Moment Approximations to the Boltzmann Equation and Their Numerical Solutions Based on Finite Element Methods

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der RWTH Aachen University zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Edilbert Christhuraj, M.Sc
aus
Trichy, Indien

Berichter: Prof. Dr. rer. nat. Manuel Torrilhon
Dr. rer. nat. Julian Koellermeier

Tag der mündlichen Prüfung: 21.03.2025

Moment-Approximationen der Boltzmann-Gleichung und deren numerische Lösungen auf Basis von Finite-Elemente-Methoden

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der RWTH Aachen University zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Edilbert Christhuraj, M.Sc
aus
Trichy, Indien

Berichter: Prof. Dr. rer. nat. Manuel Torrilhon
Dr. rer. nat. Julian Koellermeier

Tag der mündlichen Prüfung: 21.03.2025

Declaration of Authorship

I, Edilbert Christhuraj, declare that this thesis and the work presented in it are my own and has been generated by me as the result of my own original research.

I do solemnly swear that:

- This work was done wholly or mainly while in candidature for the doctoral degree at this faculty and university;
- Where any part of this thesis has been previously been submitted for a degree or any other qualification at this university or any other institution, this has been clearly stated;
- Where I have consulted the published work of others or myself, this is always clearly attributed;
- Where I have quoted from the work of others or myself, the source is always given. This thesis is entirely my own work, with the exception of such quotations;
- I have acknowledged all major sources of assistance;
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- Parts of this work have been published before as:
  - **E. Christhuraj**, M. Torrilhon, *Generic moment systems and FEM-based numericals solutions for the Botlzmann equation*, AIP Conference Proceedings, DOI:10.1063/5.0187559
  - J. Bünger, **E. Christhuraj**, A. Hanke, M. Torrilhon, *Structured derivation of moment equations and stable boundary conditions with an introduction to symmetric, trace-free tensors*, Kinetic and Related Models, DOI:10.3934/krm.2022035

Edilbert Christhuraj

Acknowledgements

I would like to express my deepest gratitude to Professor Manuel Torrilhon for believing in my capabilities. He introduced me to this research avenue five years ago, and since then, he has been with me on this journey, advising me at every step of the development. I am inspired by his sincerity, which extends not only to his academic pursuits but also to his interactions with others. Thank you, Professor Manuel Torrilhon, for supporting me during both my highs and lows.

During my time at ACoM, I have met many wonderful people around the world who have contributed in various ways to the betterment of my professional and personal endeavors. Given the length of the list and the risk of inadvertently missing someone's name, I want to thank each and every one of you. It is the thought-provoking conversations, philosophical debates, and joyful moments I shared with you that have shaped my perspectives on life.

Last but not least, I want to thank my dear family for patiently waiting for me while I was writing my thesis and for cheering me up when I felt down. It is because of you that I am who I am today.

Contents

Abstract

Non-equilibrium gas flows occur in many physical processes, such as the re-entry of space vehicles into the atmosphere, vacuum devices like pumps, and miniaturized micro-electro-mechanical systems, to name a few. Simulating these rarefied gas flows has remained an active research area within the kinetic gas theory community. Kinetic models, such as the Boltzmann equation, accurately describe the evolution of gas flows across a wide range of flow regimes—from near-continuum to highly rarefied. The narrow regime that connects near-continuum flows to highly non-equilibrium flows is known as the transition regime. Simulating gas flows in this regime is challenging because classical fluid dynamics equations, such as the Navier-Stokes-Fourier equations, are not valid, and while the Boltzmann equation is valid, it is costly to solve directly.

Tracing its origins in kinetic gas theory, the moment method describes an approximation to the Boltzmann equation. It offers a balance between accuracy and computational cost. The inherent nature of the moment method leads to a hierarchy of moment models, each of increasing moment order. Lower-order moment models are viewed as extensions of classical fluid dynamics theory. As the moment order increases, the models become more comprehensive and, in theory, are better suited for simulating complex processes.

This thesis advances the application and development of moment equations for simulating non-equilibrium gas flows by extending their use to three-dimensional problems and providing a generic numerical framework for solving moment systems. It consists of two primary works.

In the first part of the thesis, we demonstrate the applicability of moment equations to a real-world, three-dimensional problem—a significant step forward, as previous studies have predominantly focused on one- or two-dimensional processes. This work explores the three-dimensional regularized 13-moment (R13) equations. It covers the theoretical foundation of these equations, their variational formulation, and implementation using FEniCS. The implementation is validated through simulations of flow over a sphere, and its practical applicability is demonstrated by investigating the Crookes radiometer. The study quantifies the radiometric force and examines how factors like gas pressure, vane thickness and length affect this force.

In the second part, we present a systematic approach for deriving generic moment systems and develop a finite-element-based numerical framework capable of solving moment equations of arbitrary order. We discuss the advantages of formulating higher-order partial differential equations as first-order systems and explore entropy-stable boundary conditions. The work also includes a generic variational formulation and an implementation of the 13-moment equations, accompanied by various numerical simulations to validate the framework. This approach not only broadens the scope of moment models but also provides a systematic and scalable pathway to higher-order moment equations, enhancing their applicability across a wider range of non-equilibrium gas flow problems.

Together, these contributions significantly advance the utility and adaptability of moment equations, bridging the gap between theoretical models and practical, computationally feasible solutions for non-equilibrium gas dynamics.

Abstract

Nicht-Gleichgewichts-Gasströmungen treten in vielen physikalischen Prozessen auf, wie etwa beim Wiedereintritt von Raumfahrzeugen in die Atmosphäre, in Vakuumgeräten wie Pumpen oder in miniaturisierten mikro-elektromechanischen Systemen, um nur einige Beispiele zu nennen. Die Simulation solcher verdünnter Gasströmungen ist ein aktives Forschungsfeld in der Gemeinschaft der kinetischen Gastheorie geblieben. Kinetische Modelle wie die Boltzmann-Gleichung beschreiben die Entwicklung von Gasströmungen über ein breites Spektrum von Strömungsregimen hinweg – von nahezu kontinuierlich bis hin zu stark verdünnt. Das enge Regime, das nahezu kontinuierliche Strömungen mit starkem Nicht-Gleichgewicht verbindet, wird als Übergangsregime bezeichnet. Die Simulation von Gasströmungen in diesem Regime ist besonders anspruchsvoll, da die klassischen Gleichungen der Fluiddynamik, wie die Navier-Stokes-Fourier-Gleichungen, hier nicht mehr gültig sind. Die Boltzmann-Gleichung ist zwar gültig, aber ihre direkte Lösung ist rechnerisch sehr aufwendig.

Die Momentenmethode, die ihren Ursprung in der kinetischen Gastheorie hat, stellt eine Näherung der Boltzmann-Gleichung dar. Sie bietet einen Kompromiss zwischen Genauigkeit und Rechenaufwand. Aufgrund ihrer Struktur ergibt sich eine Hierarchie von Momentenmodellen mit zunehmender Momentenordnung. Modelle niedriger Ordnung gelten als Erweiterung der klassischen Fluiddynamik. Mit steigender Momentenordnung werden die Modelle umfassender und sind theoretisch besser geeignet, komplexe Prozesse zu simulieren.

Diese Dissertation erweitert die Anwendung und Entwicklung von Momentengleichungen zur Simulation von Nicht-Gleichgewichts-Gasströmungen durch ihre Anwendung auf dreidimensionale Probleme und die Bereitstellung eines generischen numerischen Frameworks zur Lösung von Momentensystemen. Sie besteht aus zwei Hauptarbeiten.

Im ersten Teil der Dissertation demonstrieren wir die Anwendbarkeit der Momentengleichungen auf ein reales, dreidimensionales Problem – ein bedeutender Fortschritt, da bisherige Studien überwiegend ein- oder zweidimensionale Prozesse betrachteten. Diese Arbeit untersucht die dreidimensionalen regularisierten 13-Momenten-Gleichungen (R13). Sie behandelt die theoretischen Grundlagen dieser Gleichungen, ihre Variationsformulierung und die Implementierung mit FEniCS. Die Implementierung wird durch Strömungssimulationen über einer Kugel validiert, und ihre praktische Anwendbarkeit wird durch die Untersuchung des Crookes-Radiometers demonstriert. Die Studie quantifiziert die radiometrische Kraft und untersucht, wie Faktoren wie Gasdruck, Dicke und Länge der Schaufeln diese Kraft beeinflussen.

Im zweiten Teil präsentieren wir einen systematischen Ansatz zur Herleitung generischer Momentensysteme und entwickeln ein numerisches Framework auf Basis der Finite-Elemente-Methode, das in der Lage ist, Momentengleichungen beliebiger Ordnung zu lösen. Wir diskutieren die Vorteile der Formulierung höherordentlicher partieller Differentialgleichungen als Systeme erster Ordnung und untersuchen entropiestabile Randbedingungen. Die Arbeit beinhaltet außerdem eine generische Variationsformulierung und eine Implementierung der 13-Momenten-Gleichungen, begleitet von verschiedenen numerischen Simulationen zur Validierung

des Frameworks. Dieser Ansatz erweitert nicht nur den Anwendungsbereich von Momentenmodellen, sondern bietet auch einen systematischen und skalierbaren Weg zu höherordentlichen Momentengleichungen, wodurch ihre Anwendbarkeit auf ein breiteres Spektrum von Nicht-Gleichgewichts-Gasströmungsproblemen verbessert wird.

Zusammen leisten diese Beiträge einen wesentlichen Fortschritt in der Nützlichkeit und Anpassungsfähigkeit von Momentengleichungen und schlagen eine Brücke zwischen theoretischen Modellen und praktischen, rechnerisch machbaren Lösungen für die Nicht-Gleichgewichts-Gasdynamik.

Chapter

I Introduction

I.1. Motivation

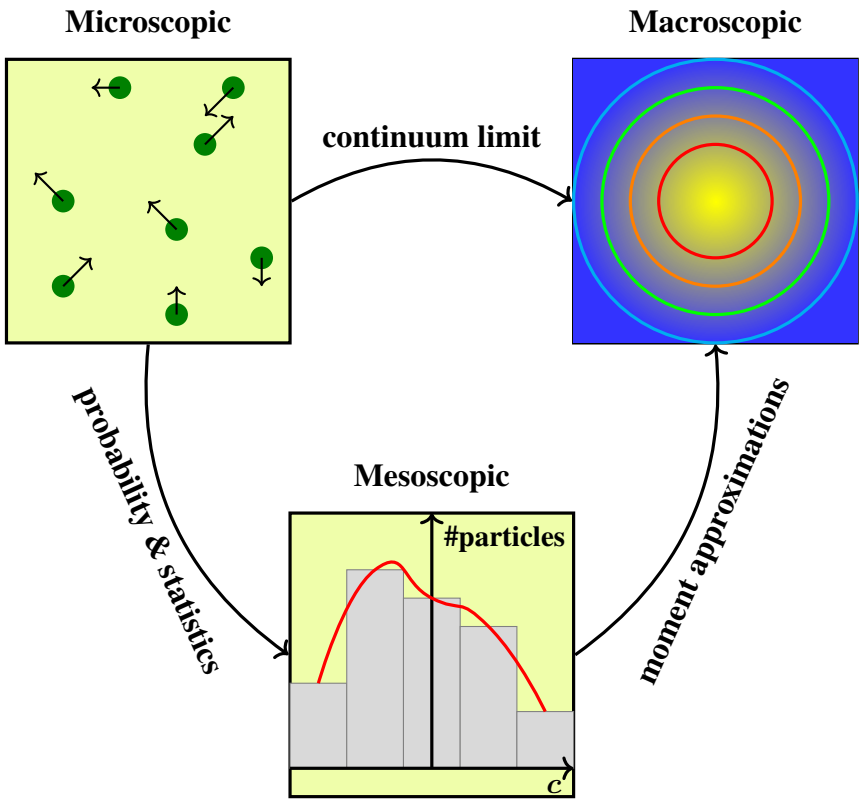


Figure I.1.: Overview of various approaches for simulating non-equilibrium gas flows.

Non-equilibrium gas flows. It is widely accepted that a gas is composed of particles that frequently collide with each other. These collisions play a crucial role in maintaining the gas's equilibrium. However, when the particle density is low, the frequency of collisions decreases, resulting in the gas being in a non-equilibrium state. Non-equilibrium gas flows occur in various real-world applications, such as during the atmospheric re-entry of space vehicles, within vacuum pumps, and in micro-electro-mechanical devices like micro-motors, among others.

The need for new models. Classical fluid dynamics models, such as the Navier-Stokes and Fourier equations, are typically used to model gases at equilibrium. However, these classical models are inadequate for non-equilibrium gases because they rely on equilibrium assumptions in their closure relations, which are not valid under non-equilibrium conditions. On the other end of the modelling spectrum, kinetic models like the Boltzmann equation provide accurate descriptions but are computationally expensive to solve, especially in three dimensions and under real-world conditions. Therefore, there is a need for new mathematical models that strike a balance between these two extremes, offering sufficient accuracy while maintaining reasonable computational costs.

Kinetic gas theory. From kinetic gas theory's perspective, gas is viewed as a collection of molecules, each endowed with a set of properties such as mass m, position $\boldsymbol{x}$, and velocity $\boldsymbol{v}$. These particles undergo random elastic collisions in an enclosed container. This simple setup serves as a classical model for simulation of ideal gases where no interactions between particles are considered. The kinetic theory of gases explains the macroscopic properties of gases, such as density ρ, pressure p, temperature θ, and as well as transport properties, such as viscosity and thermal conductivity.

Probability density function. In statistical mechanics, probability density function, $f(\boldsymbol{c}, \boldsymbol{x}, t)$, is a statistical quantity which gives the number of particles with velocity $\boldsymbol{c}$ in a given position $\boldsymbol{x}$ at a given time t. Knowing $f(\boldsymbol{c}, \boldsymbol{x}, t)$, we can calculate macroscopic properties of gas such as density, velocity, and temperature by the following relations

$$\rho = \int_{\mathbb{R}^3} m f(\boldsymbol{c}, \boldsymbol{x}, t)\, \mathrm{d}\boldsymbol{c}, \quad \rho v_i = \int_{\mathbb{R}^3} m c_i f(\boldsymbol{c}, \boldsymbol{x}, t)\, \mathrm{d}\boldsymbol{c}, \quad \frac{3}{2}\rho\theta = \int_{\mathbb{R}^3} \frac{m}{2}(c_i - v_i)^2 f(\boldsymbol{c}, \boldsymbol{x}, t)\, \mathrm{dc}. \tag{I.1}$$

Further non-equilibrium properties such as stress tensor $\boldsymbol{\sigma}$ and heat flux $\boldsymbol{q}$ can be calculated by the following relations

$$\sigma_{ij} = \int_{\mathbb{R}^3} m(c_i - v_i)(c_j - v_j) f(\boldsymbol{c}, \boldsymbol{x}, t)\, \mathrm{dc}, \quad q_i = \int_{\mathbb{R}^3} \frac{m}{2}(c_i - v_i)(c_j - v_j)^2 f(\boldsymbol{c}, \boldsymbol{x}, t)\, \mathrm{dc}. \tag{I.2}$$

Equilibrium distribution function. Under equilibrium conditions the probability density function has the following form

$$f^{(eq)} = \frac{\rho/m}{\sqrt{2\pi\theta}^3} \exp\left(-\frac{(\boldsymbol{c} - \boldsymbol{v})^2}{2\theta}\right). \tag{I.3}$$

Boltzmann equation. In 1900s, Ludwig Boltzmann derived the following equation

$$\partial_t f(\boldsymbol{c}, \boldsymbol{x}, t) + \boldsymbol{c} \cdot \nabla_{\boldsymbol{x}} f(\boldsymbol{c}, \boldsymbol{x}, t) = S(f(\boldsymbol{c}, \boldsymbol{x}, t)), \tag{I.4}$$

which, bears his name Boltzmann equation, and describes the evolution of the probability density function in space and time. Once the function $f(\boldsymbol{c}, \boldsymbol{x}, t)$ which satisfies Equation (I.4) is found, it is possible to calculate the desired macroscopic quantities of a gas as a function of

$f(\boldsymbol{c}, \boldsymbol{x}, t)$ as in Equations (I.1) and (I.2). However, solving Equation (I.4) in three-dimension is found to be very computationally intensive task as a result of high dimensionality of the problem. For example, in a three-dimensional setting, $f(\boldsymbol{c}, \boldsymbol{x}, t)$ becomes a 7-dimension function: three spatial components , three velocity components, and a time component. Despite its complexity, there exist several direct numerical solvers for the Equation (I.4), for example lattice Boltzmann method. In order to circumvent the high-dimensionality problem, there have been developments in approximation methods to the Boltzmann equation which, instead of directly solving it, proposes techniques to approximate the Boltzmann equation in order to reduce the computational cost. One of such approximation theories to Boltzmann equation is moment method.

Moment method. In the context of kinetic theory of gases, moment method is an approximation technique which aims to approximate the unknown probability density function $f(\boldsymbol{c}, \boldsymbol{x}, t)$ in Equation (I.4) by velocity moments of $f(\boldsymbol{c}, \boldsymbol{x}, t)$. This idea of solving for "moments" of a distribution function instead of the distribution function itself dates back to Maxwell's work. In 1950's Herald Grad studied polynomial-based approximation techniques in which the function $f(\boldsymbol{c}, \boldsymbol{x}, t)$ in Equation (I.4) is approximated by a family of polynomials expanded around a reference equilibrium. Following this technique one obtains a set of partial differential equations (PDEs) for the moments of the distribution function. This set of PDEs are collectively known as moment equations. So, moment equations are in essence an approximation to the Boltzmann equation.

