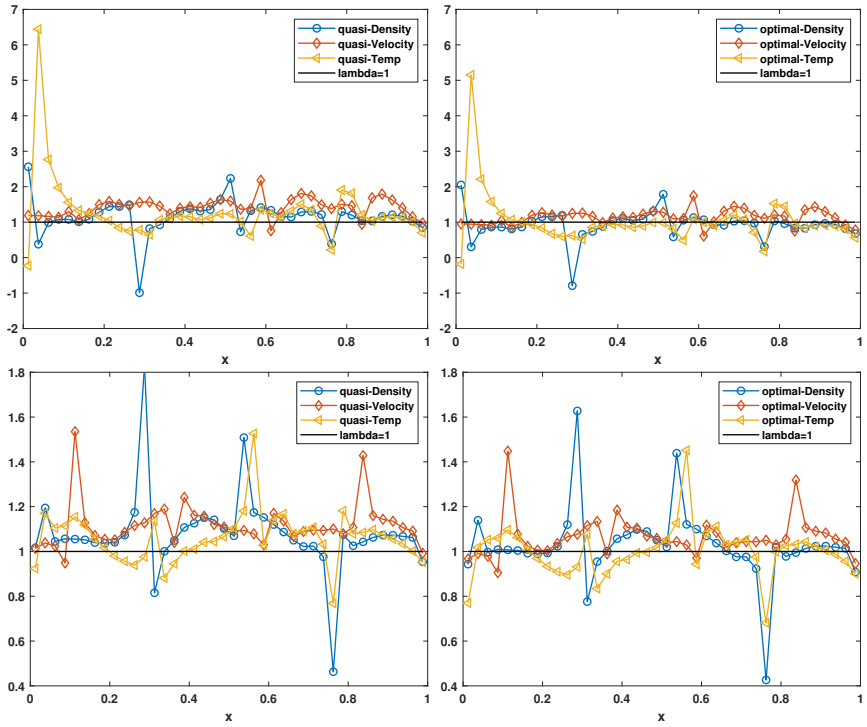

Figure 4.5. Test 1: Values of λ_1 in quasi-optimal (left) and optimal (right) MLMC methods (top row). Values of λ_2 in quasi-optimal (left) and optimal (right) MLMC methods (bottom row).

For problem (I), similarly as the previous example, we perform a series of tests to determine the optimal number of samples needed in MC and MLMC methods as well as the sample ratio across levels. Figure 4.6 shows the analogous tests as those in Figure 4.1. The main difference from the previous example is that the errors saturate much quicker as the number of samples increases. This is due to the low regularity of the solution so that the error from spatial/temporal discretization dominants easily. In Figure 4.7 we combine both MC and MLMC results under the scale of workload. Similarly as what we observed in Figure 4.2, with the same workload, the MLMC methods can achieve better accuracy compared to MC. In addition, the MLMC methods with ratios $a = 2, 4$ are more accurate than $a = 8, 16$. This is consistent to our earlier theoretical prediction, see Remark 4.3.3. From the right plot in Figure 4.6, we also see that $M_1 \approx 320$ is the minimum number of samples needed for the MLMC method to balance the statistical error and spatial/temporal error. Therefore, we choose the following parameters in the MLMC methods: mesh sizes $N_1 = 10$, $N_2 = 20$,

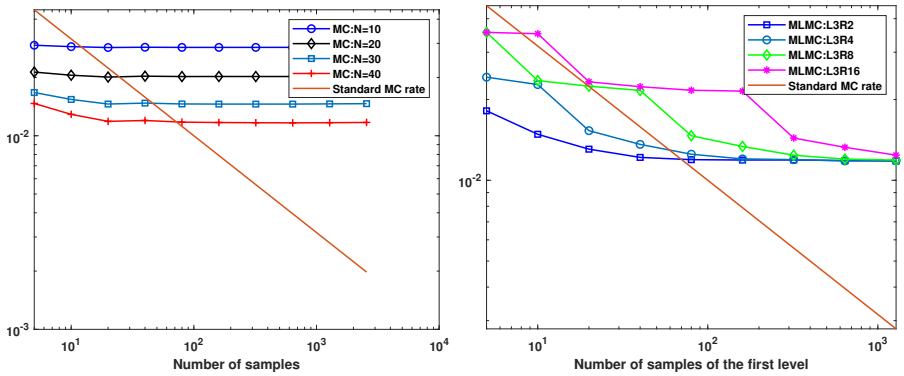

Figure 4.6. Test 2 (I): Error (4.4.1) (density ρ) of MC method (left) and MLMC method (right) v.s. number of samples (for MLMC, it is the number of samples in the first level).

$N_3 = 40$, and sample sizes $M_1 = 320$, $M_2 = 80$, $M_3 = 20$. In Figures 4.8-4.9, we report the results obtained using the standard MLMC, quasi-optimal MLMC, and optimal MLMC methods. We mainly examine the approximation to the expectation $\mathbb{E}[q]$ as the proposed quasi-MLMC and optimal MLMC methods are especially designed to minimize the variance in the estimation of $\mathbb{E}[q]$. As a by-product, we also plot the approximation to the variance $\mathbb{V}[q]$ using the samples generated for expectation. Note that the MLMC methods are based on the linearity of the expectation operator, not the variance operator. Hence to approximate the variance, we approximate separately two different expectations $\mathbb{E}[q^2]$ and $\mathbb{E}[q]$ and use them to obtain $\mathbb{V}[q] = \mathbb{E}[q^2] - (\mathbb{E}[q])^2$. We refer to [75] for other approaches to variance approximation including error control. The results clearly show that both control variate MLMC methods outperform the standard MLMC in regions where the solution presents strong variations, namely close to the shock position. Although the results are very close, as expected, the optimal MLMC method performs slightly better than the quasi-optimal MLMC.

For problem (II), we choose the following parameters: mesh sizes $N_1 = 10, N_2 = 20, N_3 = 40$, and number of samples $M_1 = 640, M_2 = 160, M_3 = 40$ (these parameters are chosen based on a similar test as problem (I) and we omit the detail). The results are shown in Figure 4.10 and Figure 4.11, where the same observation as problem (I) is obtained.

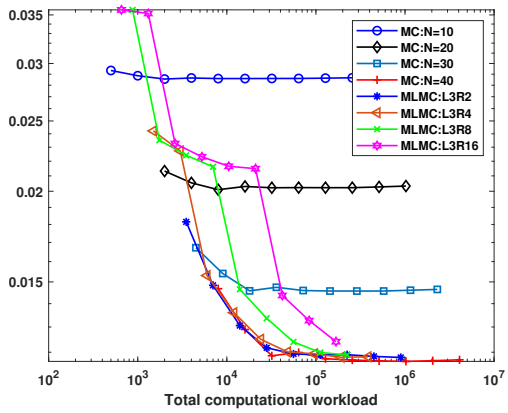

Figure 4.7. Test 2 (I): Error (4.4.1) (density ρ) of MC and MLMC methods v.s. computational workload.

To better see the difference of the three MLMC methods, we plot the values of λ_1 and λ_2 in the quasi-optimal and optimal MLMC methods for both problems (I) and (II) in (4.12) and (4.13). It is clear that for these problems with shocks/discontinuities the values are far from one in various regions of the computational domain. This is particularly true for the temperature and velocity in agreement with the corresponding errors observed in the previous figures.