Moment method using abstract notations. The basic principle behind moment method can be likened to the concept of best-approximation in subspaces as known from Galerkin projections. The high dimensional probability distribution function $f(\boldsymbol{c}, \boldsymbol{x}, t)$ depends on space $\boldsymbol{x}$, time t, and particle velocity $\boldsymbol{c}$. At a fixed point in space and time, the distribution function $f(\boldsymbol{c}, \boldsymbol{x}, t)$ is projected on to moment variables $u_\alpha(\boldsymbol{x}, t)$ using orthogonal polynomials, which depend only on space and time. Notationally

$$u_\alpha = \int_{\mathbb{R}^3} \Psi_\alpha(\boldsymbol{c}) f(\boldsymbol{c}) \, \mathrm{d}\boldsymbol{c}, \qquad \alpha \in I_M, \tag{I.5}$$

where I_M is a suitable set of multi-indices α which counts the polynomials and the moment variables, and Ψ_α are orthogonal polynomials with respect to a reference distribution $f^{(\mathrm{ref})}(\boldsymbol{c})$ which is assumed to be constant in space and time, and u_α are the coefficients of the following expansion

$$f(\boldsymbol{c}) \approx f^{(\mathrm{exp})}(\boldsymbol{c}) = \sum_{\beta \in I_M} u_\beta \Psi_\beta(\boldsymbol{c}) f^{(\mathrm{ref})}(\boldsymbol{c}). \tag{I.6}$$

This can be verified by plugging Equation (I.6) in Equation (I.5) and using orthogonality property of polynomials in Equation (I.8)

$$u_\alpha = \int_{\mathbb{R}^3} \Psi_\alpha(\boldsymbol{c}) \sum_{\beta \in I_M} u_\beta \Psi_\beta(\boldsymbol{c}) f^{(\mathrm{ref})}(\boldsymbol{c}) \, \mathrm{d}\boldsymbol{c} \tag{I.7}$$

$$= \sum_{\beta \in I_M} u_\beta \left(\int_{\mathbb{R}^3} \Psi_\alpha(\boldsymbol{c}) \Psi_\beta(\boldsymbol{c}) f^{(\mathrm{ref})}(\boldsymbol{c}) \, \mathrm{d}\boldsymbol{c} \right) \tag{I.8}$$

$$= \sum_{\beta \in I_M} u_\beta \delta_{\alpha\beta} \tag{I.9}$$

$$= u_\alpha. \tag{I.10}$$

We recall the Boltzmann equation in three dimensions ($i = 1, 2, 3$)

$$\frac{\partial f}{\partial t} + \sum_{i=1}^{i=3} c_i \frac{\partial f}{\partial x_i} = S(f) \tag{I.11}$$

which describes the evolution of the unknown distribution function $f(\boldsymbol{c}, \boldsymbol{x}, t)$. Projecting Equation (I.11) as in Equation (I.5) yields

$$\int_{\mathbb{R}^3} \Psi_\alpha(\boldsymbol{c}) \frac{\partial f}{\partial t} \, \mathrm{d}\boldsymbol{c} + \sum_{i=1}^{i=3} \int_{\mathbb{R}^3} c_i \Psi_\alpha(\boldsymbol{c}) \frac{\partial f}{\partial x_i} \, \mathrm{d}\boldsymbol{c} = \int_{\mathbb{R}^3} \Psi_\alpha(\boldsymbol{c}) S\left(f\right) \, \mathrm{d}\boldsymbol{c}. \tag{I.12}$$

Since the orthogonal polynomials Ψ depends only on $\boldsymbol{c}$, we can rewrite the above equation as

$$\frac{\partial}{\partial t} \left(\int_{\mathbb{R}^3} \Psi_\alpha(\boldsymbol{c}) f \, \mathrm{d}\boldsymbol{c} \right) + \sum_{i=1}^{i=3} \frac{\partial}{\partial x_i} \left(\int_{\mathbb{R}^3} c_i \Psi_\alpha(\boldsymbol{c}) f \, \mathrm{d}\boldsymbol{c} \right) = \int_{\mathbb{R}^3} \Psi_\alpha(\boldsymbol{c}) S\left(f\right) \, \mathrm{d}\boldsymbol{c}. \tag{I.13}$$

Plugging the approximation (I.6) in the place of f in (I.13) yields

$$\begin{aligned} \frac{\partial}{\partial t} \left(\sum_{\beta \in I_M} u_\beta \int_{\mathbb{R}^3} \Psi_\alpha \Psi_\beta f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{c} \right) + \sum_{i=1}^{i=3} \frac{\partial}{\partial x_i} \left(\sum_{\beta \in I_M} u_\beta \int_{\mathbb{R}^3} c_i \Psi_\alpha \Psi_\beta f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{c} \right) \\ = \int_{\mathbb{R}^3} \Psi_\alpha S \left(\sum_{\beta \in I_M} u_\beta \Psi_\beta f^{(\mathrm{ref})} \right) \, \mathrm{d}\boldsymbol{c}. \end{aligned} \tag{I.14}$$

Thus, we arrive at a generic moment system

$$\frac{\partial u_\alpha}{\partial t} + \sum_{i=1}^{i=3} \sum_{\beta \in I_M} \boldsymbol{A}^{(i)}_{\alpha\beta} \frac{\partial u_\beta}{\partial x_i} = P_\alpha(\{u_\beta\}_{\beta \in I_M}), \tag{I.15}$$

where the system matrix $\boldsymbol{A}^{(i)}_{\alpha\beta}$ in the i-th direction is defined by

$$\boldsymbol{A}^{(i)}_{\alpha\beta} := \int_{\mathbb{R}^3} c_i \Psi_\alpha \Psi_\beta f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{c} \tag{I.16}$$

and the right-hand side bilinear collision operator $S(f) = Q(f, f)$ is defined by

$$P_\alpha(\{u_\beta\}_{\beta \in I_M}) := \sum_{\beta \in I_M} \sum_{\gamma \in I_M} u_\beta u_\gamma Q^\alpha_{\beta\gamma} \tag{I.17}$$

with coefficients

$$Q^\alpha_{\beta\gamma} := \int_{\mathbb{R}^3} \Psi_\alpha Q\left(\Psi_\beta f^{(\mathrm{ref})}, \Psi_\gamma f^{(\mathrm{ref})}\right) \, \mathrm{d}\boldsymbol{c} \tag{I.18}$$

obtained from evaluating the polynomials. These collision operator projections have been recently investigated in [1, 2]. These above definitions will be made precise in Chapter III.

I.2. Organisation of the thesis

This thesis consists of four chapters.

Chapter I is organized into two sections. The first section provides a brief background on moment equations and discusses the motivation for using these equations. The second section outlines the structure of the remainder of the thesis.

The thesis is composed of two primary chapters: Chapter II and Chapter III.

Chapter II is divided into five main sections. The first section presents a brief background on the regularized 13 moment equations and describes the organization of the chapter. The second section discusses the theoretical foundation of the R13 equations and presents these equations in three dimensions. The third section details the step-by-step derivation of the variational formulation and provides brief comments on its implementation in FEniCS. In the fourth section, the implementation is validated by simulating the flow over a sphere problem. The final section applies the 3D R13 equations to investigate a real-world problem: Crookes radiometer. In this section, the working principle of the Crookes radiometer is explored, and the common hypotheses are verified using our implementation. Specifically, we aim to quantify the radiometric force, which drives the radiometer, and investigate the factors such as gas pressure, vane thickness and length that might influence this force.

Chapter III is structured into four main sections. The first section derives the first-order formulation of the Poisson system and the R13 moment system, highlighting the advantages of rewriting higher-order PDEs in the first-order form. The second section covers the derivation of generic moment systems and discusses the formulation of entropy-stable boundary conditions. In the third section, we derive a generic variational formulation and provide a concrete implementation of the 13 moment equations. The final section discusses various numerical simulations based on this implementation.

Chapter IV recapitulates the highlights and conclusions that arose from our studies. It recalls the limitations and restrictions we encountered along the way. Finally, it provides directions for future work.

Chapter II

Numerical Solution to the Steady-State 3D R13 Moment System with Application to Crookes Radiometer

II.1. Introduction

R13 moment system. The R13 moment system was devised by Struchtrup & Torrilhon in 2004 by regularising original Grad's 13-moment equations [3]. The regularisation rectified several shortcomings of the original Grad's equations and widened the applicability of 13 moment equations. Since its inception R13 moment system has been applied to various physical processes [4, 5, 6], and its properties have been studied intensively [7, 8]. Theisen *et al* discussed the derivation, linearisation, non-dimensionalisation and numerical solution of the two-dimensional R13 moment system using finite element method [9]. There are many studies on the application of R13 moment systems in one-dimensional and two-dimensional cases. However, three-dimensional simulations using R13 moment equations are scarce in the literature. In this chapter, we derive a numerical solution to the steady-state three-dimensional R13 moment equations with application to Crookes radiometer.

Scope. First, we discuss the linearised non-dimensional 3D R13 moment system. Following that, we derive the variational formulation of the system step-by-step. Then, we briefly comment on the numerical implementation in FEniCS. To validate the implementation, we simulate a flow-over-a-sphere problem for which analytical solutions exist and compare the numerical solution predicted by R13 with the analytical solution [10]. After validation, we study the Crookes radiometer. Despite its simplicity, the Crookes radiometer is a robust device that serves as an excellent introduction to the world of kinetic gas theory. We conduct a numerical study on the Crookes radiometer using moment equations and investigate the effect of gas pressure, vane thickness, and vane geometry on the radiometric force.

II.2. Discussion on Theory

3D linearised non-dimensional R13 equations. We are interested in studying the steady-state R13 equations. Therefore, we apply the linearisation and non-dimensionalisation techniques discussed in [9] to the original R13 equations [3], resulting in the following linearised non-dimensional 3D R13 equations:

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = \dot{m}, \tag{II.1}$$

$$\boldsymbol{\nabla} p + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{b}, \tag{II.2}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} + \boldsymbol{\nabla} \cdot \boldsymbol{s} = r, \tag{II.3}$$

$$\frac{4}{5}(\boldsymbol{\nabla} \boldsymbol{s})_{\text{stf}} + 2(\boldsymbol{\nabla} \boldsymbol{u})_{\text{stf}} + \boldsymbol{\nabla} \cdot \boldsymbol{m} = -\frac{1}{\text{Kn}} \boldsymbol{\sigma}, \tag{II.4}$$

$$\frac{5}{2} \boldsymbol{\nabla} \theta + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \frac{1}{2} \boldsymbol{\nabla} \cdot \boldsymbol{R} + \frac{1}{6} \boldsymbol{\nabla} \Delta = -\frac{1}{\text{Kn}} \frac{2}{3} \boldsymbol{s}, \tag{II.5}$$

$$\boldsymbol{m} = -2\text{Kn}(\boldsymbol{\nabla} \boldsymbol{\sigma})_{\text{stf}}, \tag{II.6}$$

$$\boldsymbol{R} = -\frac{24}{5}\text{Kn}(\boldsymbol{\nabla} \boldsymbol{s})_{\text{stf}}, \tag{II.7}$$

$$\Delta = -12\text{Kn}(\boldsymbol{\nabla} \cdot \boldsymbol{s}). \tag{II.8}$$

The set of variables $\{\theta, \boldsymbol{s}, p, \boldsymbol{u}, \boldsymbol{\sigma}\}$ represents the quantities of interest, such as temperature, heat flux, pressure, velocity, and stress, respectively. The set $\{\boldsymbol{m}, \boldsymbol{R}, \Delta\}$ includes higher-order moments that do not carry physical meaning but are crucial for the closure of the moment system. The set $\{\dot{m}, \boldsymbol{b}, r\}$ denotes external source terms involved in the problem, such as mass source, body force, and heat source, respectively. Equations (II.1) - (II.3) represent the typical conservation laws for mass, momentum, and energy. Equations (II.4) and (II.5) describe the evolution equations for stress and heat flux, respectively. Equations (II.6) - (II.8) are known as the closure relations.

3D linearised non-dimensional boundary conditions. To prescribe values at the boundaries of a computational domain, boundary conditions are needed. The linearised boundary conditions for a generalised moment system is discussed in [11] and for the R13 equations is discussed in [12] which is presented below:

$$(u_n - u_n^w) = 0, \tag{II.9}$$

$$\sigma_{nt_i} = \tilde{\chi} \left((u_{t_i} - u_{t_i}^w) + \frac{1}{5} s_{t_i} + m_{nnt_i} \right), \qquad i = 1, 2 \tag{II.10}$$

$$R_{nt_i} = \tilde{\chi} \left(-(u_{t_i} - u_{t_i}^w) + \frac{11}{5} s_{t_i} - m_{nnt_i} \right), \qquad i = 1, 2 \tag{II.11}$$

$$s_n = \tilde{\chi} \left(2(\theta - \theta^w) + \frac{1}{2}\sigma_{nn} + \frac{2}{5} R_{nn} + \frac{2}{15}\Delta \right), \tag{II.12}$$

$$m_{nnn} = \tilde{\chi} \left(-\frac{2}{5}(\theta - \theta^w) + \frac{7}{5}\sigma_{nn} - \frac{2}{25} R_{nn} - \frac{2}{75}\Delta \right), \tag{II.13}$$

$$\left(m_{nt_1t_1} + \frac{1}{2} m_{nnn} \right) = \tilde{\chi} \left(\frac{1}{2}\sigma_{nn} + \sigma_{t_1t_1} \right), \tag{II.14}$$

$$m_{nt_1t_2} = \tilde{\chi} \sigma_{t_1t_2}. \tag{II.15}$$

The index set $\{n, t_1, t_2\}$, used as subscripts in Equations (II.9) - (II.15), forms a local (normal-tangent) coordinate system aligned with the boundary of the computational domain. The symbols n, t_1, and t_2 represent the outer normal and two tangents at the boundary, respectively. The symbol $\tilde{\chi}$ denotes the modified accommodation coefficient, given by $\tilde{\chi} = \sqrt{\frac{2}{\pi}} \left(\frac{\chi}{2-\chi} \right)$.

Equation (II.9) implies that no exchange of matter occurs at the boundary, meaning that the component of the velocity $\boldsymbol{u}$ in the $\boldsymbol{n}$-direction is zero. A more useful formulation to accommodate inflow and outflow conditions at the boundaries is discussed in [13]. Building on these ideas, Equation (II.9) is modified to

$$(u_n - u_n^w) = \epsilon^w \tilde{\chi} \left((p - p^w) + \sigma_{nn} \right). \tag{II.16}$$

The in-/out-flow boundary condition requires introducing an artificial pressure p^w, an artificial velocity u_n^w, and a velocity prescription coefficient ϵ^w at the interface, mimicking a wall. For the case $\epsilon^w = 0$ and hence $u_n^w = 0$, Equation (II.16) reduces to the standard boundary condition given by Equation (II.9). In the limit as $\epsilon^w \to \infty$, the boundary condition described by Equation (II.16) implies that the total pressure at the interface is $p^w = p + \sigma_{nn}$[9].

II.3. Discussion on Numerics

II.3.1. Numerical method

A numerical solution for the two-dimensional R13 moment system using the finite element method is developed in [9]. The extension to the three-dimensional case is relatively straightforward. For completeness, the derivation of the 3D weak formulation is provided in this section. The procedure for deriving the weak formulation of the three-dimensional R13 moment system closely follows the work of Theisen *et al.* [9].

Notation. Let us define a trial function vector $\boldsymbol{\mathcal{U}}$ as

$$\boldsymbol{\mathcal{U}} := (\boldsymbol{s}, \theta, \boldsymbol{\sigma}, \boldsymbol{u}, p), \tag{II.17}$$

which includes the variables for heat flux (vector), temperature (scalar), stress (tensor), velocity (vector), and pressure (scalar), respectively. Correspondingly, the test function vector is given by

$$\boldsymbol{\mathcal{V}} := (\boldsymbol{r}, \kappa, \boldsymbol{\psi}, \boldsymbol{v}, q). \tag{II.18}$$

We choose a suitable Sobolev function space $\mathbb{V}_*$ for each test function $*$ in the vector $\boldsymbol{\mathcal{U}}$. We employ the Galerkin method, which uses the same function spaces for both test and trial functions. Thus, $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{V}}$ are elements of the space $\mathbb{H} := \mathbb{V}_{\boldsymbol{s}} \times \mathbb{V}_\theta \times \mathbb{V}_{\boldsymbol{\sigma}} \times \mathbb{V}_{\boldsymbol{u}} \times \mathbb{V}_p$.

Variational formulation. The variational formulation of the R13 moment system is as follows: Find $\boldsymbol{\mathcal{U}} \in \mathbb{H}$ such that

$$\mathcal{A}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) = \mathcal{L}(\boldsymbol{\mathcal{V}}), \qquad \forall \boldsymbol{\mathcal{V}} \in \mathbb{H}, \tag{II.19}$$

where

$$\begin{aligned} \mathcal{A}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) := {} & a(\boldsymbol{s}, \boldsymbol{r}) + b(\boldsymbol{s}, \kappa) - b(\theta, \boldsymbol{r}) + c(\boldsymbol{s}, \boldsymbol{\psi}) - c(\boldsymbol{\sigma}, \boldsymbol{r}) + d(\boldsymbol{\sigma}, \boldsymbol{\psi}) + e(\boldsymbol{u}, \boldsymbol{\psi}) - e(\boldsymbol{\sigma}, \boldsymbol{v}) \\ & + f(p, \boldsymbol{\psi}) - f(\boldsymbol{\sigma}, q) + g(p, \boldsymbol{v}) - g(q, \boldsymbol{u}) + h(p, q), \end{aligned} \tag{II.20}$$

$$\mathcal{L}(\boldsymbol{\mathcal{V}}) := l_1(\boldsymbol{r}) + l_2(\kappa) + l_3(\boldsymbol{\psi}) + l_4(\boldsymbol{v}) + l_5(q). \tag{II.21}$$

The bilinear *functional* $\mathcal{A}$ in Equation (II.20) is expressed in terms of several *sub-functionals* $a, b, \ldots, h$. Similarly, the linear functional $\mathcal{L}$ in Equation (II.21) is composed of sub-functionals $l_1, l_2, \ldots, l_5$. The exact expressions for these sub-functionals are derived step-by-step in Appendix A.1. For brevity, we provide the final expressions for these sub-functionals below.

$$\begin{aligned} a(\boldsymbol{s}, \boldsymbol{r}) := & \frac{1}{\mathrm{Kn}} \frac{4}{15} \int_\Omega \boldsymbol{s} \cdot \boldsymbol{r} \, \mathrm{d}\boldsymbol{x} + \frac{24}{25} \mathrm{Kn} \int_\Omega (\boldsymbol{\nabla} \boldsymbol{s})_{\mathrm{stf}} : \boldsymbol{\nabla} \boldsymbol{r} \, \mathrm{d}\boldsymbol{x} \\ & + \frac{12}{15} \mathrm{Kn} \int_\Omega (\boldsymbol{\nabla} \cdot \boldsymbol{s})(\boldsymbol{\nabla} \cdot \boldsymbol{r}) \, \mathrm{d}\boldsymbol{x} + \int_\Gamma \left(\frac{1}{2} \frac{1}{\tilde{\chi}} s_n r_n + \frac{12}{25} \tilde{\chi} s_{t_1} r_{t_1} + \frac{2}{5} \tilde{\chi} s_{t_2} r_{t_2} \right) \mathrm{d}\boldsymbol{l}, \end{aligned} \tag{II.22}$$

$$b(\theta, \boldsymbol{r}) := \int_\Omega \theta (\nabla \cdot \boldsymbol{r}) \, \mathrm{d}\boldsymbol{x}, \tag{II.23}$$

$$c(\boldsymbol{\sigma}, \boldsymbol{r}) := \frac{2}{5} \int_\Omega \boldsymbol{\sigma} : \boldsymbol{\nabla} \boldsymbol{r} \, \mathrm{d}\boldsymbol{x} - \int_\Gamma \left(\frac{3}{20} \sigma_{nn} r_n + \frac{1}{5} \sigma_{nt_1} r_{t_1} + \frac{1}{5} \sigma_{nt_2} r_{t_2} \right) \mathrm{d}\boldsymbol{l}, \tag{II.24}$$

$$\begin{aligned} d(\boldsymbol{\sigma}, \boldsymbol{\psi}) := & \, \mathrm{Kn} \int_\Omega (\boldsymbol{\nabla} \boldsymbol{\sigma})_{\mathrm{stf}} \vdots \boldsymbol{\nabla} \boldsymbol{\psi}) \, \mathrm{d}\boldsymbol{x} + \frac{1}{2} \frac{1}{\mathrm{Kn}} \int_\Omega \boldsymbol{\sigma} : \boldsymbol{\psi} \, \mathrm{d}\boldsymbol{x} \\ & + \int_\Gamma \left(\tilde{\chi} \left(\frac{9}{8} \sigma_{nn} \psi_{nn} + \sigma_{t_1 t_2} \psi_{t_1 t_2} \right) + \frac{1}{\tilde{\chi}} \left(\sigma_{nt_1} \psi_{nt_1} + \sigma_{nt_2} \psi_{nt_2} \right) \right) \mathrm{d}\boldsymbol{l} \\ & + \tilde{\chi} \int_\Omega \left(\frac{1}{2} \sigma_{nn} + \sigma_{t_1 t_1} \right) \left(\frac{1}{2} \psi_{nn} + \psi_{t_1 t_1} \right) \mathrm{d}\boldsymbol{l}, \end{aligned} \tag{II.25}$$

$$e(\boldsymbol{u}, \boldsymbol{\psi}) := \int_\Omega \boldsymbol{u} \cdot (\boldsymbol{\nabla} \cdot \boldsymbol{\psi}) \, \mathrm{d}\boldsymbol{x}, \tag{II.26}$$

$$f(p, \boldsymbol{\psi}) := \epsilon^w \tilde{\chi} \int_\Gamma p \psi_{nn} \, \mathrm{d}\boldsymbol{l}, \tag{II.27}$$

$$g(p, \boldsymbol{v}) := \int_\Omega \boldsymbol{\nabla} p \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}, \tag{II.28}$$

$$h(p, q) := \epsilon^w \tilde{\chi} \int_\Gamma p q \, \mathrm{d}\boldsymbol{l}, \tag{II.29}$$

$$l_1(\boldsymbol{r}) := - \int_\Gamma \theta^w r_n \, \mathrm{d}\boldsymbol{l}, \tag{II.30}$$

$$l_2(\kappa) := - \int_\Omega (r - \dot{m}) \kappa \, \mathrm{d}\boldsymbol{x}, \tag{II.31}$$

$$l_3(\boldsymbol{\psi}) := - \int_\Gamma (u_n^w \psi_{nn} + u_{t_1}^w \psi_{nt_1} + u_{t_1}^w \psi_{nt_2} - \epsilon^w \tilde{\chi} p^w \psi_{nn}) \, \mathrm{d}\boldsymbol{l}, \tag{II.32}$$

$$l_4(\boldsymbol{v}) := \int_\Omega \boldsymbol{b} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}, \tag{II.33}$$

$$l_5(q) := \int_\Omega \dot{m} q \, \mathrm{d}\boldsymbol{x} - \int_\Gamma (u_n^w - \epsilon^w \tilde{\chi} p^w) q \, \mathrm{d}\boldsymbol{l}. \tag{II.34}$$

Stabilisation. The 2D finite element discretization of the R13 system discussed in [9] shows that the resulting algebraic system has a saddle point structure and therefore requires stabilization. To address the stabilization-related issues, a residual-based CIP stabilization technique, developed in [14], is proposed in [9].