4.4.4 Test 3: Sudden heating problem

In the last test, we consider a problem with random boundary condition. The gas is initially in a constant state with $\rho_0 = 1$, $\boldsymbol{U}_0 = (0, 0, 0)$, $T_0 = 1$ and $f_0(\mathbf{x}, \mathbf{v}) = \mathcal{M}[\rho_0, \boldsymbol{U}_0, T_0]$. At time $t = 0$, we suddenly change the wall temperature at left boundary $x = 0$ to

$$T_w(z) = 3(T_0 + sz), \ \ s = 0.2, \tag{4.4.9}$$

where the random variable z obeys the uniform distribution on $[-1, 1]$. We assume purely diffusive Maxwell boundary condition at $x = 0$ and homogeneous Neumann boundary condition at $x = 1$. The Knudsen number is set as $\varepsilon = 0.1$. This is a classical benchmark test in kinetic theory. With the sudden rise of the wall temperature, the gas close to the wall is

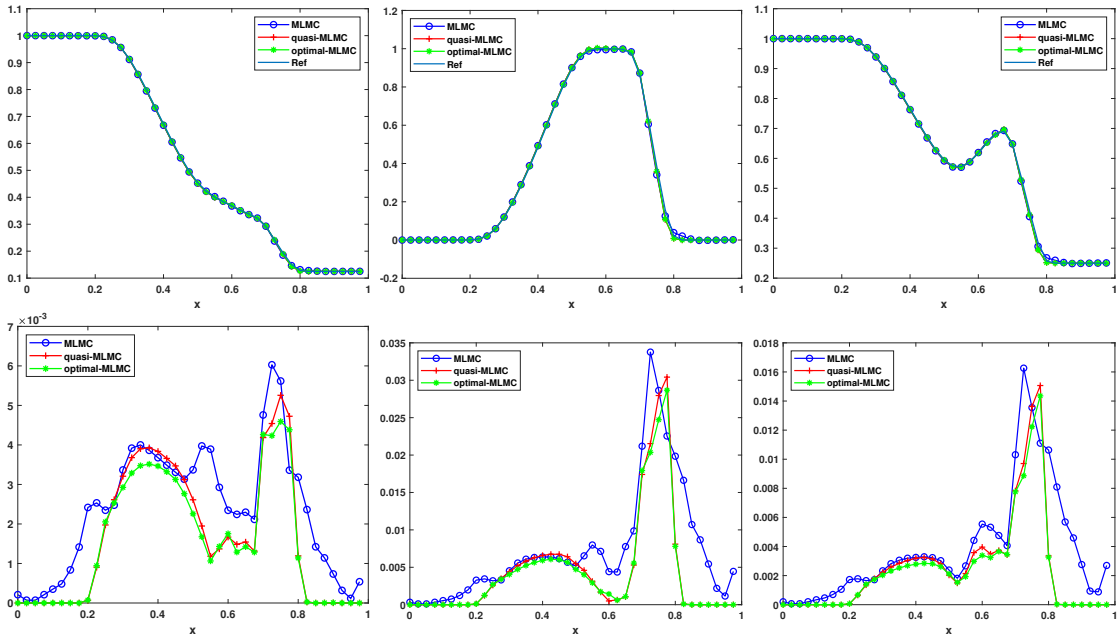

Figure 4.8. Test 2 (I): Approximated expectation of density $\mathbb{E}[\rho]$ (left), velocity $\mathbb{E}[U]$ (middle) and temperature $\mathbb{E}[T]$ (right) using MLMC, quasi-optimal MLMC and optimal MLMC methods at time $t = 0.15$ (top row). Relative error (4.4.3) of expectation of density (left), velocity (middle) and temperature (right) using three MLMC methods (bottom row).

heated and accordingly the pressure rises sharply and pushes the gas away forming a shock propagating into the domain.

We compare the three MLMC methods using parameters: mesh sizes $N_1 = 10, N_2 = 20, N_3 = 40$, and number of samples $M_1 = 1280, M_2 = 320, M_3 = 80$ (these parameters are chosen based on a similar test as in previous examples). The results are shown in Figure 4.14 and Figure 4.15. Again the control variate MLMC methods outperform the standard MLMC in all simulations and the optimal MLMC method yields slightly better results than the quasi-optimal MLMC.

4.5 Conclusions of this chapter

We have introduced a control variate multilevel Monte Carlo method for the BGK model of the Boltzmann equation with uncertainty. Well-posedness of the BGK equation with

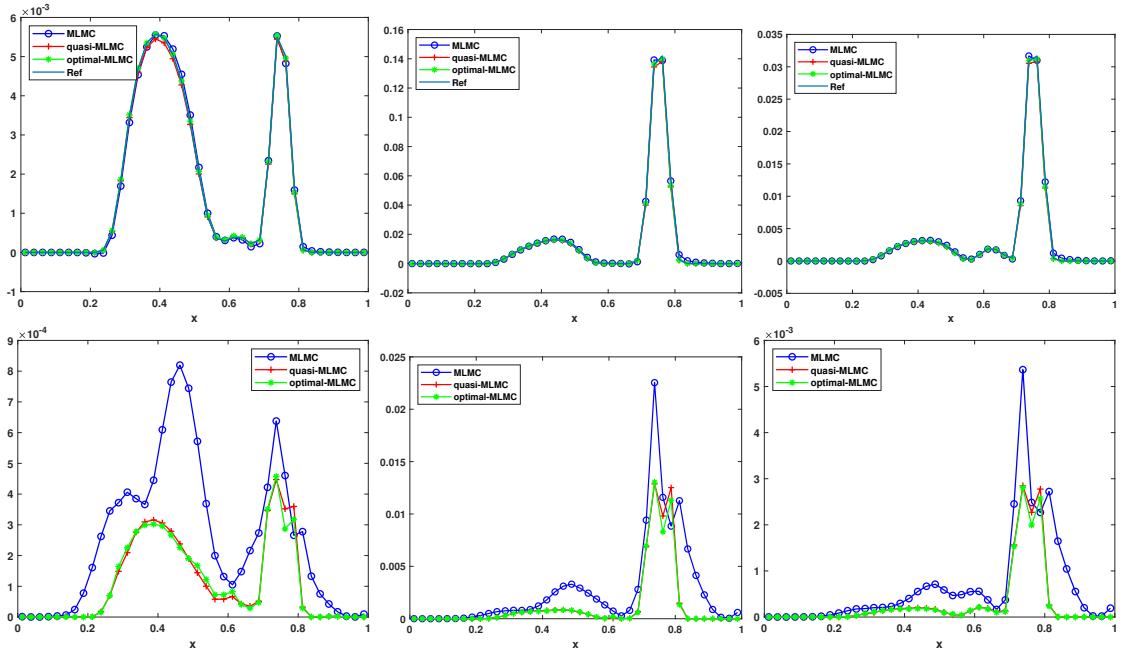

Figure 4.9. Test 2 (I): Approximated variance of density $\mathbb{V}[\rho]$ (left), velocity $\mathbb{V}[U]$ (middle) and temperature $\mathbb{V}[T]$ (right) using MLMC, quasi-optimal MLMC and optimal MLMC methods at time $t = 0.15$ (top row). Relative error (4.4.3) of variance of density (left), velocity (middle) and temperature (right) using three methods (bottom row).

random parameters, consistency and convergence analysis for various MC type methods are established. Extensive numerical results confirm that the MLMC methods perform much better than the standard MC, and the control variate MLMC is capable to provide further improvement over the conventional MLMC, in particular for problems close to fluid regimes and in presence of discontinuities, where the fidelity degree of the various levels is reduced and traditional gPC-SG based methods may fail (see [35]). In addition to an optimal strategy, we have introduced a simplified quasi-optimal approach that does not require solving a tridiagonal system of linear equations. In the numerical examples, this simplified approach provided only slightly less accurate results than those obtained with the optimal strategy. The control variate multilevel Monte Carlo methods here developed naturally extend to other kinetic equations of Boltzmann type which combines deterministic discretizations in the phase space with Monte Carlo sampling in the random space. In particular, even if