We note that the arguments presented in [9] also apply to the 3D finite element discretization of the R13 system. To address the stabilization-related issues, this work employs the residual-based Galerkin Least Squares (GLS) stabilization method. The stabilization term $\mathcal{S}$ is defined as

$$\mathcal{S}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) := \mathcal{S}_\theta(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) + \mathcal{S}_{\boldsymbol{s}}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) + \mathcal{S}_p(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) + \mathcal{S}_{\boldsymbol{u}}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) + \mathcal{S}_{\boldsymbol{\sigma}}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}), \tag{II.35}$$

where,

$$\mathcal{S}_\theta(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) := \tau_\theta \, h_\theta \int_\Omega (\nabla \cdot \boldsymbol{s} + \nabla \cdot \boldsymbol{u} - r) \cdot (\nabla \cdot \boldsymbol{r} + \nabla \cdot \boldsymbol{v}) \, \mathrm{d}\boldsymbol{x} \tag{II.36}$$

$$\begin{aligned} \mathcal{S}_{\boldsymbol{s}}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) := \tau_{\boldsymbol{s}} \, h_{\boldsymbol{s}} \int_\Omega & \left(\frac{1}{5}\nabla\theta - \frac{12}{5}\mathrm{Kn}(\nabla \cdot (\nabla \boldsymbol{u})_{\mathrm{stf}}) - 2\mathrm{Kn}\nabla(\nabla \cdot \boldsymbol{s}) + \frac{1}{\mathrm{Kn}}\frac{2}{3}\boldsymbol{s} \right) \\ & \cdot \left(\frac{1}{5}\nabla\kappa - \frac{12}{5}\mathrm{Kn}(\nabla \cdot (\nabla \boldsymbol{v})_{\mathrm{stf}}) - 2\mathrm{Kn}\nabla(\nabla \cdot \boldsymbol{r}) + \frac{1}{\mathrm{Kn}}\frac{2}{3}\boldsymbol{r} \right) \, \mathrm{d}\boldsymbol{x} \end{aligned} \tag{II.37}$$

$$\mathcal{S}_p(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) := \tau_p \, h_p \int_\Omega (\nabla \cdot \boldsymbol{v}) \cdot ((\nabla \cdot \boldsymbol{u}) - \dot{m}) \, \mathrm{d}\boldsymbol{x} \tag{II.38}$$

$$\mathcal{S}_{\boldsymbol{u}}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) := \tau_{\boldsymbol{u}} \, h_{\boldsymbol{u}} \int_\Omega (\nabla q + (\nabla \cdot \boldsymbol{\psi})) \cdot (\nabla p + (\nabla \cdot \boldsymbol{\sigma}) - \boldsymbol{b}) \, \mathrm{d}\boldsymbol{x} \tag{II.39}$$

$$\begin{aligned} \mathcal{S}_{\boldsymbol{\sigma}}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) := \tau_{\boldsymbol{\sigma}} \, h_{\boldsymbol{\sigma}} \int_\Omega & \left(\frac{4}{5}(\nabla \boldsymbol{r})_{\mathrm{stf}} + 2(\nabla \boldsymbol{v})_{\mathrm{stf}} - 2\mathrm{Kn}\nabla \cdot (\nabla \boldsymbol{\psi})_{\mathrm{stf}} + \frac{1}{\mathrm{Kn}}\boldsymbol{\psi} \right) \\ & \because \left(\frac{4}{5}(\nabla \boldsymbol{s})_{\mathrm{stf}} + 2(\nabla \boldsymbol{u})_{\mathrm{stf}} - 2\mathrm{Kn}\nabla \cdot (\nabla \boldsymbol{\sigma})_{\mathrm{stf}} + \frac{1}{\mathrm{Kn}}\boldsymbol{\sigma} \right) \, \mathrm{d}\boldsymbol{x} \end{aligned} \tag{II.40}$$

The stabilization parameters $\tau_{(\bullet)}$ and the mesh-dependent length scales $h_{(\bullet)}$ — typically defined by the element size — used in Equations (II.36)–(II.40) are introduced to ensure numerical stability and to alleviate violations of the LBB condition. These terms act as artificial diffusivities tailored to each field variable.

In the absence of a universally accepted theoretical framework for determining optimal stabilization parameters in this context, we adopt an empirical approach. Parameter values were selected based on numerical experimentation for the problem at hand to ensure stable and convergent behaviour across a range of mesh resolutions. However, we note that these parameters were not systematically validated across different problem configurations, and the chosen values may not be optimal for all the moment models. Further investigation is required to better understand the sensitivity of stabilization parameters to model choice and problem setup.

Stabilised weak formulation. The stabilised weak formulation is then expressed as

$$\mathcal{A}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) + \mathcal{S}(\boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{V}}) = \mathcal{L}(\boldsymbol{\mathcal{V}}). \tag{II.41}$$

II.3.2. Notes on implementation

FEniCSR13. FEniCSR13, a Python program originally developed by Theisen *et al.* [9], solves the 2D R13 moment system. It is built on FEniCS, a computational environment in C++ and Python, which provides a direct correspondence between mathematical formulation and computational implementation. Extensions to the 3D case involve a few modifications to the original program, which were performed by Srinivasan and Christhuraj. Details for the implementation of Equation (II.41) and instructions for running the FEniCSR13 software can be found in [15]. Below, we highlight a few important modifications.

3D weak formulation snippets. The code snippet in Listing (1) shows the implementation of the functional a in Equation (II.22). Thanks to FEniCS, it is possible to establish a one-

```
def a(s, r):
  return sum([(
    + 24 / 25 * regs[reg]["kn"] * df.inner(
    df.sym(df.grad(s)), df.sym(df.grad(r)))
    - 24 / 75 * regs[reg]["kn"] * df.div(s) * df.div(r)
    + 4 / 5 * cpl * regs[reg]["kn"] * df.div(s) * df.div(r)
    + 4 / 15 * (1 / regs[reg]["kn"]) * df.inner(s, r)
    ) * df.dx(reg) for reg in regs.keys()
    ]) + sum([( + 1 / (2 * bcs[bc]["chi_tilde"]) * n(s) * n(r)
    + 11 / 25 * bcs[bc]["chi_tilde"] * t1(s) * t1(r)
    + cpl * 1 / 25 * bcs[bc]["chi_tilde"] * t1(s) * t1(r)
    + 11 / 25 * bcs[bc]["chi_tilde"] * t2(s) * t2(r)
    + cpl * 1 / 25 * bcs[bc]["chi_tilde"] * t2(s) * t2(r)
    ) * df.ds(bc) for bc in bcs.keys()]
  )
```

Listing 1: Implementation of the functional a in the weak formulation

to-one correspondence between the mathematical formulation and its implementation. Key terms such as `df.inner`, `df.sym`, and `df.grad` represent inner product, symmetric tensor, and gradient, respectively, and are provided by the FEniCS library. The three vectors `n`, `t1`, and `t2` denote the normal and tangent vectors, and they are defined below.

Similarly, Listing (2) shows the implementation of the functional d in the weak formulation given in Equation (II.25). Despite its extensive application interface, FEniCS does not provide built-in functions for calculating the gradient of trace-free tensors. Therefore, the functions `to.grad3dOf2` and `to.gen3DTFdim2`, which compute the gradient of a 3D vector and a trace-free tensor, respectively, are implemented externally in the tensor operations (`to`) module [9].

3D Normal-Tangent Coordinate System. In two dimensions, finding a unique tangential vector given a normal vector is straightforward. However, in three dimensions, identifying unique tangential vectors given a normal vector is not unique due to multiple possible solutions. All boundary equations can be expressed in terms of vector components that are invariant with respect to the chosen coordinate system. Therefore, any set of orthogonal unit vectors within

```
def d(si, ps):
  return sum([(
    + regs[reg]["kn"] * df.inner(
    to.stf3d3(to.grad3dOf2(to.gen3DTFdim2(si), nsd)),
    to.stf3d3(to.grad3dOf2(to.gen3DTFdim2(ps), nsd)))
    + (1 / (2 * regs[reg]["kn"])) * df.inner(
    to.gen3DTFdim2(si), to.gen3DTFdim2(ps))
    ) * df.dx(reg) for reg in regs.keys()]) + sum([(
    + bcs[bc]["chi_tilde"] * 21 / 20 * nn(si) * nn(ps)
    + bcs[bc]["chi_tilde"] * cpl * 3 / 40 * nn(si) * nn(ps)
    + bcs[bc]["chi_tilde"] * ( (t1t1(si) + (1 / 2) * nn(si)) *
    (t1t1(ps) + (1 / 2) * nn(ps)))
    + (1 / bcs[bc]["chi_tilde"]) * nt1(si) * nt1(ps)
    + (1 / bcs[bc]["chi_tilde"]) * nt2(si) * nt2(ps)
    + bcs[bc]["epsilon_w"] * bcs[bc]["chi_tilde"] * nn(si) * nn(ps)
    + bcs[bc]["chi_tilde"] * t1t2(si) * t1t2(ps)
    ) * df.ds(bc) for bc in bcs.keys()]
  )
```

Listing 2: Implementation of the functional d in the weak formulation

the tangent plane is admissible. Listing (3) shows the implementation of the calculation of normal and tangential vectors at the boundary. We identify the normal vector at the boundary. Then we compute a vector orthogonal to the normal vector using the cross product. Finally, a second orthogonal vector is determined by taking the cross product of the initial two vectors.

```
def normal(self):
    v1 = df.as_vector([1.0, 0.0, 0.0])
    v2 = df.as_vector([0.0, 1.0, 0.0])
    n_vec = df.FacetNormal(self.mesh)
    t_vec1 = df.conditional(
        df.gt(abs(n_vec[0]), np.finfo(float).eps),
        df.cross(n_vec, v2 / df.sqrt(n_vec[0] ** 2 + n_vec[2] ** 2)),
        df.cross(n_vec, v1 / df.sqrt(n_vec[1] ** 2 + n_vec[2] ** 2))
    )
    t_vec2 = df.cross(n_vec, t_vec1)
    return n_vec, t_vec1, t_vec2
```

Listing 3: Implementation of the calculation of the normal-tangent vector at the boundary

II.4. Validation of Numerical Framework

Goal. To validate the FEniCSR13 numerical framework [16], we consider the flow-over-a-sphere problem, for which analytical solutions are known [10]. We compare the numerical solutions obtained by FEniCSR13 with these known analytical solutions and perform a convergence study through mesh refinements.

Description of computational domain. The flow-over-a-sphere problem consists of two concentric spheres of different radii situated at the origin. The setup is illustrated in Figure II.1. The origin, denoted as O, located at $(0, 0, 0)$. The radius of the inner sphere, denoted by $\overrightarrow{OC}$, is 0.5 unit and the radius of the outer sphere, denoted by $\overrightarrow{OR}$, is 2 units.

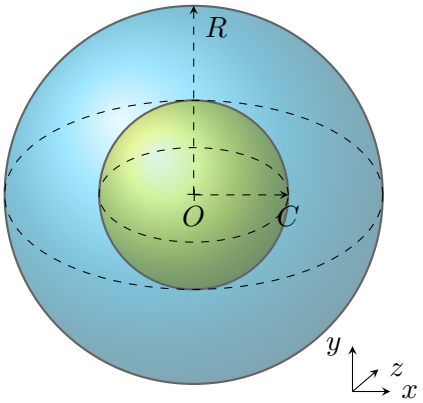


Figure II.1.: Computational domain of flow-over-a-sphere

Simulation Data. We consider the scenario where gas flows axially along the z direction over the inner sphere. The following parameters are chosen for the simulations to produce spherically symmetric solutions: Kn $= 0.005$ or Kn $= 0.2$ and $\tilde{\chi} = 1.0$. The following boundary conditions are used in the following convergence study.

inner sphere	outer sphere
$p = 0$	$p = -0.027\frac{z}{\sqrt{x^2+y^2+z^2}}$
$\theta = 1$	$\theta = 2$
$v_x = 0$	$v_x = 0$
$v_y = 0$	$v_y = 0$
$v_z = 0$	$v_z = 1$
$\epsilon^{(w)} = 10^{-2}$	$\epsilon^{(w)} = 10^{2}$

Simulation results. In Figures II.2 and II.3, the convergence plots for the R13 moment variables are presented for Knudsen numbers 0.005 and 0.2, respectively. Each plot shows the normalised L_2 error norm and the normalised l_∞ error norm. The convergence rate depends on several factors, including the mesh size and geometry, stabilization parameters, and the polynomial order of the finite element. By mesh size, we refer to the minimum edge length in the mesh. In 3D simulations, it is computationally expensive to simulate on a very fine mesh. To accurately represent the curved boundaries at the inner and outer spheres, the mesh must be refined near the boundaries. Furthermore, we used the simplest finite element, namely the P1 element, to approximate all variables. For the chosen stabilization parameters, we observe that all variables converge between first and second order. For the P1 element, the theoretical maximum convergence rate is expected to be of second order. As at Kn = 0.2 the non-equilibrium

quantities $\boldsymbol{s}$ and $\boldsymbol{\sigma}$ play a significant role compared to Kn = 0.005, the R13 numerical solution better approximates the analytical solution. Hence, an

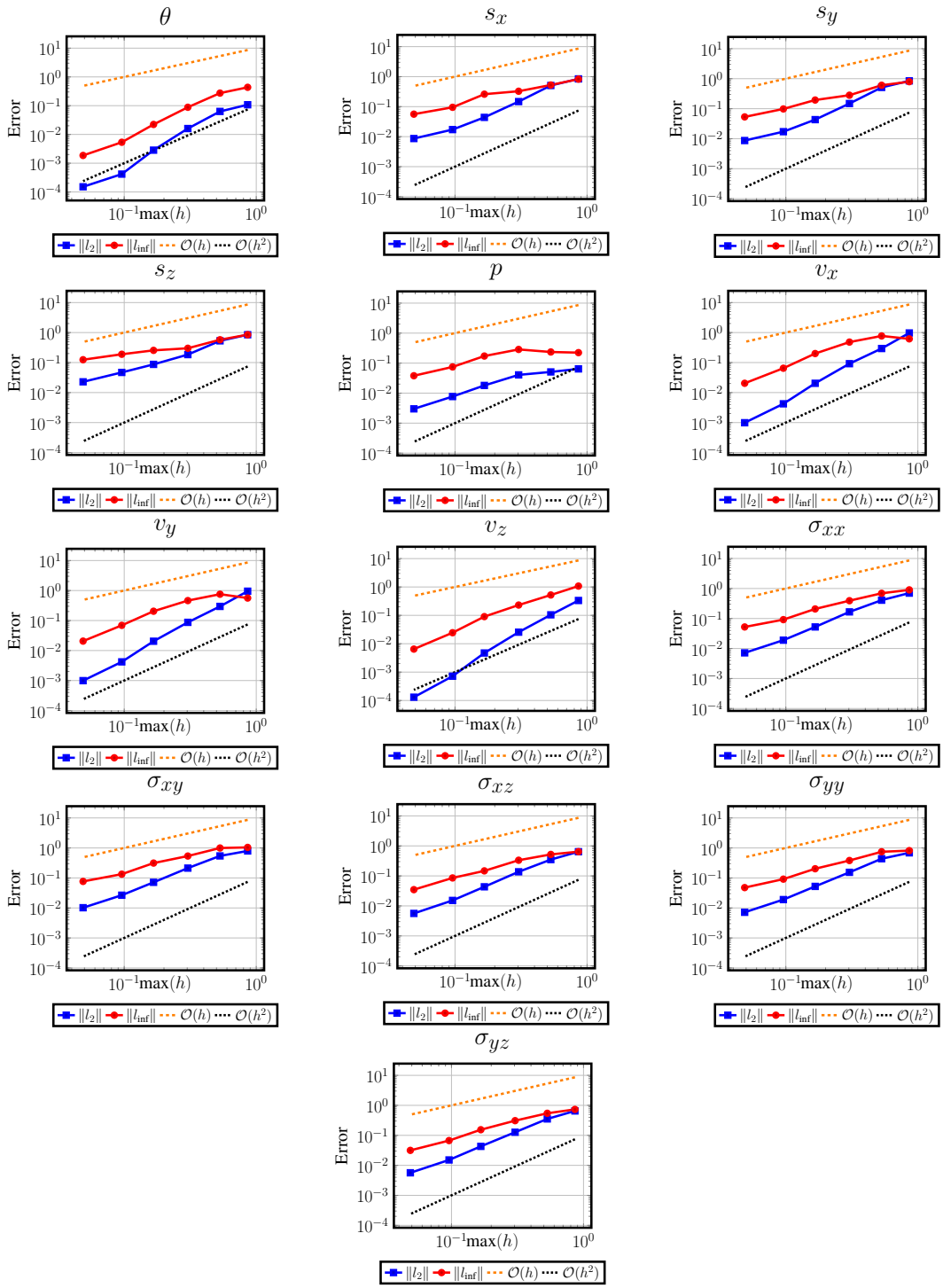


Figure II.2.: Convergence of R13 moment variables for Kn = 0.005

improved convergence rate of the higher-order moments $\boldsymbol{s}$ and $\boldsymbol{\sigma}$ is observed for Kn = 0.2. To counter the LBB conditions, we applied GLS stabilization as discussed in the previous section.

However, the stabilization parameters could be further optimized to achieve better convergence rates. The data used for generating the convergence plots is included in Appendix A.2.

This flow-over-a-sphere test problem is used to verify the correctness of our implementation. The convergence plots indicate that the numerical solutions predicted by the FEniCSR13 implementation converge to the analytical solutions asymptotically.

;

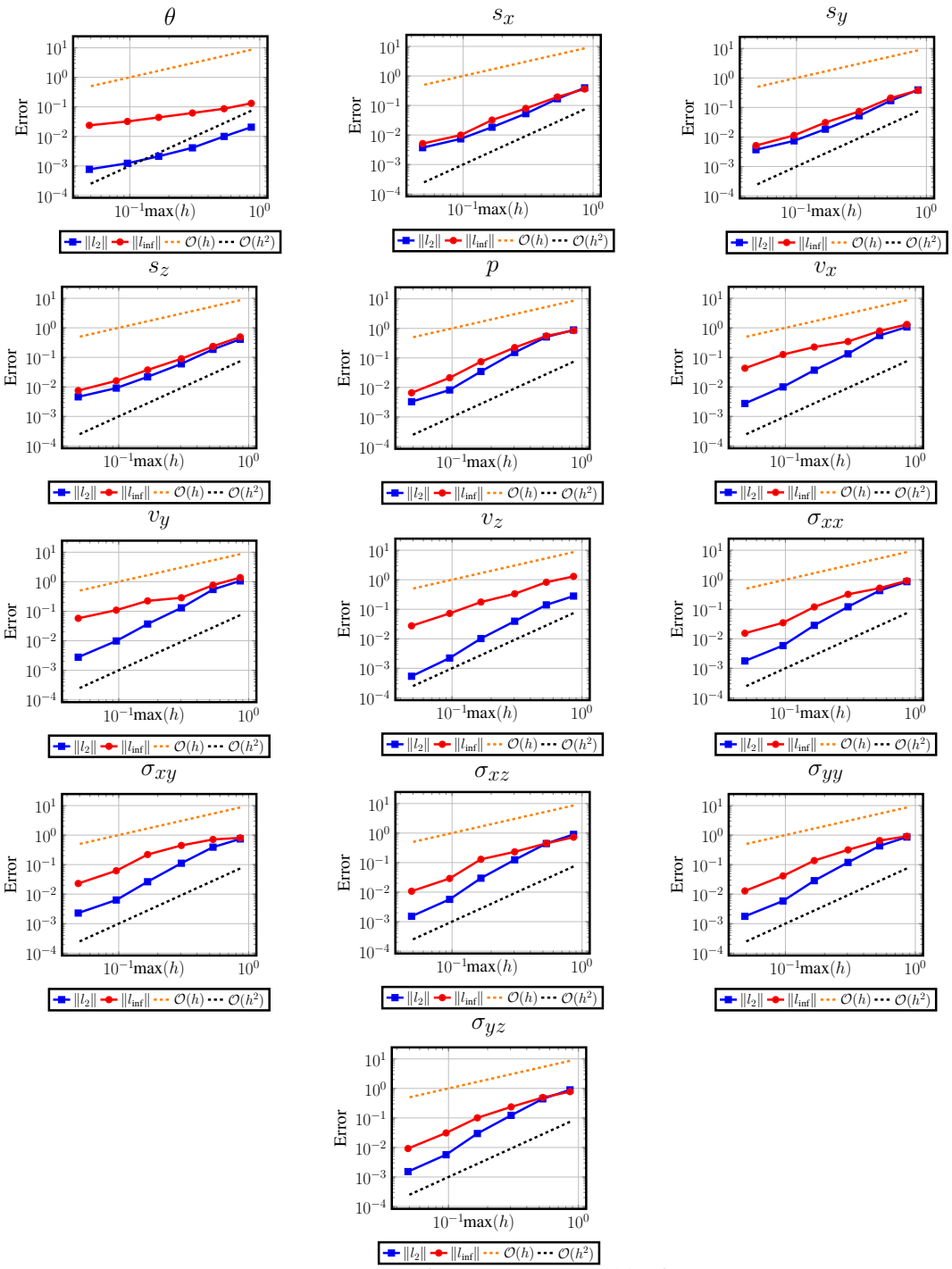


Figure II.3.: Convergence of R13 moment variables for Kn = 0.2

II.5. Application: Crookes Radiometer

II.5.1. Motivation

Crookes radiometer. Crookes radiometer is a simple device that operates on the principle of radiometric phenomena. This device was invented by Sir William Crookes in the 19th century while investigating the measurement of molecular weight of a chemical element, namely Thallium [17]. Nowadays, Crookes radiometer can be found in any optician's shops and can be bought inexpensively. While momentarily providing joy for children and adults alike, it serves as an excellent introduction to the world of kinetic theory of gases.

Construction. Crookes radiometer consists of a set of four vanes mounted on a rotor spindle which is encased in a glass casing. Each vane has two sides: a light-coloured side, often denoted as the white side, and a dark-coloured side, often denoted as the black side. The white side of the vane is coated with highly photo-reflective material while the darker side of the vane is coated with highly photo-absorptive material. The entire set-up inside the glass housing is kept in a semi-vacuum condition. When exposed to an intense light source, the vanes in the Crookes radiometer start rotating with white side leading while black side trailing in the direction of rotation.

Working principle. Since its discovery, the working principle of the Crookes radiometer has long been a source of debate for researchers. There have been several false hypotheses, incorrect explanations and a few correct understandings along the way.

For some time, it was widely believed that the rotation happened due to the pressure of light. However, further experiments have dispelled this view, citing the following reasons: Firstly, despite exposure to a light source in the absence of air molecules inside the glass chamber—complete vacuum—there was no rotation observed. Secondly, according to this view, the white side must be trailing while the black side must be leading, which contradicts observation.

There was another partially correct theory proposed, which argues that the rotation happens because gas molecules hitting the warmer side of the vane will pick up some heat and impart it with increased speed. This energy exchange leads to a minute pressure exerted on the vane. However, this explanation was challenged by a counter-argument, which asserts that while recoiling molecules moving with high velocity will produce high force, they also hinder other molecules from hitting the wall. Years later, Einstein showed that the forces at the edges of the vane do not cancel each other out. There would be some net force acting on the edges of the vane.

The currently widely accepted theory of the Crookes radiometer was formulated by Osborne Reynolds, who theorised that thermal transpiration was the cause of rotation [18]. As the vanes are relatively thin with respect to their length, gas molecules move from the cooler side to the warmer side over the space past the edges of the vane. This causes pressure on the hotter side to increase, and eventually, this pressure difference leads to rotation. From a molecular point of view, vanes move due to the tangential force of the rarefied gas colliding differently with the edges of the vane between the hot and cold sides.

Radiometric force quantification. Very early in the Crookes radiometer's history several attempts were made to quantify the magnitude of the radiometric force, mostly through experiments. It was found that the magnitude of the radiometric force was very small and was believed to be of no use to real-world applications. With the progress in technology Crookes radiometer has received renewed interest [19].

Existing models. Several numerical studies have been conducted on the Crookes radiometer. A few of them are as follows: Yu Anikin *et al.* investigated the radiometric force by directly solving the Boltzmann equation on a two-dimensional cross-shaped radiometer [20]. They studied the radiometric force over a wide range of Knudsen numbers and temperatures. Gimelshein *et al.* examined the impact of vane thickness and edge shape on the radiometric force using the ES-BGK kinetic equation in a two-dimensional setting[19]. Dechristé *et al.* simulated the classical three-dimensional Crookes radiometer using the BGK equation. Li-Hsin Han *et al.* studied a Crookes radiometer-inspired light-powered micromotor using DSMC simulations[21].

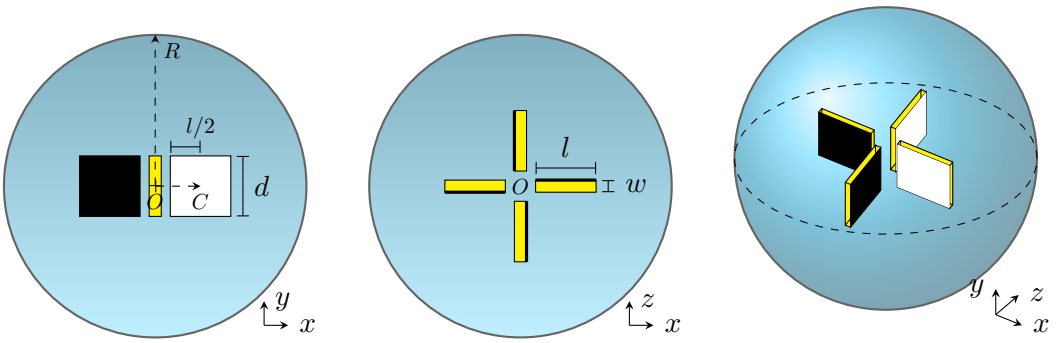
In this section, we discuss the results of three investigations: the effects of gas pressure, vane thickness, and vane length on the radiometric force.

II.5.2. Effect of gas pressure on the radiometric force.

Goal. The goal of the following study is to find the relation between gas pressure p and radiometric force. Knudsen number (Kn) is the ratio of the distance travelled by gas molecules between successive collisions λ and the characteristic length of the problem w. As such we note that Kn is inversely proportional to the gas pressure p.

General description of computational domain. A simplified computational domain of a 3D Crookes radiometer consists of the essential parts of a radiometer: the vanes and the encasing glass wall. In this study, we consider three different vane geometries: rectangle, circle, diamond. But we restrict ourselves to a radiometer with four vanes, housed symmetrically inside a spherical glass wall. In the following paragraphs, we discuss each radiometer's geometry in detail.

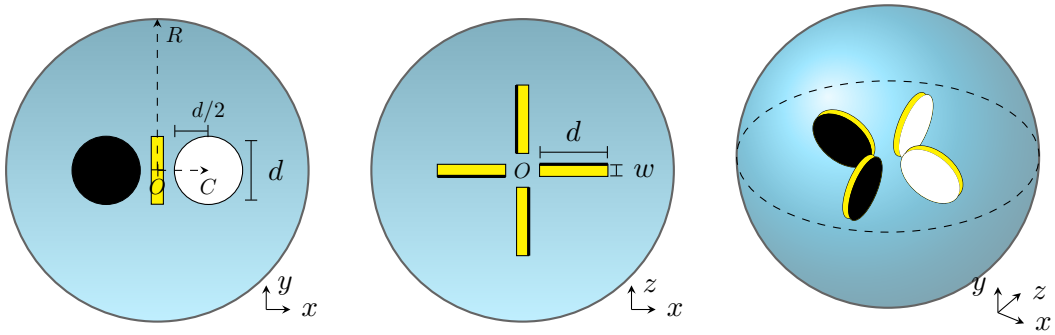
Description of a square-shaped radiometer. Figure II.4 illustrates the geometry of the computational domain of the square-shaped radiometer from various angles: sub-figure II.4a shows the orthographic view viewed from the -z axis, sub-figure II.4b shows the orthographic view viewed from the +y axis and finally sub-figure II.4c provides a perspective view of the computational domain. The glass vessel radius $\overrightarrow{OR}$ is 2.5 units. The centre (0,0,0) of the radiometer is denoted by O. Each vane has a length l of 1 unit and a depth of d of 1 unit and a width w of 0.01 unit. Further, the centre $\overrightarrow{OC}$ of each vane is located at the same distance from the origin. For example, the centre of the vane oriented along the positive x-axis is $(x = 0.25 + l/2,\ y = 0,\ z = 0)$.



(a) Orthographic view : $-z$ axis (b) Orthographic view : $+y$ axis (c) Perspective view

Figure II.4.: Computational domain of a square-shaped radiometer.

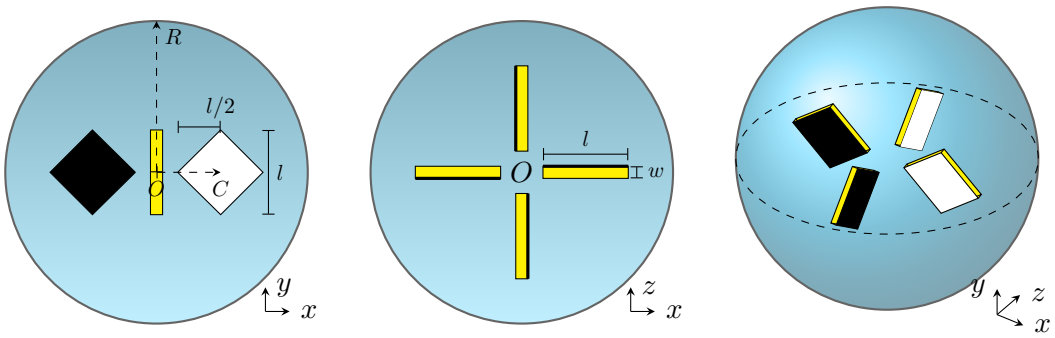
Description of a circle-shaped radiometer. Figure II.5 displays the geometry of the circle-shaped radiometer from various angles: sub-figure II.5a shows the orthographic view viewed from the -z axis, sub-figure II.5b shows the orthographic view viewed from the +y axis and finally sub-figure II.5c provides a perspective view of the computational domain. The glass vessel radius $\overrightarrow{OR}$ is 2.5 units. The centre (0,0,0) of the radiometer is denoted by O. Each vane has a diameter d of $2\sqrt{1/\pi}$ unit. As described previously, the centre $\overrightarrow{OC}$ of each vane is located at the same distance from the origin. For example, the centre of the vane oriented along the positive x-axis is $(x = 0.25 + d/2,\ y = 0,\ z = 0)$.



(a) Orthographic view : $-z$ axis (b) Orthographic view : $+y$ axis (c) Perspective view

Figure II.5.: Computational domain of a circle-shaped radiometer.

Description of a diamond-shaped radiometer. Figure II.6 illustrates the geometry of the computational domain of the square-shaped radiometer from various angles: sub-figure II.6a shows the orthographic view viewed from the -z axis, sub-figure II.6b shows the orthographic view viewed from the +y axis and finally sub-figure II.6c provides a perspective view of the computational domain. The glass vessel radius $\overrightarrow{OR}$ is 2.5 units. The centre (0,0,0) of the radiometer is denoted by O. Each diamond vane has a length l of $\sqrt{2}$ unit and a width w of 0.01 unit. The centre of vane oriented along the $+x$-axis is $x = 0.25 + l/2$, y and $z = 0$.



(a) Orthographic view : $-z$ axis (b) Orthographic view : $+y$ axis (c) Perspective view

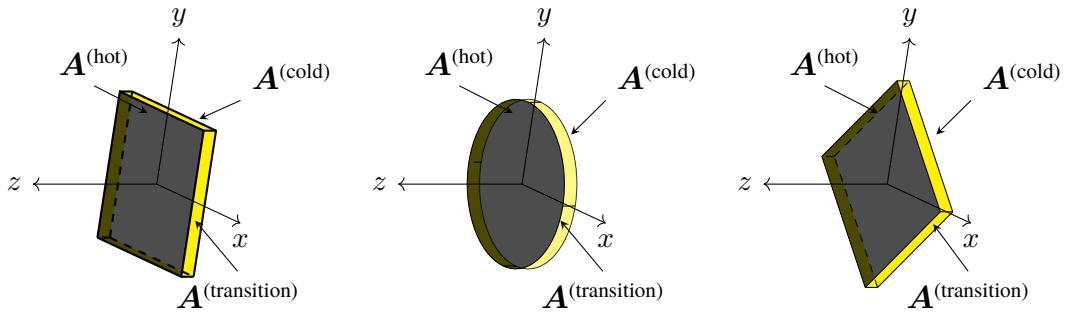
Figure II.6.: Computational domain of a diamond-shaped radiometer.

Calculation of radiometric force. The dimensionless force $\boldsymbol{F}$ experienced by a surface $\boldsymbol{A}$ due to radiometric phenomena is given by

$$\boldsymbol{F} = \int_{\Gamma} (\boldsymbol{\sigma} + p\boldsymbol{I}) \cdot \boldsymbol{n} \, \mathrm{d}\boldsymbol{A}, \tag{II.42}$$

where $\boldsymbol{\sigma}$ is the stress tensor, p is the pressure, $\boldsymbol{I}$ is the identity tensor and $\boldsymbol{n}$ is the vector normal to the surface $\boldsymbol{A}$. Spelling out the components involved in Equation II.42 yields

$$\boldsymbol{F} = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} \int_{\Gamma} \left((\sigma_{xx} + p)n_x + \sigma_{xy} n_y + \sigma_{xz} n_z \right) \, \mathrm{d}\boldsymbol{A} \\ \int_{\Gamma} \left(\sigma_{yx} n_x + (\sigma_{yy} + p)n_y + \sigma_{yz} n_z \right) \, \mathrm{d}\boldsymbol{A} \\ \int_{\Gamma} \left(\sigma_{zx} n_x + \sigma_{zy} n_y + (\sigma_{zz} + p)n_z \right) \, \mathrm{d}\boldsymbol{A} \end{bmatrix}. \tag{II.43}$$



(a) Perspective view of a square-shaped vane (b) Perspective view of a circle-shaped vane (c) Perspective view of a diamond-shaped vane

Figure II.7.: Illustration of grouping of a vane's surfaces: hot surface (indicated by transparant black colour), transition surface (indicated by yellow colour) and cold surface (indicated by white colour but not directly visible in the diagram).

Calculation of net radiometric force. Following the Figure II.42, the net radiometric force $\boldsymbol{F}^{(\mathrm{net})}$ acting on a single vane

$$\boldsymbol{F}^{(\mathrm{net})} = \boldsymbol{F}^{(\mathrm{hot})} + \boldsymbol{F}^{(\mathrm{cold})} + \boldsymbol{F}^{(\mathrm{transition})} \tag{II.44}$$

is calculated by considering the summation of radiometric forces acting on the hot, cold and transition surfaces, denoted respectively as $A^{(\mathrm{hot})}$, $A^{(\mathrm{hot})}$, $A^{(\mathrm{hot})}$ in Figure II.7.

Simulation parameters. The dimensionless temperature on the cold surface and the glass wall is $T^{(\text{cold})}$=1 and on the hot surface is $T^{(\text{hot})}$ = 1.0307167. In units with dimensions, if we assume that the reference temperature of the radiometer is $T^{(\text{ref})} = 323K$, the above temperature setup would translate approximately to $T^{(\text{cold})} = 323K$, and $T^{(\text{hot})} = 333K$. The temperature of the transition surfaces of the vanes which are oriented parallel to the x-axis is governed by the following linear gradient relation

$$T^{(\text{transition})} = T^{(\text{cold})} \pm \frac{\Delta T}{w}\left(z \pm \frac{w}{2}\right), \tag{II.45}$$

where $\pm$ is decided with respect to $+x$ or $-x$ axis, and the temperature of the transition surfaces of the vanes which are oriented parallel to the z-axis is governed by the following linear gradient relation

$$T^{(\text{transition})} = T^{(\text{cold})} \mp \frac{\Delta T}{w}\left(x \mp \frac{w}{2}\right), \tag{II.46}$$

where $\mp$ is decided with respect to $+z$ or $-z$ axis. The Knudsen numbers are chosen from 0.01 till 5.12, effectively doubling at each step. For the fixed temperature gradient, simulations are performed for each Knudsen number and the respective net radiometric forces are calculated.

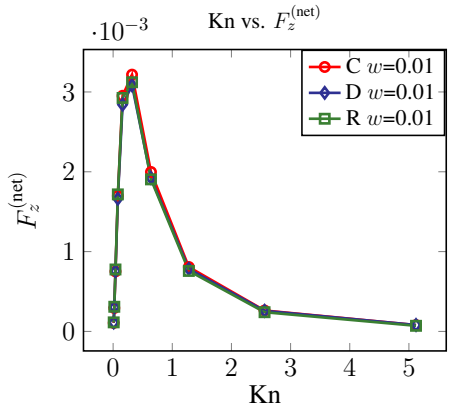


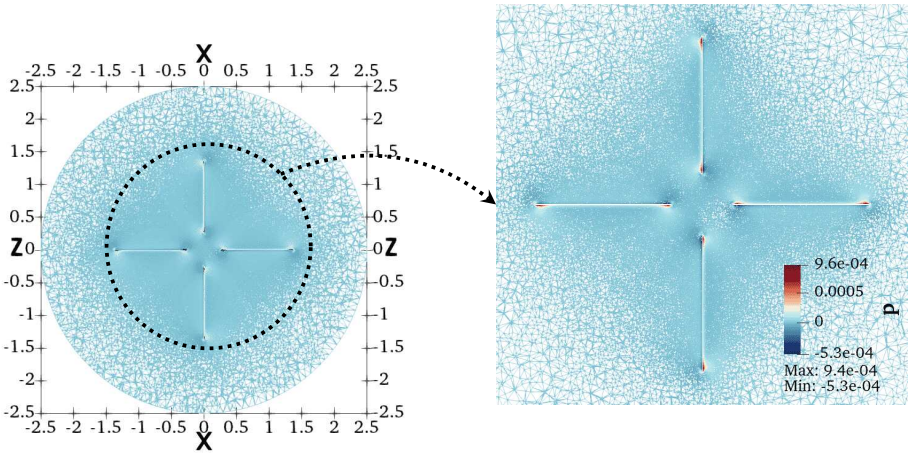
Figure II.8.: Effect of gas pressure ($p \propto \text{Kn}^{-1}$) on the net radiometric force on a single vane

Discussion of the results. Figure II.8 illustrates the relationship between the Knudsen number and the net radiometric force acting on a single vane in a four-vane configuration. The actual data used to produce the plots can be found in the Appendix. Results are shown for three different vane geometries: circle, diamond, and rectangle, denoted C, D, and R respectively. It is observed that regardless of the vane geometry, the radiometric force exhibits similar behaviour with respect to gas pressure.

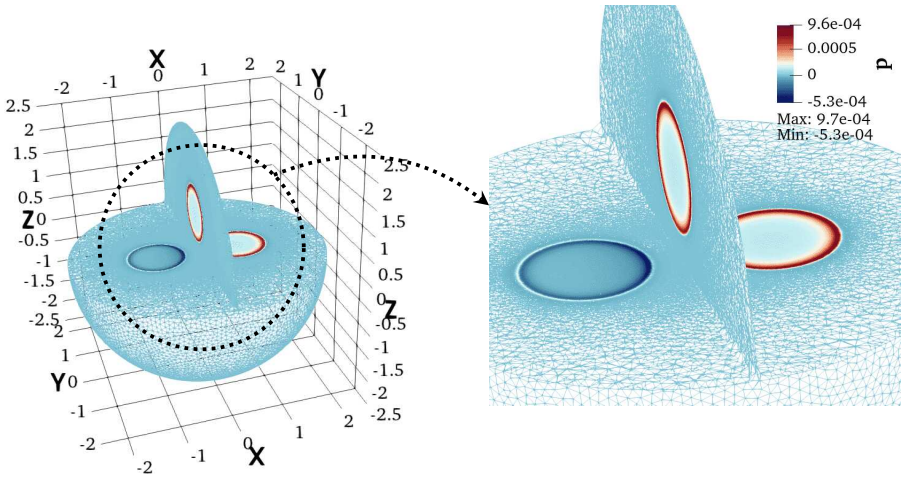
Detailed normal pressure distributions for each vane configuration are provided in FiguresII.9, II.10, and II.11 at a Knudsen number of 0.01. We see that for each vane configuration, at low Knudsen number (high pressure), the normal pressure is concentrated along the edges of the vanes. Furthermore, we observe the comparable order of magnitude of normal pressure acting on each side of a vane. In the following section, tangential stress distribution is illustrated and

discussed in detail. This is due to thermal transpiration, postulated by Reynolds. This was also predicted by Einstein. However, these explanations hold only for low Knudsen numbers.