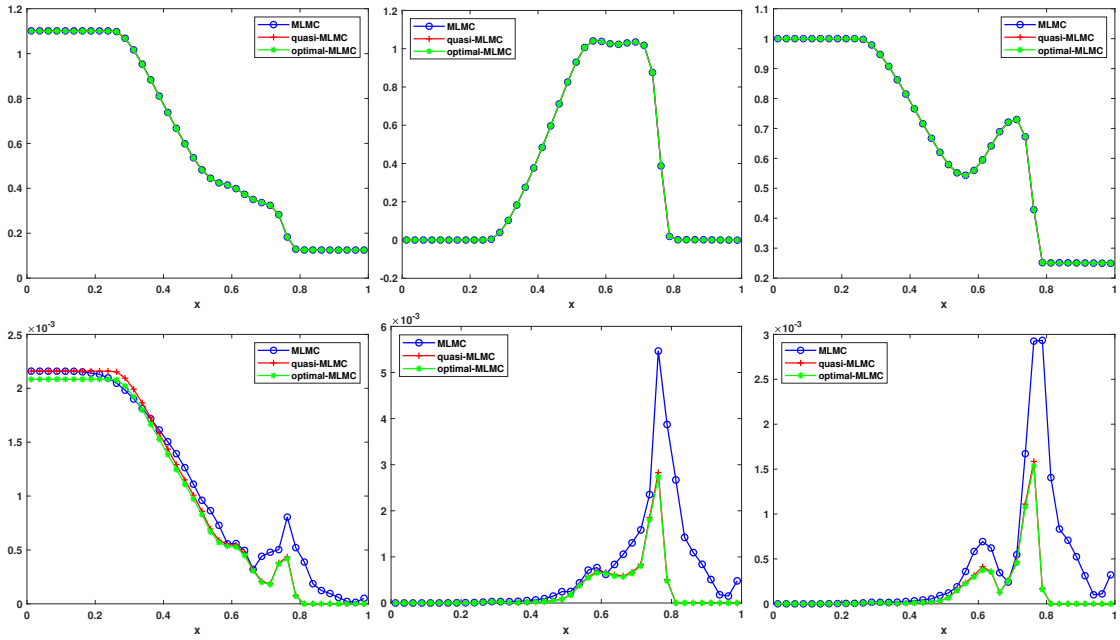

Figure 4.10. Test 2 (II): Approximated expectation of density $\mathbb{E}[\rho]$ (left), velocity $\mathbb{E}[U]$ (middle) and temperature $\mathbb{E}[T]$ (right) using MLMC, quasi-optimal MLMC and optimal MLMC methods at time $t = 0.15$ (top row). Relative error 4.4.3 of expectation of density (left), velocity (middle) and temperature (right) using three MLMC methods (bottom row).

our study was limited to one space dimension, we expect the gains of MLMC methods over standard MC to be even more significant in higher dimensions.

4.A Dimension reduction method and deterministic solver for the BGK equation

In this Appendix, we briefly describe the dimension reduction method adopted to reduce the computational complexity of the BGK equation and the details of the numerical methods used to discretize time, physical space and velocity space. Since the Monte Carlo methods are non-intrusive, our discussion will be based on the deterministic equation (1.0.6) for simplicity.

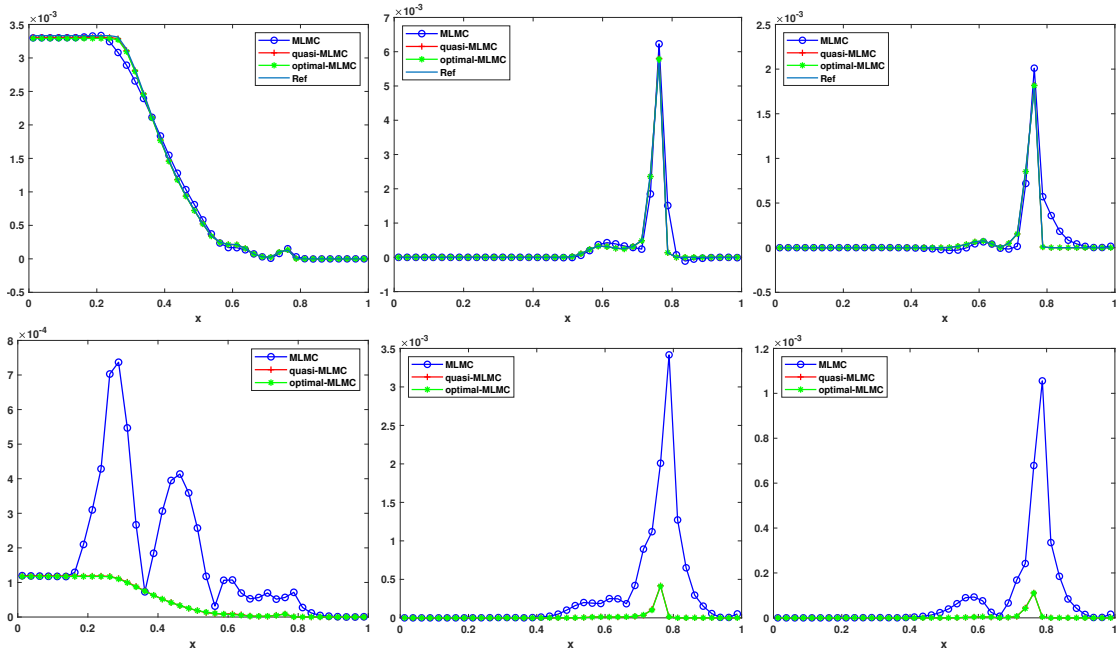

Figure 4.11. Test 2 (II): Approximated variance of density $\mathbb{V}[\rho]$ (left), velocity $\mathbb{V}[U]$ (middle) and temperature $\mathbb{V}[T]$ (right) using MLMC, quasi-optimal MLMC and optimal MLMC methods at time $t = 0.15$ (top row). Relative error (4.4.3) of variance of density (left), velocity (middle) and temperature (right) using three methods (bottom row).

4.A.1 The Chu reduction method

The BGK equation (1.0.6) is formulated in a six-dimensional phase-space where computations can be extremely expensive. Under certain homogeneity assumptions, one can reduce the dimension using the so-called Chu reduction [52].

Let $\mathbf{x} = (x_1, x_2, x_3)$, $\mathbf{v} = (v_1, v_2, v_3)$, and $\boldsymbol{U} = (U_1, U_2, U_3)$. If the solution f only varies in one spatial dimension, then effectively we are solving a one-dimensional problem and it is reasonable to assume the following:

$$\partial_{x_2} f = \partial_{x_3} f = 0, \quad U_2 = U_3 = 0. \tag{4.A.1}$$

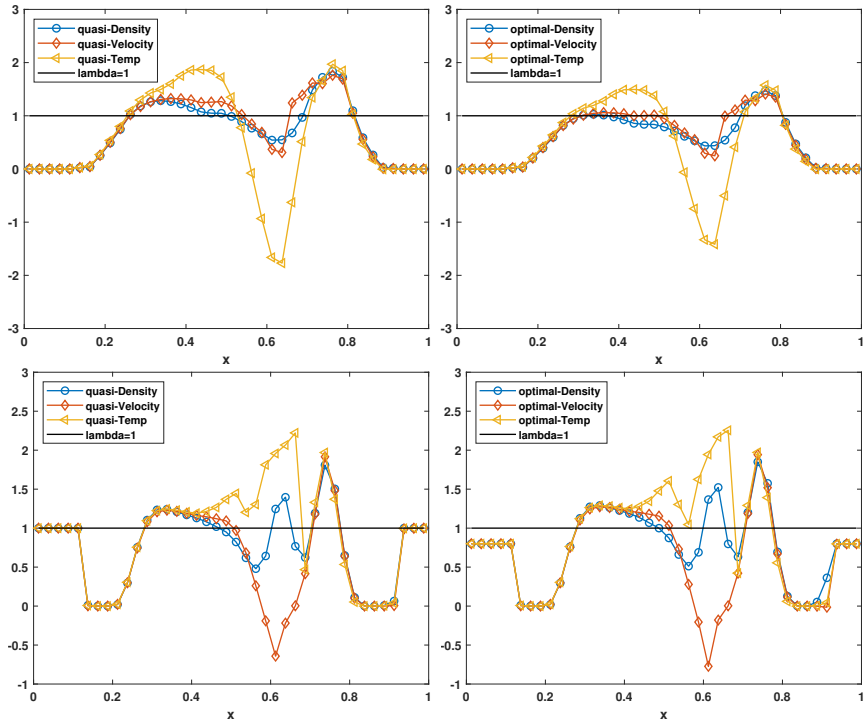

Figure 4.12. Test 2 (I): Values of λ_1 in quasi-optimal (left) and optimal (right) MLMC methods (top row). Values of λ_2 in quasi-optimal (left) and optimal (right) MLMC methods (bottom row).