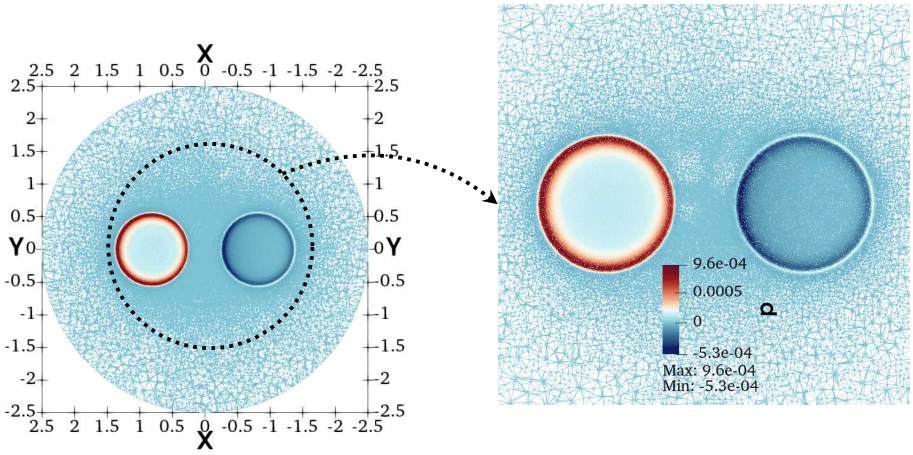
From Figure II.8, we note that at low pressure (high Knudsen number) and at high pressure (low Knudsen number), the radiometric force is very small. This can be explained as follows: at low pressure (high Kn), the collision rate is too low, resulting in poor momentum transfer of gas molecules; at high pressure (low Kn), the collision rate is too high, which hinders the transport of gas momentum. However, there exists an optimal pressure range where the net radiometric force is maximized. This range (Kn = 0.16 to Kn = 0.32 as seen in the Figure II.8) provides a good balance between particle collision rates and the movement of gas molecules. Similar study was conducted for a 2D case [20, 22]. Our findings are consistent with the above studies' findings qualitatively.



(a) Left side: Wireframe view of the x-z plane at $y = 0$, viewed from the y-axis. Right side: Zoomed-in view of the x-z plane; this is the top view.

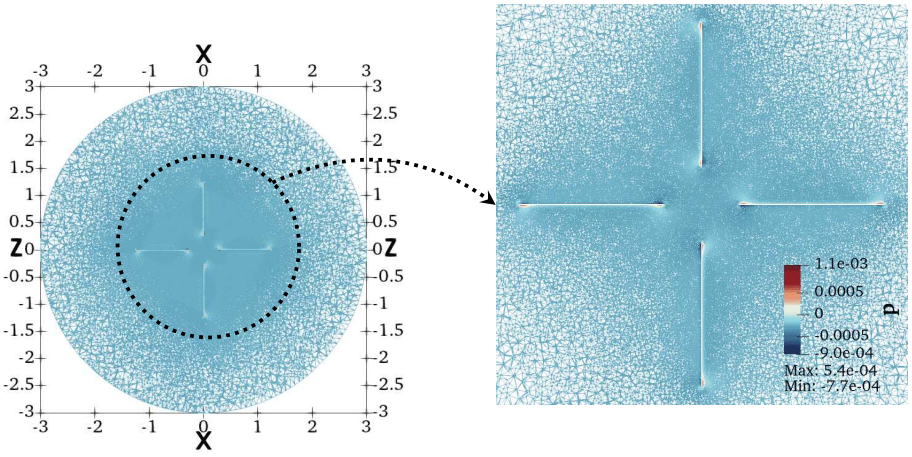


(b) Left side: Perspective view of the wireframe of the x-y-z space. Right side: Zoomed-in view of the vanes in the 3D space; this is the perspective view.

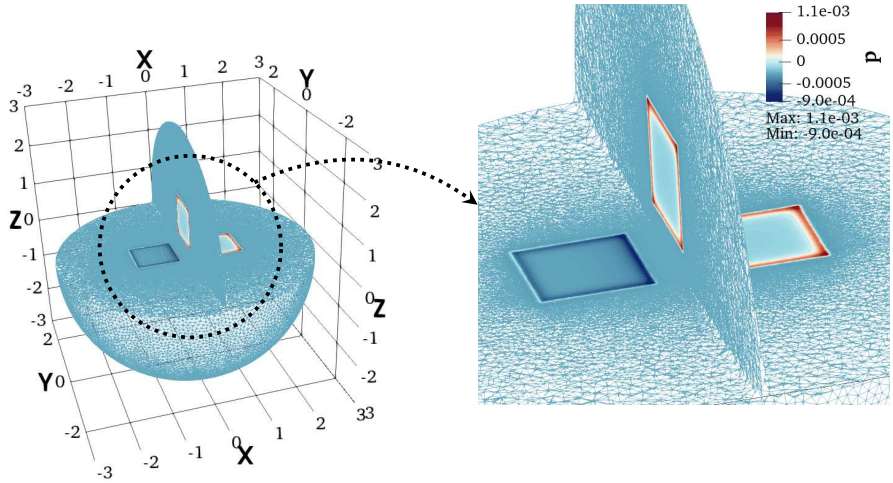


(c) Left side: Wireframe view of the x-y plane at $y = 0$, viewed from the z-axis. Right side: Zoomed-in view of the x-y plane; this is the front view.

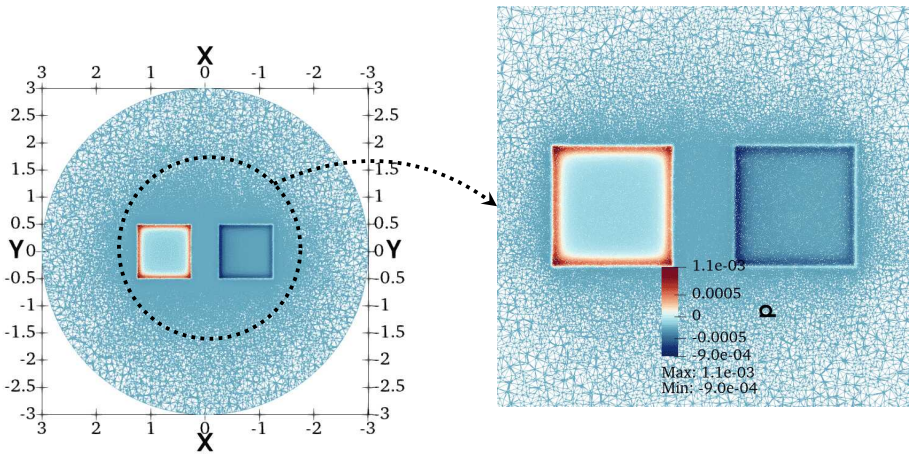
Figure II.9.: Distribution of normal pressure on circle-shaped vanes at a low Knudsen number (Kn = 0.01).



(a) Left side: Wireframe view of the x-z plane at $y = 0$, viewed from the y-axis. Right side: Zoomed-in view of the x-z plane; this is the top view.

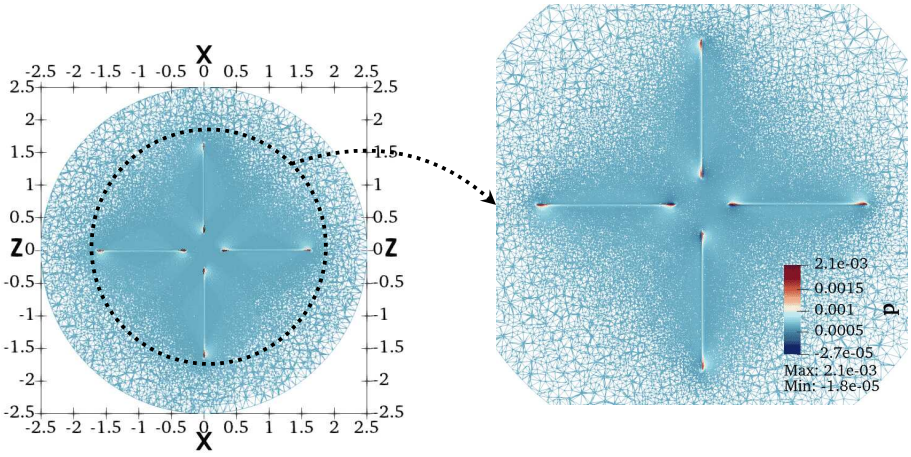


(b) Left side: Perspective view of the wireframe of the x-y-z space. Right side: Zoomed-in view of the vanes in the 3D space; this is the perspective view.

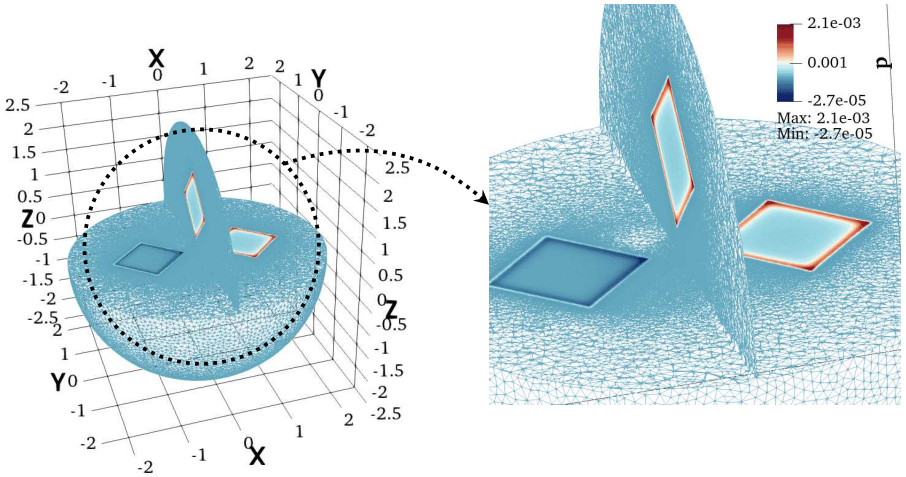


(c) Left side: Wireframe view of the x-y plane at $y = 0$, viewed from the z-axis. Right side: Zoomed-in view of the x-y plane; this is the front view.

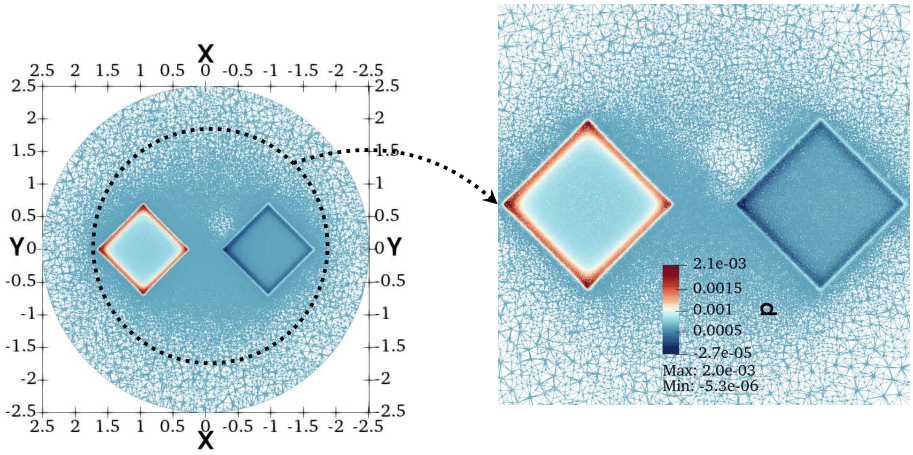
Figure II.11.: Distribution of normal pressure on rectangle-shaped vanes at a low Knudsen number (Kn = 0.01).



(a) Left side: Wireframe view of the x-z plane at $y = 0$, viewed from the y-axis. Right side: Zoomed-in view of the x-z plane; this is the top view.



(b) Left side: Perspective view of the wireframe of the x-y-z space. Right side: Zoomed-in view of the vanes in the 3D space; this is the perspective view.



(c) Left side: Wireframe view of the x-y plane at $y = 0$, viewed from the z-axis. Right side: Zoomed-in view of the x-y plane; this is the front view.

Figure II.10.: Distribution of normal pressure on diamond-shaped vanes at a low Knudsen number (Kn = 0.01).

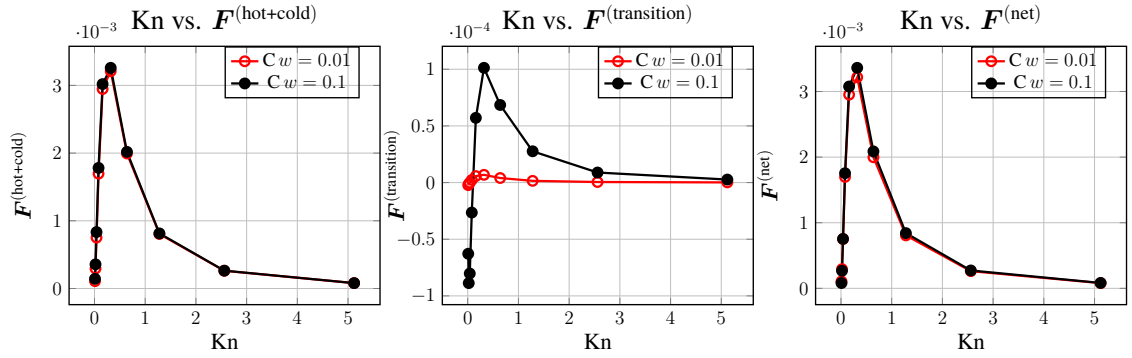
II.5.3. Effect of vane thickness on the radiometric force.

Goal. The goal of this study is to investigate the effect of vane thickness on the radiometric force. Specifically, the focus is on examining the influence of shear forces on the transition surfaces and the normal forces on the hot and cold surfaces. To achieve this, the thickness of the radiometer vanes is varied, and the resulting effects on the radiometric force are compared. The motivation for this study is derived from a similar study conducted on 2D vanes with varied thicknesses and edge angles [23]. The authors have studied the effect of shear stress on the radiometric force.

Simulation parameters. The three radiometer geometries — circle, diamond, and rectangle — from the previous study are considered, with vane thicknesses varied from $w = 0.01$ to $w = 0.1$. All other simulation parameters remain identical to those used in the prior study. The simulations are conducted across a range of Knudsen numbers, from 0.01 to 5.12, doubling at each step.

Discussion of the results. Figure II.12 shows the comparison of the effect of vane thickness on the radiometric force for various vane geometries. The actual data used to produce the plots can be found in the table in the appendix. Each plot shows two curves: one for the vanes with thickness $w = 0.01$ (filled marks) and another for the vanes with thickness $w = 0.1$ (hollow marks).

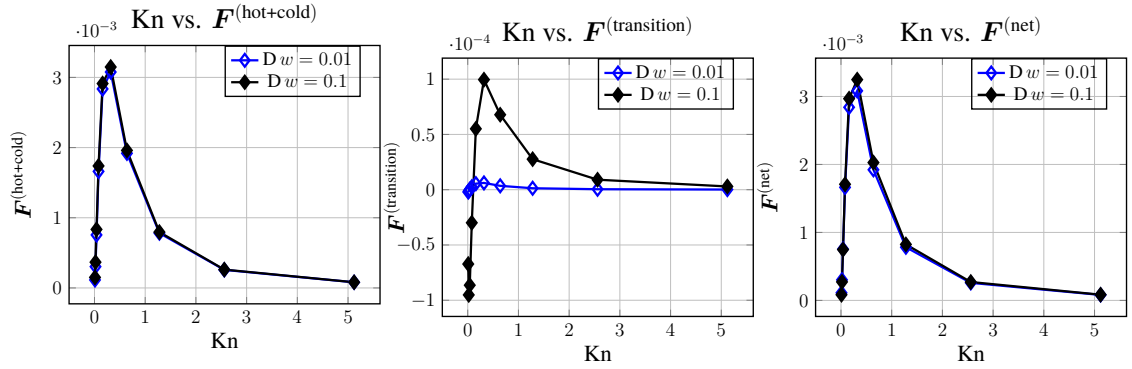
Firstly, we observe that the contribution of forces acting on the cold and hot surfaces of the vane to the net radiometric force is two orders of magnitude larger compared to the force acting on the transition surface. Secondly, the thicker vanes produce an order of magnitude larger force on the transition surface than the thinner vanes at Knudsen numbers where the net force is maximised. Our observations align well with a similar study conducted on 2D radiometer vanes [23].



(a) Comparison of the sum of force on the hot and cold surfaces of the circle vane

(b) Comparison of the force on the transition surface of the circle vane

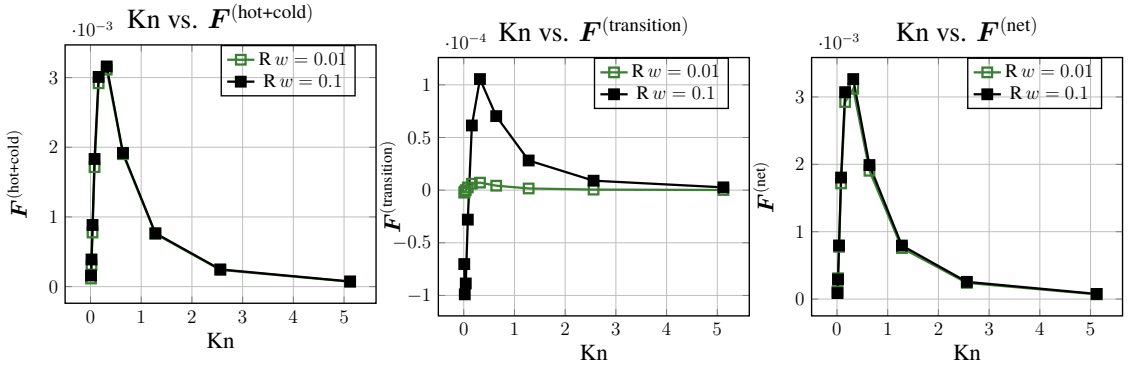
(c) Comparison of the net force on the circle vane



(d) Comparison of the sum of force on the hot and cold surfaces of the diamond vane

(e) Comparison of the force on the transition surface of the diamond vane

(f) Comparison of the net force on the diamond vane



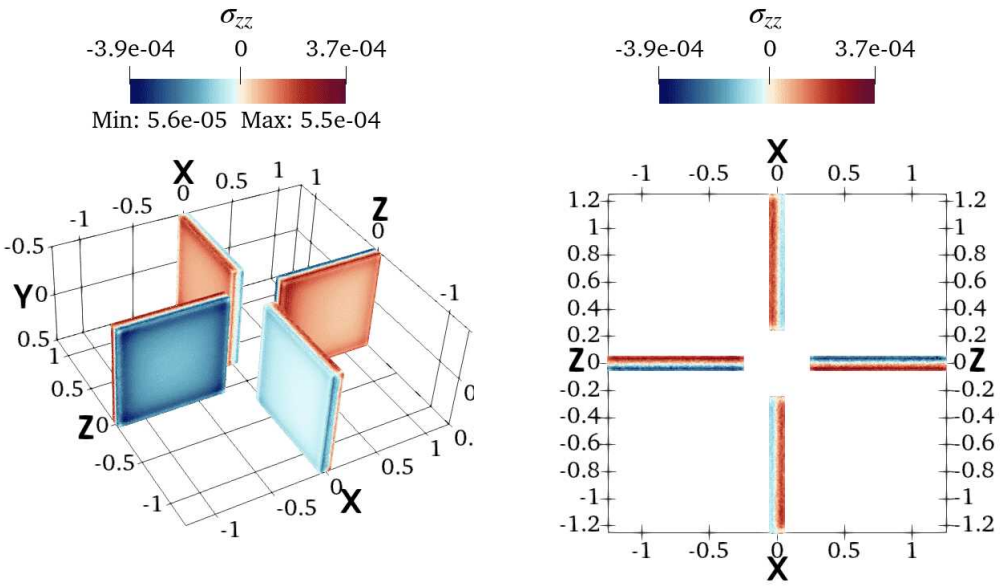
(g) Comparison of the sum of force on the hot and cold surfaces of the rectangle vane

(h) Comparison of the force on the transition surface of the rectangle vane

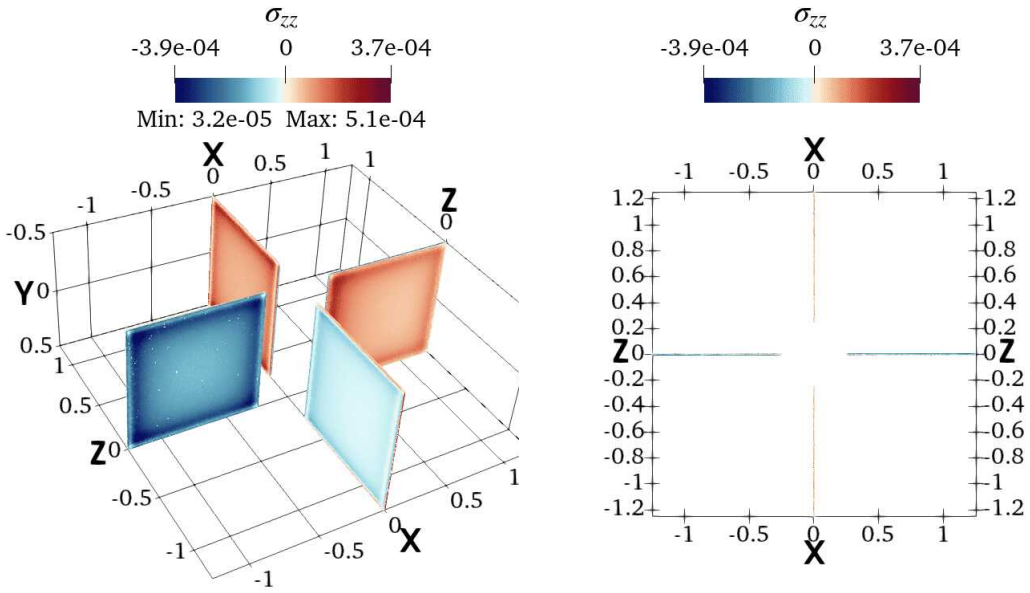
(i) Comparison of the net force on the rectangle vane

Figure II.12.: Effect of vane thickess w on the net radiometric force for various vane geometries.