Then the equation (1.0.6) becomes

$$\partial_t f(t,x_1,v_1,v_2,v_3) + v_1 \partial_{x_1} f(t,x_1,v_1,v_2,v_3) = \frac{1}{\varepsilon}\left(\mathcal{M}[\rho,\boldsymbol{U},T] - f(t,x_1,v_1,v_2,v_3)\right), \quad (4.A.2)$$

where

$$\mathcal{M}[\rho,\boldsymbol{U},T](t,x_1,v_1,v_2,v_3) = \frac{\rho(t,x_1)}{(2\pi T(t,x_1))^{\frac{3}{2}}} \exp\left(-\frac{(v_1-U_1(t,x_1))^2+v_2^2+v_3^2}{2T(t,x_1)}\right). \quad (4.A.3)$$

The Chu reduction proceeds by introducing two distribution functions:

$$\phi(t,x_1,v_1) := \iint_{\mathbb{R}^2} f(t,x_1,v_1,v_2,v_3)\,\mathrm{d}v_2\mathrm{d}v_3, \quad (4.A.4)$$

$$\psi(t,x_1,v_1) := \iint_{\mathbb{R}^2} \left(\frac{1}{2}v_2^2+\frac{1}{2}v_3^2\right) f(t,x_1,v_1,v_2,v_3)\,\mathrm{d}v_2\mathrm{d}v_3. \quad (4.A.5)$$

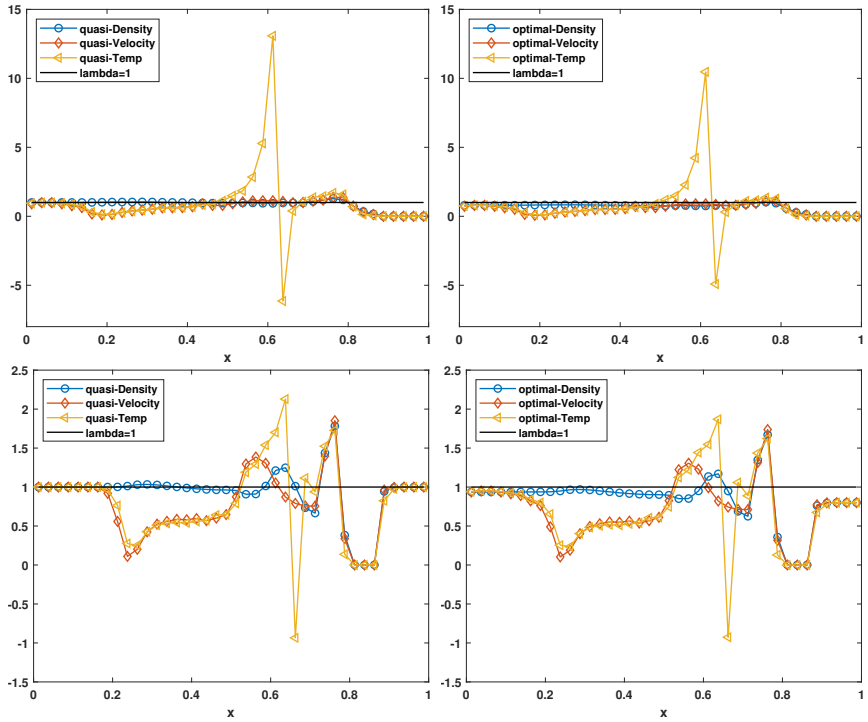

Figure 4.13. Test 2 (II): Values of λ_1 in quasi-optimal (left) and optimal (right) MLMC methods (top row). Values of λ_2 in quasi-optimal (left) and optimal (right) MLMC methods (bottom row).

It is then easy to derive that ϕ and ψ satisfy the following system:

$$\partial_t \phi(t,x_1,v_1) + v_1 \partial_{x_1} \phi(t,x_1,v_1) = \frac{1}{\varepsilon}\left(M_\phi(t,x_1,v_1) - \phi(t,x_1,v_1)\right), \tag{4.A.6}$$

$$\partial_t \psi(t,x_1,v_1) + v_1 \partial_{x_1} \psi(t,x_1,v_1) = \frac{1}{\varepsilon}\left(M_\psi(t,x_1,v_1) - \psi(t,x_1,v_1)\right), \tag{4.A.7}$$

where

$$M_\phi(t,x_1,v_1) := \iint_{\mathbb{R}^2} \mathcal{M}[\rho,\boldsymbol{U},T]\,\mathrm{d}v_2\mathrm{d}v_3 = \frac{\rho(t,x_1)}{\sqrt{2\pi T(t,x_1)}} \exp\left(-\frac{(v_1 - U_1(t,x_1))^2}{2T(t,x_1)}\right), \tag{4.A.8}$$

$$M_\psi(t,x_1,v_1) := \iint_{\mathbb{R}^2} \left(\frac{1}{2}v_2^2 + \frac{1}{2}v_3^2\right) \mathcal{M}[\rho,\boldsymbol{U},T]\,\mathrm{d}v_2\mathrm{d}v_3 = T(t,x_1) M_\phi. \tag{4.A.9}$$

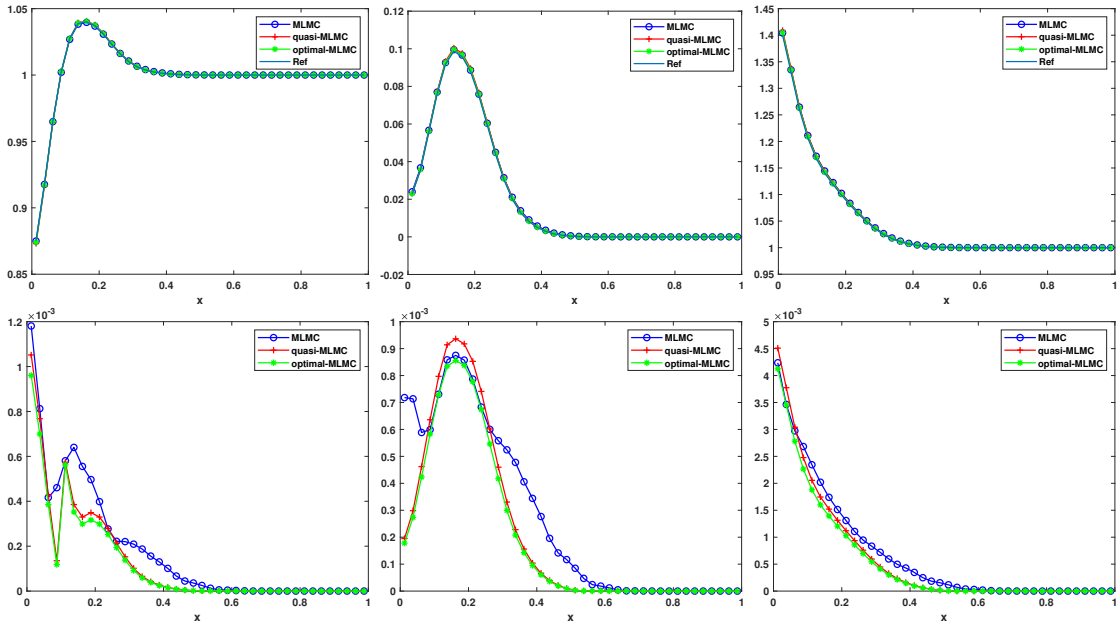

Figure 4.14. Test 3: Approximated expectation of density $\mathbb{E}[\rho]$ (left), velocity $\mathbb{E}[U]$ (middle) and temperature $\mathbb{E}[T]$ (right) using MLMC, quasi-optimal MLMC and optimal MLMC methods at time $t = 0.1$ (top row). Relative error (4.4.3) of expectation of density (left), velocity (middle) and temperature (right) using three methods (bottom row).

Denoting $\int_{\mathbb{R}} \cdot \, \mathrm{d}v_1 = \langle \cdot \rangle$, it is easy to see the following relation holds

$$\begin{aligned} \rho &= \int_{\mathbb{R}} \phi \, \mathrm{d}v_1 = \int_{\mathbb{R}} M_\phi \, \mathrm{d}v_1, \\ m &= \rho U_1 = \int_{\mathbb{R}} v_1 \phi \, \mathrm{d}v_1 = \int_{\mathbb{R}} v_1 M_\phi \, \mathrm{d}v_1, \\ E &= \frac{1}{2}\rho U_1^2 + \frac{3}{2}\rho T = \int_{\mathbb{R}} \left(\frac{1}{2} v_1^2 \phi + \psi \right) \mathrm{d}v_1 = \int_{\mathbb{R}} \left(\frac{1}{2} v_1^2 M_\phi + M_\psi \right) \mathrm{d}v_1. \end{aligned} \tag{4.A.10}$$

Now our task is to solve the reduced 1D BGK system (4.A.6)-(4.A.7).

4.A.2 The fully discrete scheme

The fully discrete scheme used to solve (4.A.6)-(4.A.7) consists of three components: velocity discretization, time discretization, and spatial discretization.

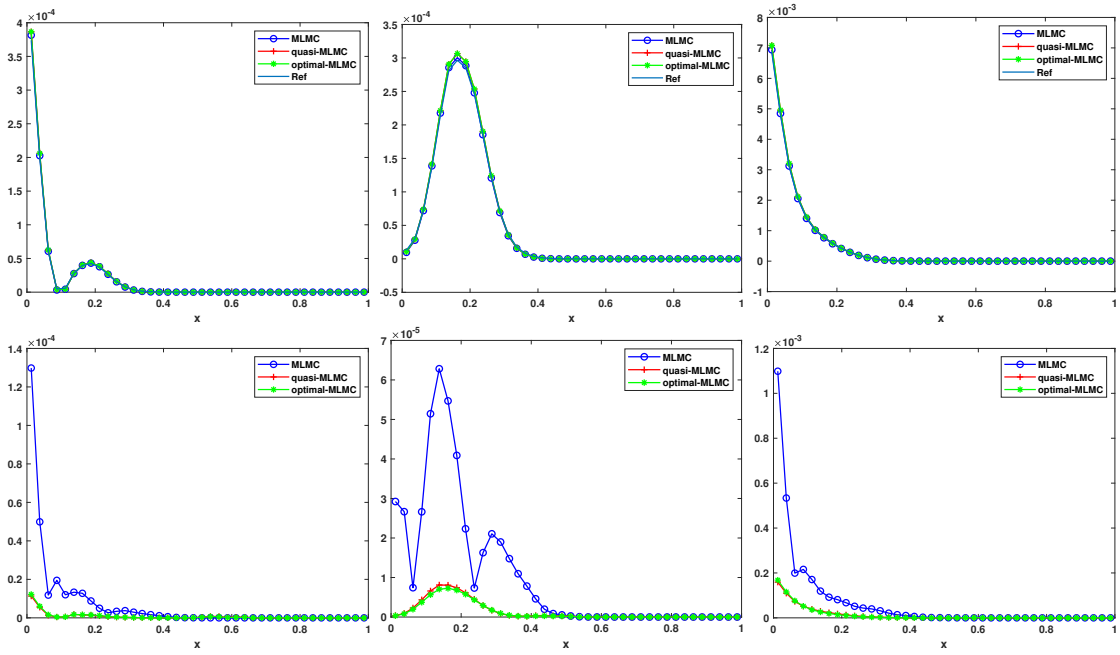

Figure 4.15. Test 3: Approximated variance of density $\mathbb{V}[\rho]$ (left), velocity $\mathbb{V}[U]$ (middle) and temperature $\mathbb{V}[T]$ (right) using MLMC, quasi-optimal MLMC and optimal MLMC methods at time $t = 0.1$ (top row). Relative error (4.4.3) of variance of density (left), velocity (middle) and temperature (right) using three methods (bottom row).

Velocity discretization

In the velocity space, we follow the discrete velocity method (see Section 4.1.1 in [76] or [21] for example), which satisfies a discrete entropy decay property.

We first truncate the infinite velocity domain into a bounded interval $[-R, R]$ and then discretize it using N_v-point Gauss quadrature with (ξ_k, w_k), $k = 1, 2, \ldots, N_v$ as abscissae and weights. To obtain M_ϕ, M_ψ from ϕ and ψ, normally one could use the relation in (4.A.10), where the continuous integral is replaced by the Gauss quadrature. However, due to the domain truncation error, the resulting moments are not sufficiently accurate. To remove this error, we assume

$$M_\phi = \exp(\alpha_1 + \alpha_2 v_1 + \alpha_3 v_1^2), \quad M_\psi = -\frac{1}{2\alpha_3} M_\phi, \tag{4.A.11}$$

and determine α_1, α_2, α_3 such that

$$\begin{bmatrix} \langle M_\phi \rangle \\ \langle v_1 M_\phi \rangle \\ \langle \frac{1}{2} v_1^2 M_\phi + M_\psi \rangle \end{bmatrix} = \begin{bmatrix} \langle \phi \rangle \\ \langle v_1 \phi \rangle \\ \langle \frac{1}{2} v_1^2 \phi + \psi \rangle \end{bmatrix} := \begin{bmatrix} \rho \\ m \\ E \end{bmatrix}, \tag{4.A.12}$$

where $\langle u(v_1) \rangle := \sum_{k=1}^{N_v} u(\xi_k) w_k$ denotes the quadrature sum in the interval $[-R, R]$. The above nonlinear system is solved by the Newton-Raphson algorithm.

Time discretization

Due to the possibly stiff collision term, we use the implicit-explicit Runge-Kutta (IMEX-RK) scheme [77], [78] for the time discretization. In particular, we employ the second-order IMEX-RK scheme proposed in [51], which is positivity preserving and asymptotic preserving (preserving the Euler limit without Δt resolving ε).

Specifically, we discretize (4.A.6) and (4.A.7) as

$$\begin{aligned}
\phi^{(\mathrm{i})} &= \phi^n - \Delta t \sum_{\mathrm{j}=1}^{\mathrm{i}-1} \tilde{a}_{\mathrm{ij}} v_1 \partial_{x_1} \phi^{(\mathrm{j})} + \Delta t \sum_{\mathrm{j}=1}^{\mathrm{i}} a_{\mathrm{ij}} \frac{1}{\varepsilon} \left(M_\phi^{(\mathrm{j})} - \phi^{(\mathrm{j})} \right), \quad \mathrm{i} = 1, \ldots, \nu, \\
\psi^{(\mathrm{i})} &= \psi^n - \Delta t \sum_{\mathrm{j}=1}^{\mathrm{i}-1} \tilde{a}_{\mathrm{ij}} v_1 \partial_{x_1} \psi^{(\mathrm{j})} + \Delta t \sum_{\mathrm{j}=1}^{\mathrm{i}} a_{\mathrm{ij}} \frac{1}{\varepsilon} \left(M_\psi^{(\mathrm{j})} - \psi^{(\mathrm{j})} \right), \quad \mathrm{i} = 1, \ldots, \nu, \\
\phi^{n+1} &= \phi^{(\nu)} + \alpha \Delta t^2 \frac{1}{\varepsilon^2} \left(M_\phi^{n+1} - \phi^{n+1} \right), \\
\psi^{n+1} &= \psi^{(\nu)} + \alpha \Delta t^2 \frac{1}{\varepsilon^2} \left(M_\psi^{n+1} - \psi^{n+1} \right),
\end{aligned} \tag{4.A.13}$$

where the values of the coefficients $\tilde{a}_{\mathrm{ij}}, a_{\mathrm{ij}}, \alpha$ are given in Section 2.6.1 of [51]. To implement the above scheme explicitly, we first solve the moment system for $\mathrm{i} = 1, \ldots, \nu$