We note that across all three vane geometries at low Knudsen numbers (high pressure), the thicker vanes ($w = 0.1$, represented by filled plot marks) generate slightly less net radiometric force compared to the thinner vanes ($w = 0.01$, represented by open plot marks). Please refer to the appendix for the exact numerical data. Although the differences in net radiometric forces between varying vane thicknesses are small, they prompt the question of why these slight differences occur. Figure II.13 illustrates the distribution of the tangential stress component σ_{zz} on square-shaped vanes with different thicknesses at a Knudsen number of Kn=0.01.



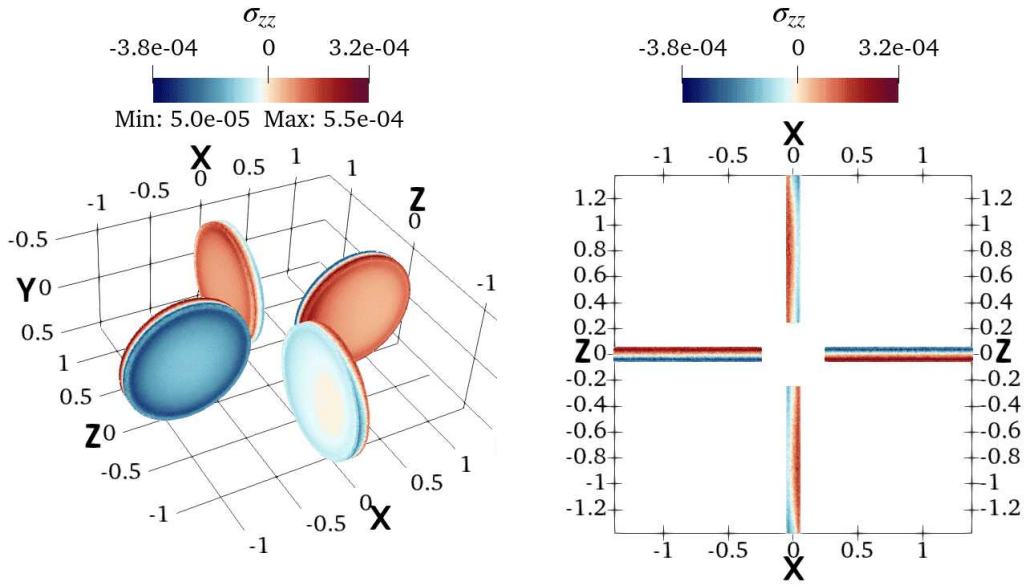
(a) Left side: Perspective view of vanes with $w = 0.1$. Right side: Top view of the x-z plane.



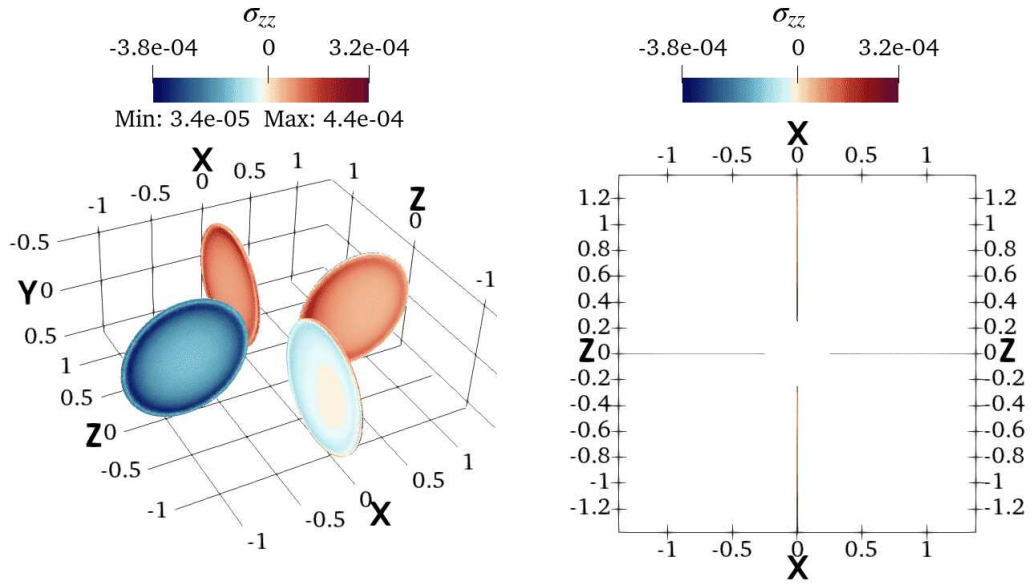
(b) Left side: Perspective view of vanes with $w = 0.01$. Right side: Top view of the x-z plane.

Figure II.13.: Distribution of the stress component σ_{zz} on square-shaped vanes with different thicknesses ($w = 0.01$ and $w = 0.1$) at a low Knudsen number (Kn = 0.01).

Regardless of the vane geometry, at high pressure (low Kn), the shear force opposes the normal radiometric force. This is clearly demonstrated by the data presented in the table in the Appendix A.3. As shown, at low Kn (high pressure), the sign of the shear force is opposite to that of the normal force, leading to a reduction in the net radiometric force.



(a) Left side: Perspective view of vanes with $w = 0.1$. Right side: Top view of the x-z plane.

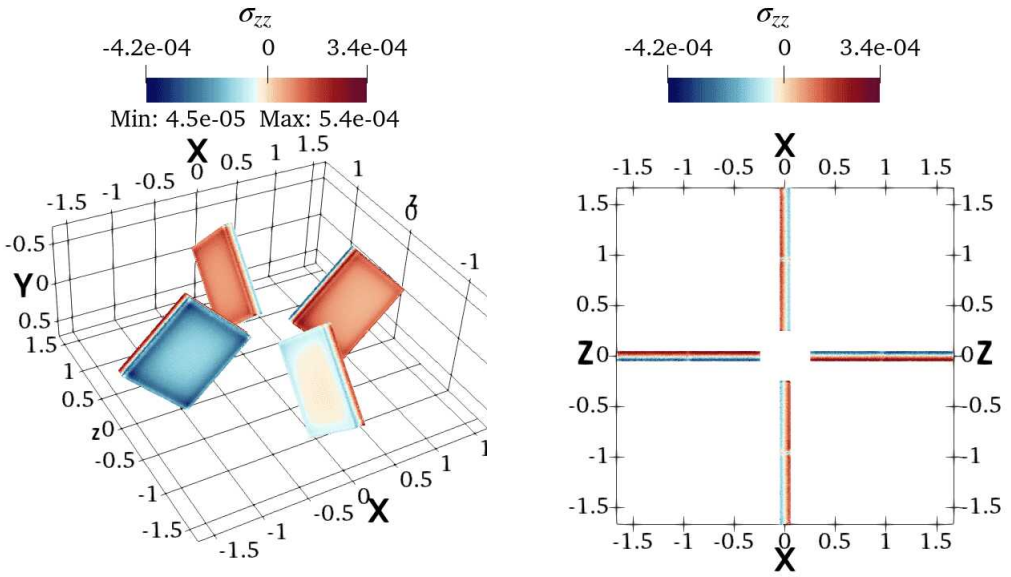


(b) Left side: Perspective view of vanes with $w = 0.01$. Right side: Top view of the x-z plane.

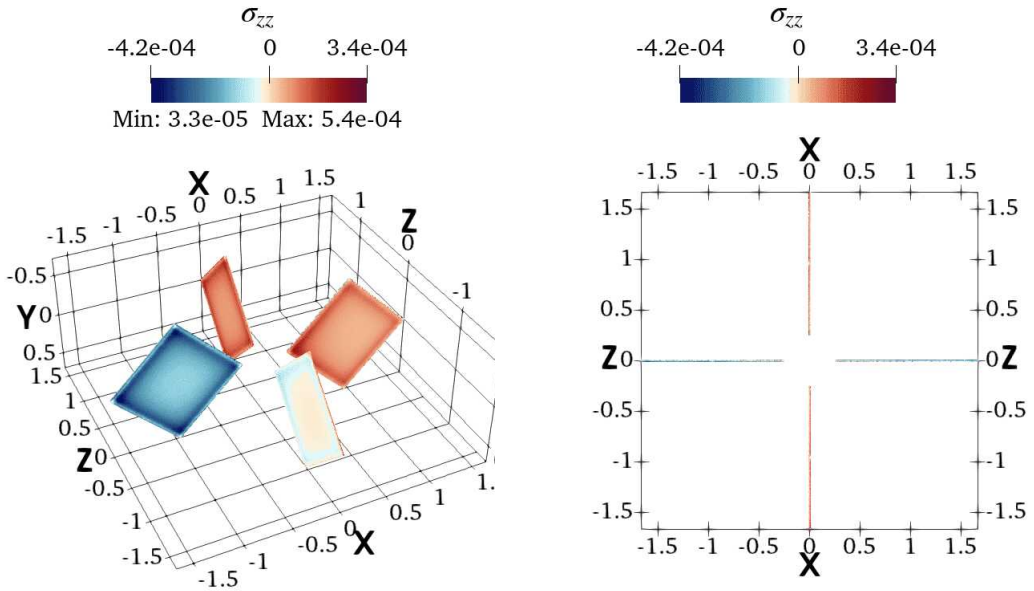
Figure II.14.: Distribution of the stress component σ_{zz} on circle-shaped vanes with different thicknesses ($w = 0.01$ and $w = 0.1$) at a low Knudsen number (Kn = 0.01).

This observation can be explained as follows: at low Kn (high pressure) and moderately low temperatures, the gas density around the vane edge is influenced by the surrounding regions. For a Knudsen number of 0.01, the distribution of the shear stress component σ_{zz} is visualized

in Figures II.14 and II.15. Increasing the vane thickness adversely affects gas recirculation, leading to less efficient thermal transpiration. Therefore, a thicker vane is expected to reduce the total radiometric force in the high-pressure regime. However, as the Knudsen number increases (pressure decreases), the contribution of shear forces becomes negligible, resulting in the net radiometric force remaining unaffected.



(a) Left side: Perspective view of vanes with $w = 0.1$. Right side: Top view of the x-z plane.



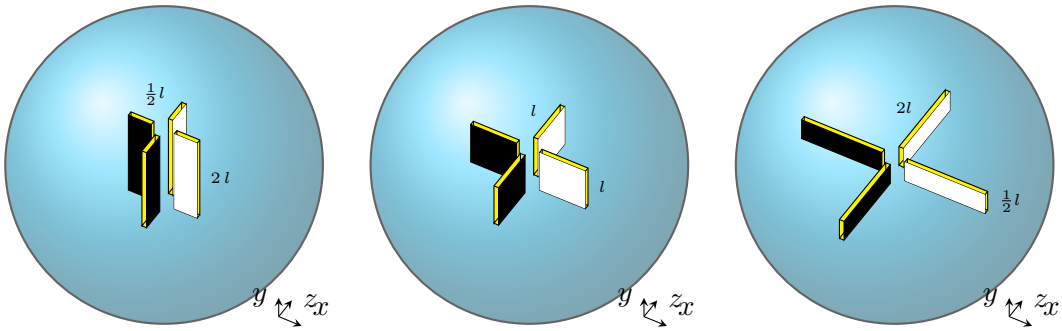
(b) Left side: Perspective view of vanes with $w = 0.01$. Right side: Top view of the x-z plane.

Figure II.15.: Distribution of the stress component σ_{zz} on diamond-shaped vanes with different thicknesses ($w = 0.01$ and $w = 0.1$) at a low Knudsen number (Kn = 0.01).

II.5.4. Effect of vane length on the radiometric force.

Objective. This study aims to investigate the impact of vane length on the radiometric force. We focus on the rectangular vanes discussed in previous sections. Simulations are conducted on vanes of differing lengths across a broad range of Knudsen numbers, and the resulting net radiometric force is compared.

Computational domain. Figure II.16 illustrates three rectangular radiometers, each with a different length-to-depth aspect ratio (l/d). All three geometries have the same width of $w = 0.01$. The simulation parameters are consistent with those used in the previous studies.



(a) Rectangle 1 (R1): vane length l is only half the size of vane depth d.

(b) Rectangle 2 (R2): vane length l is the same size of the vane depth d.

(c) Rectangle 3 (R3): vane length l is double the size of vane depth d.

Figure II.16.: Perspective views of three rectangle-shaped radiometers with different aspect ratios.

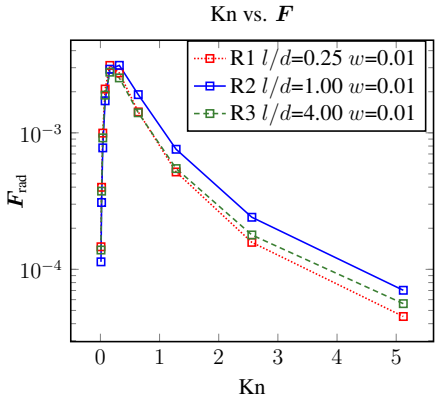
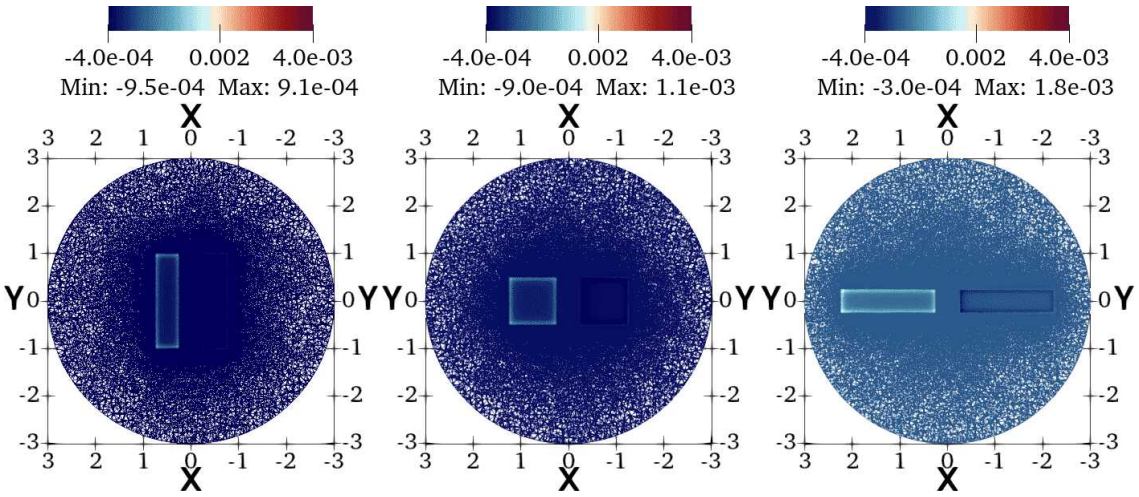
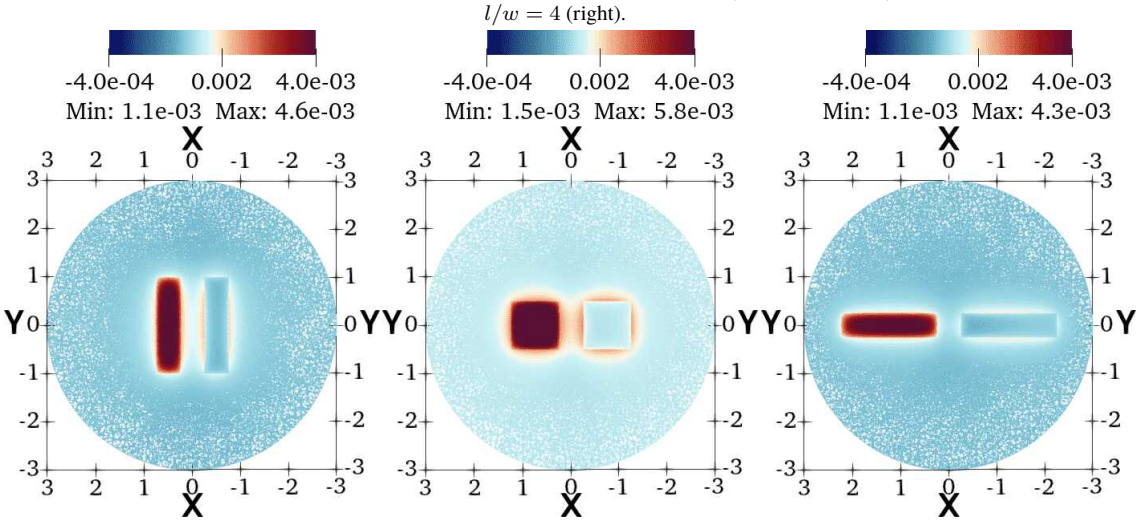


Figure II.17.: Effect of vane aspect ratios (l/d) on the net radiometric force.

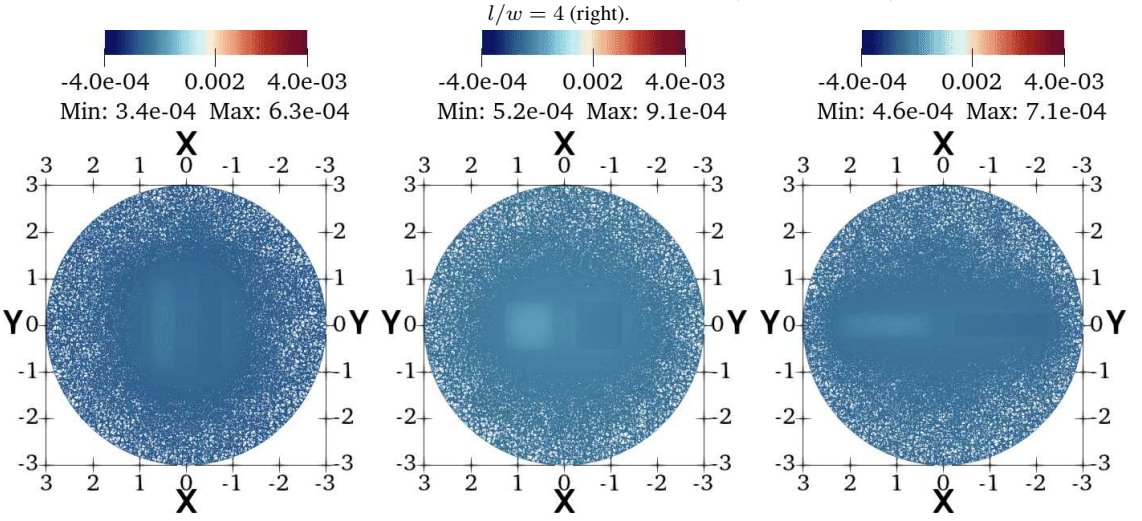
Discussion of the results. Figure II.17 compares the simulation results for three different aspect ratios. We analyse the results based on two criteria: peak radiometric force and range of applicability. Geometry R1, with an aspect ratio of $l/d = 0.25$, performs well at high pressure but shows reduced performance as the pressure decreases compared to the other geometries. Geometry R3, with an aspect ratio of $l/d = 4$, maintains relatively good performance across a wide range of pressures but exhibits a lower peak radiometric force compared to the others. Overall, geometry R2, with an aspect ratio of $l/d = 1$, provides the best performance across all three geometries over a broad range of pressures.