$$\begin{aligned}
&\begin{bmatrix} \langle \phi^{(\mathrm{i})} \rangle \\ \langle v_1 \phi^{(\mathrm{i})} \rangle \\ \langle \frac{1}{2} v_1^2 \phi^{(\mathrm{i})} + \psi^{(\mathrm{i})} \rangle \end{bmatrix} = \begin{bmatrix} \langle \phi^{n} \rangle \\ \langle v_1 \phi^{n} \rangle \\ \langle \frac{1}{2} v_1^2 \phi^{n} + \psi^{n} \rangle \end{bmatrix} - \Delta t \sum_{\mathrm{j}=1}^{\mathrm{i}-1} \tilde{a}_{\mathrm{ij}} \begin{bmatrix} \langle v_1 \partial_{x_1} \phi^{(\mathrm{j})} \rangle \\ \langle v_1^2 \partial_{x_1} \phi^{(\mathrm{j})} \rangle \\ \langle \frac{1}{2} v_1^3 \partial_{x_1} \phi^{(\mathrm{j})} + v_1 \partial_{x_1} \psi^{(\mathrm{j})} \rangle \end{bmatrix}, \\
&\begin{bmatrix} \langle \phi^{n+1} \rangle \\ \langle v_1 \phi^{n+1} \rangle \\ \langle \frac{1}{2} v_1^2 \phi^{n+1} + \psi^{n+1} \rangle \end{bmatrix} = \begin{bmatrix} \langle \phi^{(\nu)} \rangle \\ \langle v_1 \phi^{(\nu)} \rangle \\ \langle \frac{1}{2} v_1^2 \phi^{(\nu)} + \psi^{(\nu)} \rangle \end{bmatrix},
\end{aligned} \tag{4.A.14}$$

which is obtained by taking the moments of (4.A.13) and using (4.A.12). Hence we can obtain $\rho^{(\mathrm{i})}$, $m^{(\mathrm{i})}$ and $E^{(\mathrm{i})}$ first, and use them to define $M_\phi^{(\mathrm{i})}$ and $M_\psi^{(\mathrm{i})}$. Finally we solve (4.A.13) to get $\phi^{(\mathrm{i})}$ and $\psi^{(\mathrm{i})}$.

Spatial discretization

In the physical space, we use the second order MUSCL finite volume scheme [79].

Here we take the following first order in time scheme for ϕ as an illustration (suppose it is evaluated at velocity point $v_1 = \xi_k$):

$$\frac{\phi_k^{n+1}(x_1) - \phi_k^n(x_1)}{\Delta t} + \xi_k \partial_{x_1} \phi_k^n(x_1) = \frac{1}{\varepsilon} \left((M_\phi)_k^{n+1}(x_1) - \phi_k^{n+1}(x_1) \right). \tag{4.A.15}$$

Suppose $x_1 \in [a, b]$ and $[a, b]$ is divided into N_x uniform cells with size $\Delta x = (b-a)/N_x$, where $a = x_{\frac{1}{2}}$, $b = x_{N_x + \frac{1}{2}}$. In the cell $[x_{\mathrm{j}-\frac{1}{2}}, x_{\mathrm{j}+\frac{1}{2}}]$, define the cell average as

$$\phi_{\mathrm{j},k}^n := \frac{1}{\Delta x} \int_{x_{\mathrm{j}-\frac{1}{2}}}^{x_{\mathrm{j}+\frac{1}{2}}} \phi_k^n(x_1) \, \mathrm{d}x_1. \tag{4.A.16}$$

Then integrating (4.A.15) over $[x_{\mathrm{j}-\frac{1}{2}}, x_{\mathrm{j}+\frac{1}{2}}]$ yields

$$\frac{\phi_{\mathrm{j},k}^{n+1} - \phi_{\mathrm{j},k}^n}{\Delta t} + \frac{F_{\mathrm{j}+\frac{1}{2},k}^n - F_{\mathrm{j}-\frac{1}{2},k}^n}{\Delta x} = \frac{1}{\varepsilon} \left((M_\phi)_{\mathrm{j},k}^{n+1} - \phi_{\mathrm{j},k}^{n+1} \right), \tag{4.A.17}$$

where $(M_\phi)^{n+1}_{\mathrm{j},k} := (M_\phi)^{n+1}_k(x_\mathrm{j})$. Note that we have replaced the cell average of $(M_\phi)^{n+1}_k$ by its point value at cell center x_j (the error introduced by this is $O(\Delta x^2)$ which does not destroy the overall order of the method). $F^n_{\mathrm{j}+\frac{1}{2},k}$ is the flux at interface $x_{\mathrm{j}+\frac{1}{2}}$ and is defined as

$$F^n_{\mathrm{j}+\frac{1}{2},k} = \max(0,\xi_k)\phi^n_{l,\mathrm{j},k} + \min(0,\xi_k)\phi^n_{r,\mathrm{j}+1,k}, \tag{4.A.18}$$

with the left interface and right interface values $\phi^n_{l,\mathrm{j},k}, \phi^n_{r,\mathrm{j},k}$ given by

$$\begin{cases} \phi^n_{l,\mathrm{j},k} = \phi^n_{\mathrm{j},k} + \dfrac{1}{2}\Delta x\sigma^n_{\mathrm{j},k}, \\ \phi^n_{r,\mathrm{j},k} = \phi^n_{\mathrm{j},k} - \dfrac{1}{2}\Delta x\sigma^n_{\mathrm{j},k}, \end{cases} \tag{4.A.19}$$

where $\sigma^n_{\mathrm{j},k}$ is the slope of the linear reconstruction and is chosen to be the MC limiter ($\theta = 2$):

$$\sigma^n_{\mathrm{j},k} = \mathrm{minmod}\left(\frac{\phi^n_{\mathrm{j}+1,k} - \phi^n_{\mathrm{j}-1,k}}{2\Delta x}, \theta\left(\frac{\phi^n_{\mathrm{j},k} - \phi^n_{\mathrm{j}-1,k}}{\Delta x}\right), \theta\left(\frac{\phi^n_{\mathrm{j}+1,k} - \phi^n_{\mathrm{j},k}}{\Delta x}\right)\right). \tag{4.A.20}$$

5. CONCLUDING REMARKS AND FUTURE WORKS

In this thesis, we've introduced works in dealing with the challenges in high dimensions and uncertainty quantification problems for kinetic equations.

For high dimension problems, we focus on the dynamic low-rank method. Kinetic equations under dynamic low-rank frameworks have many remarkable properties:

- Collision operator (Boltzmann, BGK, Linearized-Boltzmann) are local in $\mathbf{x}$ while convection part are local in $\mathbf{v}$. This separated structure is well-suited for dynamic low-rank methods.
- Dynamic low-rank methods can drastically increase computational efficiency as well as reduce memory cost. Adaptivity strategy can further reduce computational cost.
- In the near fluid regime, kinetic equations are low-rank. In kinetic regime, there are some low-rank flows (e.g. normal shock wave)
- Numerical methods under low-rank framework can be well-designed to achieve AP and higher order.

We introduced works in linear transport and full Boltzmann equations. The rank dependency are investigated in both the fluid regime and kinetic regime. The numerical schemes are well-designed to achieve high orders and asymptotic preserving in linear transport equation. Adaptivity are also studied in the numerical computations of steady state solutions in full Boltzmann equation with dynamic thresholding strategy to further increase computational efficiency. A series benchmark tests verified the accuracy and efficiency of the proposed dynamic low rank methods in these equations.

On the other hand, in order to inherit good preperties from existing deterministic methods, we focus on the non-intrusive sampling method to tackle kinetic equations with uncertainties. We have introduced a control variate multilevel Monte Carlo method as well as theoretical analysis regarding the well-posedness, consistency and convergence analysis for various MC type methods. Extensive numerical results confirm that the MLMC methods perform much better than the standard MC, and the control variate MLMC is capable to

provide further improvement over the conventional MLMC. Moreover, this method naturally extend to other kinetic equations of Boltzmann type which combines deterministic discretizations in the phase space with Monte Carlo sampling in the random space.
Following works in this thesis, there are many relevant topics that worth investigating including but not limited to:

- Low-rank methods with AP properties in other types of kinetic equations
- General rank dependence investigations in kinetic equations.
- Adaptive low-rank methods for general time dependent kinetic equations.
- Uncertainty quantification in other kinetic equations

REFERENCES

[1] S. Chapman and T. G. Cowling, *The mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction and diffusion in gases*. Cambridge university press, 1990.

[2] C. Cercignani, *Rarefied gas dynamics: from basic concepts to actual calculations*. Cambridge University Press, 2000, vol. 21.

[3] C. K. Birdsall and A. B. Langdon, *Plasma physics via computer simulation*. CRC press, 2018.

[4] P. A. Markowich, C. A. Ringhofer, and C. Schmeiser, *Semiconductor equations*. Springer Science & Business Media, 2012.

[5] G. Naldi, L. Pareschi, and G. Toscani, *Mathematical modeling of collective behavior in socio-economic and life sciences*. Springer Science & Business Media, 2010.

[6] C. Cercignani, "The boltzmann equation," in *The Boltzmann Equation and Its Applications*, Springer, 1988, pp. 40–103.

[7] C. Villani, "A review of mathematical topics in collisional kinetic theory," *Handbook of mathematical fluid dynamics*, vol. 1, no. 71-305, pp. 3–8, 2002.

[8] G. A. Bird, "Molecular gas dynamics and the direct simulation of gas flows," *Molecular gas dynamics and the direct simulation of gas flows*, 1994.

[9] K. Nanbu, "Direct simulation scheme derived from the boltzmann equation. i. monocomponent gases," *Journal of the Physical Society of Japan*, vol. 49, no. 5, pp. 2042–2049, 1980.

[10] G. Dimarco and L. Pareschi, "Numerical methods for kinetic equations," *Acta Numerica*, vol. 23, pp. 369–520, 2014.

[11] C. Buet, "A discrete-velocity scheme for the boltzmann operator of rarefied gas dynamics," *Transport Theory and Statistical Physics*, vol. 25, no. 1, pp. 33–60, 1996.

[12] A. Vasiljevitch Bobylev, A. Palczewski, and J. Schneider, "On approximation of the boltzmann equation by discrete velocity models," *Comptes rendus de l'Académie des sciences. Série 1, Mathématique*, vol. 320, no. 5, pp. 639–644, 1995.

[13] A. Bobylev and S. Rjasanow, "Fast deterministic method of solving the boltzmann equation for hard spheres," *European Journal of Mechanics-B/Fluids*, vol. 18, no. 5, pp. 869–887, 1999.

[14] L. Pareschi and B. Perthame, "A fourier spectral method for homogeneous boltzmann equations," *Transport Theory and Statistical Physics*, vol. 25, no. 3-5, pp. 369–382, 1996.

[15] P. L. Bhatnagar, E. P. Gross, and M. Krook, "A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems," *Physical review*, vol. 94, no. 3, p. 511, 1954.

[16] A. Jüngel, *Transport equations for semiconductors*. Springer, 2009, vol. 773.

[17] E. P. Gross and E. A. Jackson, "Kinetic models and the linearized boltzmann equation," *The physics of fluids*, vol. 2, no. 4, pp. 432–441, 1959.

[18] L. H. Holway Jr, "Kinetic theory of shock structure using an ellipsoidal distribution function," *Rarefied Gas Dynamics, Volume 1*, vol. 1, p. 193, 1965.

[19] M. K. Gobbert, S. G. Webster, and T. S. Cale, "A galerkin method for the simulation of the transient 2-d/2-d and 3-d/3-d linear boltzmann equation," *Journal of Scientific Computing*, vol. 30, no. 2, pp. 237–273, 2007.

[20] V. Kolobov, R. Arslanbekov, V. V. Aristov, A. Frolova, and S. A. Zabelok, "Unified solver for rarefied and continuum flows with adaptive mesh and algorithm refinement," *Journal of Computational Physics*, vol. 223, no. 2, pp. 589–608, 2007.

[21] L. Mieussens, "Discrete-velocity models and numerical schemes for the boltzmann-bgk equation in plane and axisymmetric geometries," *Journal of Computational Physics*, vol. 162, no. 2, pp. 429–466, 2000.

[22] J. Yang and J.-c. Huang, "Rarefied flow computations using nonlinear model boltzmann equations," *Journal of Computational Physics*, vol. 120, no. 2, pp. 323–339, 1995.

[23] S. Jaiswal, A. A. Alexeenko, and J. Hu, "A discontinuous galerkin fast spectral method for the full boltzmann equation with general collision kernels," *Journal of Computational Physics*, vol. 378, pp. 178–208, 2019.

[24] W. Su, P. Wang, Y. Zhang, and L. Wu, "Implicit discontinuous galerkin method for the boltzmann equation," *Journal of Scientific Computing*, vol. 82, no. 2, pp. 1–35, 2020.

[25] L. Einkemmer and C. Lubich, "A low-rank projector-splitting integrator for the vlasov–poisson equation," *SIAM Journal on Scientific Computing*, vol. 40, no. 5, B1330–B1360, 2018.

[26] L. Einkemmer and C. Lubich, "A quasi-conservative dynamical low-rank algorithm for the vlasov equation," *SIAM Journal on Scientific Computing*, vol. 41, no. 5, B1061–B1081, 2019.

[27] L. Einkemmer, J. Hu, and L. Ying, "An efficient dynamical low-rank algorithm for the boltzmann-bgk equation close to the compressible viscous flow regime," *arXiv preprint arXiv:2101.07104*, 2021.

[28] L. Einkemmer, "A low-rank algorithm for weakly compressible flow," *SIAM Journal on Scientific Computing*, vol. 41, no. 5, A2795–A2814, 2019.

[29] Z. Peng, R. G. McClarren, and M. Frank, "A low-rank method for two-dimensional time-dependent radiation transport calculations," *Journal of Computational Physics*, vol. 421, p. 109 735, 2020.

[30] C. Mouhot and L. Pareschi, "Fast algorithms for computing the boltzmann collision operator," *Mathematics of computation*, vol. 75, no. 256, pp. 1833–1852, 2006.

[31] C. Cercignani, R. Illner, and M. Pulvirenti, *The mathematical theory of dilute gases*. Springer Science & Business Media, 2013, vol. 106.

[32] J. Hu and S. Jin, "Uncertainty quantification for kinetic equations," in *Uncertainty quantification for hyperbolic and kinetic equations*, Springer, 2017, pp. 193–229.

[33] R. G. Ghanem and P. D. Spanos, "Stochastic finite element method: Response statistics," in *Stochastic finite elements: a spectral approach*, Springer, 1991, pp. 101–119.

[34] D. Xiu and J. S. Hesthaven, "High-order collocation methods for differential equations with random inputs," *SIAM Journal on Scientific Computing*, vol. 27, no. 3, pp. 1118–1139, 2005.

[35] B. Després, G. Poëtte, and D. Lucor, "Robust uncertainty propagation in systems of conservation laws with the entropy closure method," in *Uncertainty quantification in computational fluid dynamics*, Springer, 2013, pp. 105–149.

[36] M. B. Giles, "Multilevel monte carlo path simulation," *Operations research*, vol. 56, no. 3, pp. 607–617, 2008.

[37] S. Mishra, N. H. Risebro, C. Schwab, and S. Tokareva, "Numerical solution of scalar conservation laws with random flux functions," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 4, no. 1, pp. 552–591, 2016.

[38] G. Dimarco and L. Pareschi, "Multi-scale control variate methods for uncertainty quantification in kinetic equations," *Journal of Computational Physics*, vol. 388, pp. 63–89, 2019.

[39] Z. Ding, L. Einkemmer, and Q. Li, "Dynamical low-rank integrator for the linear boltzmann equation: Error analysis in the diffusion limit," *arXiv preprint arXiv:1907.04247*, 2019.

[40] S. Jin, "Efficient asymptotic-preserving (ap) schemes for some multiscale kinetic equations," *SIAM Journal on Scientific Computing*, vol. 21, no. 2, pp. 441–454, 1999.

[41] S. Jin, "Asymptotic preserving (ap) schemes for multiscale kinetic and hyperbolic equations: A review," *Lecture notes for summer school on methods and models of kinetic theory (M&MKT), Porto Ercole (Grosseto, Italy)*, pp. 177–216, 2010.

[42] P. Degond and F. Deluzet, "Asymptotic-preserving methods and multiscale models for plasma physics," *Journal of Computational Physics*, vol. 336, pp. 429–457, 2017.

[43] J. Hu, S. Jin, and Q. Li, "Asymptotic-preserving schemes for multiscale hyperbolic and kinetic equations," in *Handbook of Numerical Analysis*, vol. 18, Elsevier, 2017, pp. 103–129.