(a) Comparison of pressure p distributions at Kn = 0.01 on vanes with aspect ratios: $l/w = 0.5$ (left), $l/w = 1$ (middle), and $l/w = 4$ (right).



(b) Comparison of pressure p distributions at Kn = 0.32 on vanes with aspect ratios: $l/w = 0.5$ (left), $l/w = 1$ (middle), and $l/w = 4$ (right).



(c) Comparison of pressure p distributions at Kn = 2.56 on vanes with aspect ratios: $l/w = 0.5$ (left), $l/w = 1$ (middle), and $l/w = 4$ (right).

Figure II.18.: View of the x-y plane from the $+z$-axis: Comparison of normal pressure p distributions on vanes with different aspect ratios at three Knudsen numbers: Kn = 0.01, 0.32, and 2.56.

II.6. Summary

In this chapter, we explored the three-dimensional R13 moment system and its corresponding boundary conditions. We also developed the variational formulation of the 3D system step-by-step, highlighting key considerations for implementation in FEniCSR13.

We tested the framework on the flow-over-a-sphere problem and compared the numerical solutions with analytical results, observing convergence across all moment variables. We noted that the convergence rate depends on several factors, including the order of finite elements, stabilization parameters, and mesh quality. Identifying optimal values for these factors could further enhance the convergence rate.

Next, we simulated a three-dimensional Crookes radiometer and conducted three main investigations. First, we studied the effect of pressure on the radiometric force by examining three distinct vane geometries. For each geometry, we ran multiple simulations varying the Knudsen number and calculated the net radiometric force. We observed that the net radiometric force was minimal when the Knudsen number was either too small or too large, with an optimal range (Kn = 0.16 to Kn = 0.32) where the net force was maximized. In this pressure range, momentum transfer by gas molecules is the most efficient.

In our second investigation, we varied the vane thickness and repeated the simulations. We found that increased thickness negatively affected the net radiometric force at lower Knudsen numbers. At high pressures, thicker vanes exhibited inefficient thermal transpiration, leading to a reduced net radiometric force. As the Knudsen number increased, the effect of forces on the vanes' lateral surfaces became negligible and the net radiometric force remained unaffected.

Lastly, we explored the impact of aspect ratio (length/width) of rectangular vanes on the net radiometric force. We conducted simulations for various aspect ratios and compared the resulting net radiometric forces. We found that a rectangle with an aspect ratio of 1 had a wider effective range compared to the other aspect ratios. Additionally, the rectangle with a unit aspect ratio produced the peak net radiometric force around Kn = 0.32.

Chapter III

Generic Moment Approximations to the Boltzmann Equation and their Generic Numerical Solutions

III.1. Introduction

Scope. This chapter is structured into four main sections. The first section derives the first-order formulation of the Poisson system and the R13 moment system, highlighting the advantages of rewriting higher-order PDEs in the first-order form. The second section covers the derivation of generic moment systems and discusses the formulation of entropy-stable boundary conditions. In the third section, we derive a generic variational formulation and provide a concrete implementation of the 13 moment equations. The final section discusses various numerical simulations based on this implementation.

Formulation of first-order PDE system. Mathematical problem can be approached in various ways depending on our convenience. We consider two example problems namely Poisson problem and 2D R13 moment system. The Poisson problem involves a second-order derivative of some quantity. Similarly, two-dimensional R13 moment system will produce terms with second-order derivatives upon closure relation substitution. In the following, we present few ways of formulating and solving these problems. In particular, we emphasise the advantage of using first-order formulation of these problems.

III.1.1. Example: Poisson System

Poisson problem. Consider the following Poisson problem which describes the spatial distribution of the scalar quantity θ:

$$\begin{aligned} -\boldsymbol{\nabla}^2\theta &= f \quad && \text{in } \Omega, \\ \theta &= \theta_D \quad && \text{on } \Gamma_D, \\ \boldsymbol{\nabla}\theta \cdot \boldsymbol{n} = \frac{\partial\theta}{\partial \boldsymbol{n}} &= g \quad && \text{on } \Gamma_N, \end{aligned} \tag{III.1}$$

where θ_D is the given Dirichlet boundary condition at the boundary Γ_D and g is the given Neumann boundary condition at the boundary Γ_N and f is the source term.

In order to obtain the numerical solution of the above scalar PDE using finite element method, it is advantageous to work with a weak formulation of Equation III.1. Let the trial function space $\mathbb{V} \subset \mathbb{H}^1$ and the test function space $\mathbb{V}_0 \subset \mathbb{H}^1$ be defined as follows:

$$\mathbb{V} = \left\{ v \in \mathbb{H}^1 : v|_{\Gamma_D} = \theta_D \right\}, \quad \mathbb{V}_0 = \left\{ v \in \mathbb{H}^1 : v|_{\Gamma_D} = 0 \right\}. \tag{III.2}$$

To derive the weak formulation, multiply the Equation (III.1) with appropriate test function $\kappa \in \mathbb{V}_0$ and integrate it over the domain

$$-\int_\Omega \kappa \boldsymbol{\nabla}^2 \theta \, \mathrm{d}\boldsymbol{x} = \int_\Omega \kappa f \, \mathrm{d}\boldsymbol{x}. \tag{III.3}$$

And apply partial integration to the left hand side of the equation

$$+\int_\Omega \nabla\kappa \cdot \nabla\theta \, \mathrm{d}\boldsymbol{x} - \int_\Omega \nabla \cdot (\kappa \nabla\theta) \, \mathrm{d}\boldsymbol{x} = \int_\Omega \kappa f \, \mathrm{d}\boldsymbol{x}. \tag{III.4}$$

To embed the boundary condition in the weak formulation, use Gauss theorem to convert the volume integral to the surface integral

$$+\int_\Omega \nabla\kappa \cdot \nabla\theta \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma_N} \kappa(\nabla\theta \cdot \boldsymbol{n}) \, \mathrm{d}\boldsymbol{l} = \int_\Omega \kappa f \, \mathrm{d}\boldsymbol{x}. \tag{III.5}$$

Finally in the case of Dirichlet boundary condition the boundary integral vanish because we require $\kappa = 0$. Therefore, the weak formulation reads

$$+\int_\Omega \nabla\kappa \cdot \nabla\theta \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma_N} \kappa g \, \mathrm{d}\boldsymbol{l} = \int_\Omega \kappa f \, \mathrm{d}\boldsymbol{x}. \tag{III.6}$$

The final abstract weak formulation reads: find $\theta \in \mathbb{V}$ such that $\mathcal{A}(\theta, \kappa) = \mathcal{L}(\kappa)$ for all $\kappa \in \mathbb{V}_0$, where

$$\begin{aligned} \mathcal{A}(\theta, \kappa) &:= \int_\Omega \boldsymbol{\nabla}\kappa \cdot \boldsymbol{\nabla}\theta \, \mathrm{d}\boldsymbol{x}, \\ \mathcal{L}(\kappa) &:= \int_\Omega \kappa f \, \mathrm{d}\boldsymbol{x} + \int_{\Gamma_N} \kappa g \, \mathrm{d}\boldsymbol{x}. \end{aligned} \tag{III.7}$$

Mixed formulation. Let us reformulate the Poisson problem in (III.1) as a mixed problem by introducing an auxiliary variable. Recalling $-\boldsymbol{\nabla}^2\theta = \boldsymbol{\nabla} \cdot (-\boldsymbol{\nabla}\theta)$ and substituting $\boldsymbol{s} := -\boldsymbol{\nabla}\theta$ yields the divergence form of the canonical Poisson problem (III.1)

$$\begin{aligned} \boldsymbol{\nabla} \cdot \boldsymbol{s} &= f && \text{in } \Omega, \\ \boldsymbol{\nabla}\theta &= -\boldsymbol{s} && \text{in } \Omega, \\ \theta &= \theta_D && \text{on } \Gamma_D, \\ \boldsymbol{s} \cdot \boldsymbol{n} = \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{n}} &= -g && \text{on } \Gamma_N. \end{aligned} \tag{III.8}$$

Equations (III.1) and (III.8) are equivalent. Note that the original boundary condition in Equation (III.1) and the reformulated boundary condition in Equation (III.8) mean the same.

$$-g = \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{n}} = \boldsymbol{s} \cdot \boldsymbol{n} = (-\boldsymbol{\nabla}\theta) \cdot \boldsymbol{n} = -\frac{\partial \theta}{\partial \boldsymbol{n}} = -g. \tag{III.9}$$

The difference lies in how the boundary conditions are incorporated into the weak formulation. In the mixed formulation, the Neumann-type boundary condition is an essential boundary condition and therefore should be enforced on the function space, while the Dirichlet-type boundary condition is a natural boundary condition and therefore should be applied directly to the weak formulation.

To obtain the weak formulation of the Equation (III.8), we treat each equation separately. Consider the first equation. Let the test and trial function spaces be the same and is defined as follows

$$\mathbb{V} := \left\{ v \in \mathbb{H}^1 \right\} \tag{III.10}$$

We multiply the first equation by the test function $\kappa \in \mathbb{V}$

$$\int_\Omega \kappa(\boldsymbol{\nabla} \cdot \boldsymbol{s}) \, \mathrm{d}\boldsymbol{x} = \int_\Omega \kappa \, f \, \mathrm{d}\boldsymbol{x}. \tag{III.11}$$

Consider the second equation. Let the trial function space $\mathbb{S}_g \subset \mathbb{H}(\mathrm{div})$ and the test function space $\mathbb{S}_0 \subset \mathbb{H}(\mathrm{div})$ be defined as follows:

$$\mathbb{S}_g := \{ \boldsymbol{v} \in \mathbb{H}(\mathrm{div}) : \boldsymbol{v} \cdot \boldsymbol{n} = -g \}, \quad \mathbb{S}_0 := \{ \boldsymbol{v} \in \mathbb{H}(\mathrm{div}) : \boldsymbol{v} \cdot \boldsymbol{n} = 0 \}. \tag{III.12}$$

We multiply the second equation by the test function $\boldsymbol{r} \in \mathbb{S}_0$ and we get

$$\int_\Omega \boldsymbol{r} \cdot \boldsymbol{\nabla}\theta \, \mathrm{d}\boldsymbol{x} = -\int_\Omega \boldsymbol{r} \cdot \boldsymbol{s} \, \mathrm{d}\boldsymbol{x}. \tag{III.13}$$

and apply integration-by-parts technique followed by Gauss divergence theorem

$$\int_\Omega \theta(\boldsymbol{\nabla} \cdot \boldsymbol{r}) \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma_D} \theta_D(\boldsymbol{r} \cdot \boldsymbol{n}) \, \mathrm{d}\boldsymbol{l} = -\int_\Omega \boldsymbol{r} \cdot \boldsymbol{s} \, \mathrm{d}\boldsymbol{x}. \tag{III.14}$$

The final weak formulation reads: find $(\theta, \boldsymbol{s}) \in \mathbb{V} \times \mathbb{S}_g$ such that $\mathcal{A}((\theta, \boldsymbol{s}), (\kappa, \boldsymbol{r})) = \mathcal{L}((\kappa, \boldsymbol{r}))$ for all $(\kappa, \boldsymbol{r}) \in \mathbb{V} \times \mathbb{S}_0$, where

$$\begin{aligned} \mathcal{A}((\theta, \boldsymbol{s}), (\kappa, \boldsymbol{r})) &:= \int_\Omega \kappa(\boldsymbol{\nabla} \cdot \boldsymbol{s}) \, \mathrm{d}\boldsymbol{x} + \int_\Omega \theta(\boldsymbol{\nabla} \cdot \boldsymbol{r}) \, \mathrm{d}\boldsymbol{x} + \int_\Omega \boldsymbol{r} \cdot \boldsymbol{s} \, \mathrm{d}\boldsymbol{x}, \\ \mathcal{L}((\kappa, \boldsymbol{r})) &:= \int_\Omega \kappa f \, \mathrm{d}\boldsymbol{x} + \int_{\Gamma_D} \theta_D(\boldsymbol{r} \cdot \boldsymbol{n}) \, \mathrm{d}\boldsymbol{x}. \end{aligned} \tag{III.15}$$

First-order system formulation. Yet another way to formulate the mixed formulation in Equation (III.8) is to rewrite it as a matrix-vector form. To do so, we consider the mixed Poisson system in two dimensions. Writing out the involved components explicitly yields

$$\begin{aligned} \frac{\partial s_x}{\partial x} + \frac{\partial s_y}{\partial y} &= f \quad \text{in } \Omega, \\ \frac{\partial \theta}{\partial x} &= -s_x \quad \text{in } \Omega, \\ \frac{\partial \theta}{\partial y} &= -s_y \quad \text{in } \Omega, \\ \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{n}} &= -g \quad \text{on } \Gamma_N. \end{aligned} \tag{III.16}$$

By introducing a new variable $\boldsymbol{W} := [\theta, s_x, s_y]$, the Equation (III.16) can be written in matrix-vector form.

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{:=\boldsymbol{A}^{(x)}} \underbrace{\begin{bmatrix} \frac{\partial \theta}{dx} \\ \frac{\partial s_x}{dx} \\ \frac{\partial s_y}{dx} \end{bmatrix}}_{:=\frac{\partial \boldsymbol{W}}{dx}} + \underbrace{\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}}_{:=\boldsymbol{A}^{(y)}} \underbrace{\begin{bmatrix} \frac{\partial \theta}{dy} \\ \frac{\partial s_x}{dy} \\ \frac{\partial s_y}{dy} \end{bmatrix}}_{:=\frac{\partial \boldsymbol{W}}{dy}} + \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{:=P} \underbrace{\begin{bmatrix} \theta \\ s_x \\ s_y \end{bmatrix}}_{:=\boldsymbol{W}} = \begin{bmatrix} f \\ 0 \\ 0 \end{bmatrix} \quad \text{in } \Omega, \tag{III.17}$$

$$\frac{\partial \boldsymbol{s}}{\partial \boldsymbol{n}} = -g \quad \text{on } \Gamma_N$$

The generic weak formulation of the form in (III.17) is discussed in the later section.

III.1.2. Example: R13 Moment System

Mixed formulation. 3D R13 moment system is introduced and its mixed weak formulation is derived in detail in Chapter II. In this section let us present the 2D R13 moment equations

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = \dot{m}, \tag{III.18}$$

$$\boldsymbol{\nabla} p + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{b}, \tag{III.19}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} + \boldsymbol{\nabla} \cdot \boldsymbol{s} = r, \tag{III.20}$$

$$\frac{4}{5}(\boldsymbol{\nabla} \boldsymbol{s})_{\text{stf}} + 2(\boldsymbol{\nabla} \boldsymbol{u})_{\text{stf}} + \boldsymbol{\nabla} \cdot \boldsymbol{m} = -\frac{1}{\text{Kn}} \boldsymbol{\sigma}, \tag{III.21}$$

$$\frac{5}{2} \boldsymbol{\nabla} \theta + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} + \frac{1}{2} \boldsymbol{\nabla} \cdot \boldsymbol{R} + \frac{1}{6} \boldsymbol{\nabla} \Delta = -\frac{1}{\text{Kn}} \frac{2}{3} \boldsymbol{s}, \tag{III.22}$$

$$\boldsymbol{m} = -2\text{Kn} (\boldsymbol{\nabla} \boldsymbol{\sigma})_{\text{stf}}, \tag{III.23}$$

$$\boldsymbol{R} = -\frac{24}{5} \text{Kn} (\boldsymbol{\nabla} \boldsymbol{s})_{\text{stf}}, \tag{III.24}$$

$$\Delta = -12\text{Kn} (\boldsymbol{\nabla} \cdot \boldsymbol{s}), \tag{III.25}$$

and its associated 2D boundary conditions for the sake of completeness,

$$(u_n - u_n^w) = \epsilon \tilde{\chi} \left((p - p^w) + \sigma_{nn} \right), \tag{III.26}$$

$$\sigma_{nt} = \tilde{\chi} \left((u_t - u_t^w) + \frac{1}{5} s_t + m_{nnt} \right), \tag{III.27}$$

$$R_{nt} = \tilde{\chi} \left(-(u_t - u_t^w) + \frac{11}{5} s_t - m_{nnt} \right), \tag{III.28}$$

$$s_n = \tilde{\chi} \left(2(\theta - \theta^w) + \frac{1}{2} \sigma_{nn} + \frac{2}{5} R_{nn} + \frac{2}{15} \Delta \right), \tag{III.29}$$

$$m_{nnn} = \tilde{\chi} \left(-\frac{2}{5} (\theta - \theta^w) + \frac{7}{5} \sigma_{nn} - \frac{2}{25} R_{nn} - \frac{2}{75} \Delta \right), \tag{III.30}$$

$$\left(m_{ntt} + \frac{1}{2} m_{nnn} \right) = \tilde{\chi} \left(\frac{1}{2} \sigma_{nn} + \sigma_{tt} \right). \tag{III.31}$$

The mixed formulation of the Equations (III.18)-(III.25) is similar to the 3D case discussed in Section II.2 in Chapter II and is derived in detail in [9].

First-order formulation. Similar to the Poisson system discussed in the previous section, it is possible to rewrite the Equations (III.18) - (III.25) as a first-order system of PDEs. Let us define
$\boldsymbol{W} := \{p,\ u_x,\ u_y,\ \theta,\ \sigma_{xx},\ \sigma_{xy},\ \sigma_{yy},\ s_x,\ s_y,\ m_{xxx},\ m_{xxy},\ m_{xyy},\ m_{yyy},\ \Delta,\ R_{xx},\ R_{xy},\ R_{yy}\}$
such that

$$\boldsymbol{A}^{(x)}\frac{\partial \boldsymbol{W}}{\partial x} + \boldsymbol{A}^{(y)}\frac{\partial \boldsymbol{W}}{\partial y} + \boldsymbol{P}\boldsymbol{W} = \boldsymbol{F}, \tag{III.32}$$

where

$$\boldsymbol{A}^{(x)}\frac{\partial \boldsymbol{W}}{\partial x} = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{4}{3} & 0 & 0 & 0 & 0 & 0 & \frac{8}{15} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{2}{5} & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 & -\frac{4}{15} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{5}{2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{6} & \frac{1}{2} & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\
0 & 0 & 0 & 0 & \frac{9}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{8}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{2}{5} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -\frac{6}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{56}{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{14}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{28}{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \frac{\partial}{\partial x} \begin{bmatrix} p \\ u_x \\ u_y \\ \theta \\ \sigma_{xx} \\ \sigma_{xy} \\ \sigma_{yy} \\ s_x \\ s_y \\ m_{xxx} \\ m_{xxy} \\ m_{xyy} \\ m_{yyy} \\ \Delta \\ R_{xx} \\ R_{xy} \\ R_{yy} \end{bmatrix} \tag{III.33}$$

$$\boldsymbol{A}^{(y)}\frac{\partial \boldsymbol{W}}{\partial y} = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 & -\frac{4}{15} & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{2}{5} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{4}{3} & 0 & 0 & 0 & 0 & 0 & \frac{8}{15} & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\
0 & 0 & 0 & \frac{5}{2} & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{6} & 0 & \frac{1}{2} \\
0 & 0 & 0 & 0 & 0 & -\frac{6}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & -\frac{2}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{8}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{9}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{28}{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{14}{5} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{56}{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \frac{\partial}{\partial y} \begin{bmatrix} p \\ u_x \\ u_y \\ \theta \\ \sigma_{xx} \\ \sigma_{xy} \\ \sigma_{yy} \\ s_x \\ s_y \\ m_{xxx} \\ m_{xxy} \\ m_{xyy} \\ m_{yyy} \\ \Delta \\ R_{xx} \\ R_{xy} \\ R_{yy} \end{bmatrix} \tag{III.34}$$

$$\boldsymbol{P}\,\boldsymbol{W} = \frac{1}{\mathrm{Kn}}\begin{bmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&\frac{2}{3}&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&\frac{2}{3}&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&\frac{3}{2}&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&\frac{3}{2}&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&\frac{3}{2}&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&\frac{3}{2}&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&\frac{2}{3}&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&\frac{7}{6}&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\frac{7}{6}&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&\frac{7}{6}
\end{bmatrix}\begin{bmatrix}
p\\u_x\\u_y\\\theta\\\sigma_{xx}\\\sigma_{xy}\\\sigma_{yy}\\s_x\\s_y\\m_{xxx}\\m_{xxy}\\m_{xyy}\\m_{yyy}\\\Delta\\R_{xx}\\R_{xy}\\R_{yy}
\end{bmatrix} \tag{III.35}$$

Similarly, the boundary equations in (III.26) - (III.31) can also be written as matrix-vector form $\boldsymbol{W}^{(\boldsymbol{n},\boldsymbol{o})} = \boldsymbol{L}\mathbb{A}\boldsymbol{W}^{(\boldsymbol{n},\boldsymbol{e})} + \boldsymbol{g}$,

$$\begin{bmatrix}u_n\\\sigma_{nt}\\s_n\\m_{nnn}\\m_{ntt}\\R_{nt}\end{bmatrix} = \tilde{\chi}\begin{bmatrix}
\epsilon&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&\frac{1}{5}\\
0&0&2&-\frac{2}{5}&\frac{1}{5}&0&0&0&0&0&0\\
0&0&-\frac{2}{5}&\frac{52}{25}&-\frac{26}{25}&0&0&0&0&0&0\\
0&0&-\frac{1}{5}&-\frac{26}{25}&\frac{38}{25}&0&0&0&0&0&0\\
0&-1&0&0&0&0&0&0&0&0&\frac{13}{5}
\end{bmatrix}\begin{bmatrix}
1&0&1&1&0&0&0&0&0&0&0\\
0&1&0&0&0&\frac{2}{5}&0&0&0&0&0\\
0&0&1&\frac{2}{5}&0&0&0&0&\frac{1}{15}&\frac{1}{5}&0\\
0&0&0&1&\frac{1}{2}&0&0&0&0&0&0\\
0&0&0&\frac{1}{2}&1&0&0&0&0&0&0\\
0&0&0&0&0&\frac{1}{5}&0&0&0&0&0
\end{bmatrix}\begin{bmatrix}
p\\u_t\\\theta\\\sigma_{nn}\\\sigma_{tt}\\s_t\\m_{nnt}\\m_{ttt}\\\Delta\\R_{nn}\\R_{tt}
\end{bmatrix} + \tilde{\chi}\begin{bmatrix}
-\epsilon p^w\\-u_t^w\\-2\theta^w\\+\frac{2}{5}\theta^w\\-\frac{1}{5}\theta^w\\+u_t^w
\end{bmatrix}. \tag{III.36}$$

III.2. Derivation of Generic Moment Systems

In this section, we discuss the derivation of generic moment systems and boundary conditions. The following subsections III.2.1 and III.2.2 mainly follow [24]. However, the mathematical notations have been adapted to fit our framework.