[44] M. Lemou and L. Mieussens, "A new asymptotic preserving scheme based on micro-macro formulation for linear kinetic equations in the diffusion limit," *SIAM Journal on Scientific Computing*, vol. 31, no. 1, pp. 334–368, 2008.

[45] J.-G. Liu and L. Mieussens, "Analysis of an asymptotic preserving scheme for linear kinetic equations in the diffusion limit," *SIAM Journal on Numerical Analysis*, vol. 48, no. 4, pp. 1474–1491, 2010.

[46] S. Boscarino, L. Pareschi, and G. Russo, "Implicit-explicit runge–kutta schemes for hyperbolic systems and kinetic equations in the diffusion limit," *SIAM Journal on Scientific Computing*, vol. 35, no. 1, A22–A51, 2013.

[47] J. Jang, F. Li, J.-M. Qiu, and T. Xiong, "High order asymptotic preserving dg-imex schemes for discrete-velocity kinetic equations in a diffusive scaling," *Journal of Computational Physics*, vol. 281, pp. 199–224, 2015.

[48] C. Lubich and I. V. Oseledets, "A projector-splitting integrator for dynamical low-rank approximation," *BIT Numerical Mathematics*, vol. 54, no. 1, pp. 171–188, 2014.

[49] A. Dektor, A. Rodgers, and D. Venturi, "Rank-adaptive tensor methods for high-dimensional nonlinear pdes," *Journal of Scientific Computing*, vol. 88, no. 2, pp. 1–27, 2021.

[50] B. Perthame and M. Pulvirenti, "Weighted l∞ bounds and uniqueness for the boltzmann bgk model," *Archive for rational mechanics and analysis*, vol. 125, no. 3, pp. 289–295, 1993.

[51] J. Hu, R. Shu, and X. Zhang, "Asymptotic-preserving and positivity-preserving implicit-explicit schemes for the stiff bgk equation," *SIAM Journal on Numerical Analysis*, vol. 56, no. 2, pp. 942–973, 2018.

[52] C. Chu, "Kinetic-theoretic description of the formation of a shock wave," *The Physics of Fluids*, vol. 8, no. 1, pp. 12–22, 1965.

[53] L. Einkemmer, J. Hu, and Y. Wang, "An asymptotic-preserving dynamical low-rank method for the multi-scale multi-dimensional linear transport equation," *Journal of Computational Physics*, vol. 439, p. 110 353, 2021.

[54] O. Koch and C. Lubich, "Dynamical low-rank approximation," *SIAM Journal on Matrix Analysis and Applications*, vol. 29, no. 2, pp. 434–454, 2007.

[55] E. Kieri, C. Lubich, and H. Walach, "Discretized dynamical low-rank approximation in the presence of small singular values," *SIAM Journal on Numerical Analysis*, vol. 54, no. 2, pp. 1020–1038, 2016.

[56] C. Lubich, "Time integration in the multiconfiguration time-dependent hartree method of molecular quantum dynamics," *Applied Mathematics Research eXpress*, vol. 2015, no. 2, pp. 311–328, 2015.

[57] C. Lubich, B. Vandereycken, and H. Walach, "Time integration of rank-constrained tucker tensors," *SIAM Journal on Numerical Analysis*, vol. 56, no. 3, pp. 1273–1290, 2018.

[58] L. Einkemmer and A. Ostermann, "An almost symmetric strang splitting scheme for the construction of high order composition methods," *Journal of computational and applied mathematics*, vol. 271, pp. 307–318, 2014.

[59] L. Einkemmer and A. Ostermann, "An almost symmetric strang splitting scheme for nonlinear evolution equations," *Computers & Mathematics with Applications*, vol. 67, no. 12, pp. 2144–2157, 2014.

[60] K. Küpper, M. Frank, and S. Jin, "An asymptotic preserving two-dimensional staggered grid method for multiscale transport equations," *SIAM Journal on Numerical Analysis*, vol. 54, no. 1, pp. 440–461, 2016.

[61] L. Einkemmer, A. Ostermann, and C. Piazzola, "A low-rank projector-splitting integrator for the vlasov–maxwell equations with divergence correction," *Journal of Computational Physics*, vol. 403, p. 109 063, 2020.

[62] *SPHERE_LEBEDEV_RULE quadrature rules for the sphere*, https://people.sc.fsu.edu/~jburkardt/datasets/sphere_lebedev_rule/sphere_lebedev_rule.html, Accessed: 2019-12-01.

[63] M. P. Laiu, M. Frank, and C. D. Hauck, "A positive asymptotic-preserving scheme for linear kinetic transport equations," *SIAM Journal on Scientific Computing*, vol. 41, no. 3, A1500–A1526, 2019.

[64] I. M. Gamba, J. R. Haack, C. D. Hauck, and J. Hu, "A fast spectral method for the boltzmann collision operator with general collision kernels," *SIAM Journal on Scientific Computing*, vol. 39, no. 4, B658–B674, 2017.

[65] S. Deshpande, "Kinetic theory based new upwind methods for inviscid compressible flows," in *24th Aerospace Sciences Meeting*, 1986, p. 275.

[66] J. Mandal and S. Deshpande, "Kinetic flux vector splitting for euler equations," *Computers & fluids*, vol. 23, no. 2, pp. 447–478, 1994.

[67] T. Ohwada, "Structure of normal shock waves: Direct numerical analysis of the boltzmann equation for hard-sphere molecules," *Physics of Fluids A: Fluid Dynamics*, vol. 5, no. 1, pp. 217–234, 1993.

[68] S. Harris, *An introduction to the theory of the Boltzmann equation*. Courier Corporation, 2004.

[69] J. Hu, L. Pareschi, and Y. Wang, "Uncertainty quantification for the bgk model of the boltzmann equation using multilevel variance reduced monte carlo methods," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 9, no. 2, pp. 650–680, 2021.

[70] B. Perthame, "Global existence to the bgk model of boltzmann equation," *Journal of Differential equations*, vol. 82, no. 1, pp. 191–205, 1989.

[71] G. Dimarco and L. Pareschi, "Multiscale variance reduction methods based on multiple control variates for kinetic equations with uncertainties," *Multiscale Modeling & Simulation*, vol. 18, no. 1, pp. 351–382, 2020.

[72] R. J. LeVeque and R. J. Leveque, *Numerical methods for conservation laws*. Springer, 1992, vol. 132.

[73] A. A. Gorodetsky, G. Geraci, M. S. Eldred, and J. D. Jakeman, "A generalized approximate control variate framework for multifidelity uncertainty quantification," *Journal of Computational Physics*, vol. 408, p. 109 257, 2020.

[74] D. Xiu, *Numerical methods for stochastic computations*. Princeton university press, 2010.

[75] S. Krumscheid, F. Nobile, and M. Pisaroni, "Quantifying uncertain system outputs via the multilevel monte carlo method—part i: Central moment estimation," *Journal of Computational Physics*, vol. 414, p. 109 466, 2020.

[76] E. Gabetta, L. Pareschi, and G. Toscani, "Relaxation schemes for nonlinear kinetic equations," *SIAM Journal on Numerical Analysis*, vol. 34, no. 6, pp. 2168–2194, 1997.

[77] G. Dimarco and L. Pareschi, "Asymptotic preserving implicit-explicit runge–kutta methods for nonlinear kinetic equations," *SIAM Journal on Numerical Analysis*, vol. 51, no. 2, pp. 1064–1087, 2013.

[78] S. Pieraccini and G. Puppo, "Implicit–explicit schemes for bgk kinetic equations," *Journal of Scientific Computing*, vol. 32, no. 1, pp. 1–28, 2007.

[79] B. Van Leer, "Towards the ultimate conservative difference scheme. v. a second-order sequel to godunov's method," *Journal of computational Physics*, vol. 32, no. 1, pp. 101–136, 1979.

VITA

Yubo Wang, born in September, 1993, is a Ph.D candidate in the Department of Mathematics, Purdue University. He was born in Hengyang, China and grew up in Shenzhen, China. He obtained his bachelor degree in Mathematics from Nankai University in June, 2016 and have become a graduate student at Purdue University since August, 2016