III.2.1. Abstract generic Moment Systems

STF moment tensors. Following the tradition of rational extended thermodynamics [25] we require the set of moment variables $\{u_\alpha\}_{\alpha\in I_M}$ be invariant tensors of increasing degrees, like

$$\{1, u_{i_1}, u_{i_1 i_2}, u_{i_1 i_2 i_3}, \cdots, u_{i_1 i_2 \cdots i_n}\}. \tag{III.37}$$

As the moment order increases, we will have to use tensors of higher degree, typically more than the second degree. Therefore, it is advantageous to decompose the moment tensors into invariant, irreducible tensorial parts to structure the set of variables. A detailed introduction into tensor decomposition using indices is given in [24]. As a consequence, in what follows we assume the set of moments variables to be symmetric, trace-free tensors.

Polynomial basis functions. We will consider a scaled and shifted velocity variable $\boldsymbol{\xi} := \left(\boldsymbol{c} - \boldsymbol{v}^{(\mathrm{ref})}\right)/\sqrt{\theta^{(\mathrm{ref})}}$ with reference velocity $\boldsymbol{v}^{(\mathrm{ref})}$ and reference temperature $\theta^{(\mathrm{ref})}$ which are assumed to be constant in space and in time. In moment approximation for kinetic equations the set of variables in (III.37) are implied by projections with polynomial test functions generating the space

$$\text{span } \{1, \xi_{i_1}, \xi_{i_1}\xi_{i_2}, \xi_{i_1}\xi_{i_2}\xi_{i_3}, \ \cdots \ , \ \xi_{i_1}, \xi_{i_2}, \ldots, \xi_{i_M}\} \tag{III.38}$$

This polynomial space in (III.38) is isomorphic to tensor variables space in (III.37) and may be decomposed analogously into rotationally invariant, irreducible subspaces.

STF representation of polynomial basis functions. We consider the basis polynomial given by symmetric, trace-free tensors

$$\text{span } \left\{\xi_{\langle i_1}\xi_{i_2}\cdots\xi_{i_M\rangle}\boldsymbol{\xi}^{2a}\right\}_{n=0,\cdots,M;\, a=0,\cdots,M_n} \tag{III.39}$$

where each STF polynomial of degree n is augmented by an isotropic monomial degree $2a$ with $a = 0, \cdots, M_n$. We denote the squared norm of the vector ξ_{i_1} by $\boldsymbol{\xi}^2$. The angular brackets indicate the symmetric, trace-free part of a tensor and for the polynomial it can be computed by

$$\xi_{\langle i_1}\xi_{i_2}\cdots\xi_{i_M\rangle} = \sum_{r=0}^{\lfloor \frac{n}{2} \rfloor} \frac{(-1)^n n!}{2^r r!(n-2r)!} \frac{\Gamma\left(n-r+\frac{1}{2}\right)}{\Gamma\left(n+\frac{1}{2}\right)} \|\boldsymbol{\xi}\|^{2r} \delta_{(i_1 i_2}\cdots\delta_{i_{2r-1}i_{2r}}\xi_{i_{2r+1}}\cdots\xi_{i_{n-2r})} \tag{III.40}$$

or other formulas. Thus, choosing M and M_n for $n = 0, \cdots, M$ appropriately facilitate construction of a polynomial space that is equivalent to (III.38). We note that different degrees of STF polynomials are orthogonal by construction while the monomials are not. The basis polynomials

$$\psi^{(a)}_{i_1 i_2 \cdots i_n}(\boldsymbol{\xi}) = \xi_{\langle i_1}\xi_{i_2}\cdots\xi_{i_n\rangle} p_a^{(n)}\left(\frac{\boldsymbol{\xi}^2}{2}\right) \tag{III.41}$$

replace the monomials by the orthogonal polynomial

$$p_a^{(n)}(x) = a! L_a^{\left(n+\frac{1}{2}\right)}(x), \tag{III.42}$$

where $L_a^{(\nu)}$ is the generalised Laguerre polynomial of degree a. The basis functions $\boldsymbol{\psi}$ are normalised such that the coefficient of the highest monomial is unity. In this way, the projections of the distribution f by $\psi^{(a)}_{i_1 i_2 \cdots i_n}$ resemble classical monomial moments in the leading order.

As a consequence, the expansion coefficients will require scaling factors. The expansion of the distribution is now given by

$$f^{(\text{normalised})}(\boldsymbol{\xi}) = \sum_{n=0}^{M} \sum_{a=0}^{M_n} \frac{1}{\mathcal{N}^{(n,a)}} w^{(a)}_{i_1 i_2 \cdots i_n} \psi^{(a)}_{i_1 i_2 \cdots i_n}(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}) \tag{III.43}$$

with symmetric trace-free tensorial coefficients $w^{(a)}_{i_1 i_2 \cdots i_n}$ for $n = 0, \cdots, M$ and $a = 0, \cdots, M_n$, where the choice of M and M_n for $n = 0, \cdots, M$ defines the moment theory. The scaling factor is

$$\mathcal{N}^{(a,n)} = n! a! \frac{\Gamma\left(n + a + \frac{3}{2}\right)}{\Gamma\left(n + \frac{3}{2}\right)} \tag{III.44}$$

and corresponds to the squared weighted norm of $\psi^{(a)}_{i_1 i_2 \cdots i_n}$. Following the classical approach of Grad [26] the reference distribution is a Maxwellian

$$f^{(\text{ref})}(\boldsymbol{\xi}) = \frac{\rho^{(\text{ref})}/m}{\sqrt{2\pi\theta^{(\text{ref})}}^3} e^{-\frac{\boldsymbol{\xi}^2}{2}}, \tag{III.45}$$

however, based on the constant reference density $\rho^{(\text{ref})}$, the reference temperature $\theta^{(\text{ref})}$, and mean velocity $v^{(\text{ref})}$.

Definition of expansion coefficients. The expansion coefficients $w^{(a)}_{i_1 i_2 \cdots i_n}$ in (III.43) can be calculated as

$$\rho^{(\text{ref})} w^{(a)}_{i_1 i_2 \cdots i_n} = m \int_{\mathbb{R}^3} \psi^{(a)}_{i_1 i_2 \cdots i_n} \left(\frac{\boldsymbol{c} - \boldsymbol{v}^{(\text{ref})}}{\sqrt{\theta^{(\text{ref})}}} \right) f(\boldsymbol{c}) \, \mathrm{d}\boldsymbol{c} \tag{III.46}$$

.

Projections using precise definitions. Replacing $\boldsymbol{c} := \boldsymbol{v}^{(\text{ref})} + \sqrt{\theta^{(\text{ref})}}\boldsymbol{\xi}$ in (I.11) yields

$$\frac{\partial f}{\partial t} + \sum_{i=1}^{3} v_i^{(\text{ref})} \frac{\partial f}{\partial x_i} + \sqrt{\theta^{(\text{ref})}} \sum_{i=1}^{3} \xi_i \frac{\partial f}{\partial x_i} = S(f). \tag{III.47}$$

Projecting the above equation leads to the evolution equations for the expansion coefficients

$$\begin{aligned} &\rho^{(\text{ref})} \frac{\partial w^{(a)}_{i_1 i_2 \cdots i_n}}{\partial t} + \rho^{(\text{ref})} \sum_{l=1}^{3} v_l^{(\text{ref})} \frac{\partial w^{(a)}_{i_1 i_2 \cdots i_n}}{\partial x_l} \\ &\qquad + \rho^{(\text{ref})} \sqrt{\theta^{(\text{ref})}} \sum_{l=1}^{3} \sum_{m=0}^{N} \sum_{b=0}^{M_n} \boldsymbol{A}^{(l,a,b)}_{i_1 i_2 \cdots i_n j_1 j_2 \cdots j_n} \frac{\partial w^{(b)}_{j_1 j_2 \cdots j_n}}{\partial x_l} = P^{(a)}_{i_1 i_2 \cdots i_n}, \end{aligned} \tag{III.48}$$

where the transportation matrix in the l-th direction is defined as

$$\rho^{(\text{ref})} \boldsymbol{A}^{(l,a,b)}_{i_1 i_2 \cdots i_n j_1 j_2 \cdots j_n} = m \int_{\mathbb{R}^3} \xi_l \psi^{(a)}_{i_1 i_2 \cdots i_n}(\boldsymbol{\xi}) \psi^{(b)}_{j_1 j_2 \cdots j_n}(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{c} \tag{III.49}$$

which are constant.

Essential components of a tensor. In moment theory we select only a set of independent components of a symmetric trace-free coefficients $w^{(a)}_{i_1 i_2 \cdots i_n}$ from a set of essential multi-indices. The selection is made such that we can construct the complete symmetric trace-free tensor using the set of independent components. Following [27], we use only the symmetric part and drop all other entries. This leads to multi-indices

$$\alpha \in I_M := \{(n, a, i_1 i_2 \cdots i_n)\}_{a \leq M_n, n \leq M}\,, \tag{III.50}$$

where M is the level of the moment theory and the values M_n for $n = 0, \cdots, M$ give further specifications of the choice of variables. As an example, consider the coefficient $w_{i_1 i_2}$ with tensor rank $n = 2$ generated by the polynomial $\psi_{i_1 i_2}$ where we write

$$w_{i_1 i_2} := \begin{bmatrix} w_{xx} & w_{xy} & w_{xz} \\ w_{xy} & w_{yy} & w_{yz} \\ w_{xz} & w_{yz} & -w_{xx} - w_{yy} \end{bmatrix}, \quad \psi_{i_1 i_2} := \begin{bmatrix} \psi_{xx} & \psi_{xy} & \psi_{xz} \\ \psi_{xy} & \psi_{yy} & \psi_{yz} \\ \psi_{xz} & \psi_{yz} & -\psi_{xx} - \psi_{yy} \end{bmatrix}, \tag{III.51}$$

and hence the essential indices are the $(xx, xy, xz, yy, yz)-$entries, because the tensor is symmetric and trace-free. A STF-tensor $w_{i_1 i_2 i_3}$ with rank $n = 3$ is described by the

$$(xxx, xxy, xxz, xyy, xyz, yyy, yyz)$$

entries. Thus, for a fixed (n, a) we have a subset of the essential indices $\beta \in I_{n,a}$ for the variable $w^{(a)}_{i_1 \cdots i_n}$.

STF-Tensor contractions. For practical implementations, we cannot replace the tensorial contraction between the coefficients $\boldsymbol{w}_\alpha$ and the polynomials $\boldsymbol{\psi}_\alpha$ in (III.43) by a scalar product. That is, for fixed $n > 1$ and any a,

$$w^{(a)}_{i_1 \cdots i_n} \psi^{(a)}_{i_1 \cdots i_n} \neq \sum_{\beta \in I_{n,a}} w^{(a)}_\beta \psi^{(a)}_\beta, \tag{III.52}$$

if the components of $w^{(a)}_\beta$, $\psi^{(a)}_\beta$ with $\beta \in I_{n,a}$ are the corresponding essential components. For $n = 2$ in (III.51), we obtain

$$\begin{aligned} w_{i_1 i_2} \psi_{i_1 i_2} &= w_{xx}(2\psi_{xx} + \psi_{yy}) + w_{xy}(2\psi_{xy}) + w_{xz}(2\psi_{xz}) + w_{yy}(2\psi_{yy} + \psi_{xx}) + w_{yz}(2\psi_{yz}), \\ \sum_{\beta \in I_{n,a}} w_\beta \psi_\beta &= w_{xx}\psi_{xx} + w_{xy}\psi_{xy} + w_{xz}\psi_{xz} + w_{yy}\psi_{yy} + w_{yz}\psi_{yz}, \\ \therefore w_{i_1 i_2} \psi_{i_1 i_2} &\neq \sum_{\beta \in I_{n,a}} w_\beta \psi_\beta. \end{aligned} \tag{III.53}$$

In summary, the tensorial contraction of trace-free tensors is not completely diagonalised, at least not for our choices of essential components. It is possible to redefine the coefficients and polynomials such that the scalar product becomes diagonal which in fact leads to spherical harmonic basis functions. However, if we want to keep the entries of the coefficients as variables and their defining polynomials in (III.46), an alternative is to find dual polynomials φ_α, such that (III.53) holds with φ_α instead of ψ_α.

Basis reformulation. We require for each fixed n and a

$$\frac{1}{\mathcal{N}^{(n,a)}} w^{(a)}_{i_1\cdots i_n} \psi^{(a)}_{i_1\cdots i_n} = \sum_{\beta\in I_{n,a}} w^{(n,a)}_{\beta} \psi^{(n,a)}_{\beta}, \tag{III.54}$$

where we included the normalisation factor from (III.44). This yields the definition of a linear combination

$$\varphi^{(n,a)}_{\beta} = \frac{1}{\mathcal{N}^{\beta}} \sum_{\gamma\in I_{n,a}} S^{(n)}_{\beta\gamma} \psi^{(n,a)}_{\gamma}, \tag{III.55}$$

where the constant symmetrisation matrix $S^{(n)}_{\beta\gamma}$ depends only on the tensor rank n and is symmetric positive definite, because it was derived from a scalar product. From the above example with $n = 2$ and $\mathcal{N}^{(2,0)} = 2$, the dual polynomials can be found by

$$\begin{aligned}
\varphi_{xx} &= \psi_{xx} + \frac{1}{2}\psi_{yy},\\
\varphi_{xy} &= \psi_{xy},\\
\varphi_{xz} &= \psi_{xz},\\
\varphi_{yy} &= \psi_{yy} + \frac{1}{2}\psi_{xx},\\
\varphi_{yz} &= \psi_{yz}.
\end{aligned} \tag{III.56}$$

The corresponding symmetrisation matrix reads

$$S^{(2)}_{\alpha\beta} = \begin{bmatrix} 1 & 0 & 0 & \frac{1}{2} & 0\\ 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0\\ \frac{1}{2} & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad \alpha,\beta \in \{xx, xy, xz, yy, yz\} \tag{III.57}$$

Generic symmetric moment system. The expansion (III.43) becomes diagonal for the essential indices I_M and reads

$$f(\boldsymbol{\xi}) = \sum_{\beta\in I_M} w_\beta \varphi_\beta(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}), \tag{III.58}$$

such that

$$\int_{\mathbb{R}^3} \psi_\alpha(\boldsymbol{\xi})\varphi_\beta(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{c} = \delta_{\alpha\beta} \qquad \alpha,\beta \in I_M. \tag{III.59}$$

The orthogonal projection of the Boltzmann equation yields (I.15), but the system matrix in the i-th direction,

$$\hat{\boldsymbol{A}}^{(i)}_{\alpha\beta} = \frac{m}{\rho^{(\text{ref})}} \int_{\mathbb{R}^3} \xi_i \psi_\alpha(\boldsymbol{\xi})\varphi_\beta(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{c}, \tag{III.60}$$

is now not symmetric. However, it is possible to symmetrise it by the multiplication with the block matrix $S_{\alpha\beta}$ built from all individual symmetrisation matrices

$$S_{\alpha\beta} = \text{diag}\left\{ S^{(n)}_{\alpha\beta} \right\}_{a\le M_n, n\le M}, \tag{III.61}$$

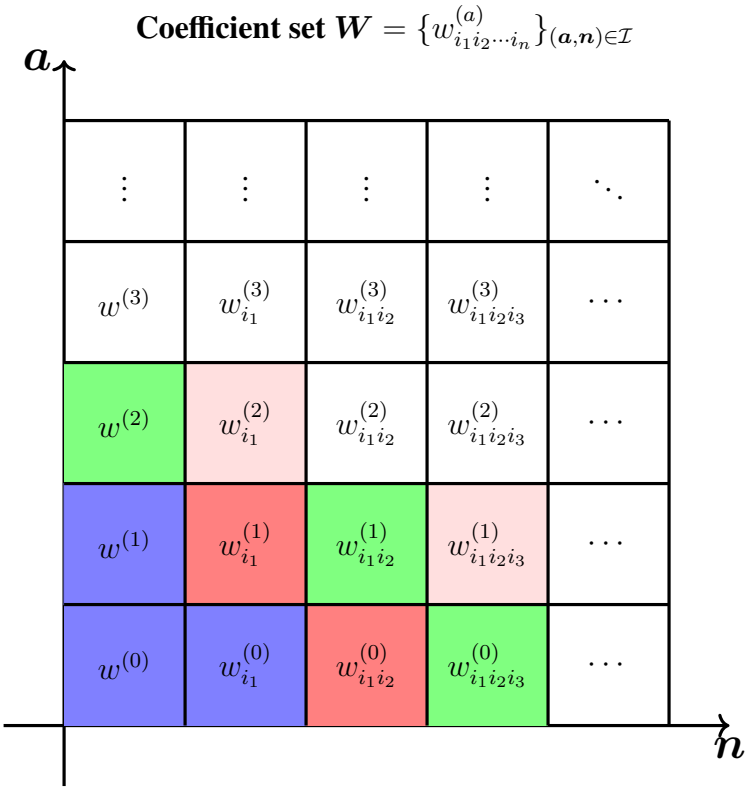
such that

$$\boldsymbol{\mathcal{A}}^{(i)}_{\alpha\beta} := \sum_{\gamma\in I_M} S_{\alpha\beta}\hat{\boldsymbol{A}}^{(i)}_{\gamma\beta} = \frac{m}{\rho^{(\mathrm{ref})}}\sum_{\gamma\in I_M}\int_{\mathbb{R}^3}\xi_i S_{\alpha\gamma}\psi_\gamma\varphi_\beta f^{(\mathrm{ref})}\,\mathrm{d}\boldsymbol{c} = \frac{m}{\rho^{(\mathrm{ref})}}\int_{\mathbb{R}^3}\xi_i\varphi_\alpha\varphi_\beta f^{(\mathrm{ref})}\,\mathrm{d}\boldsymbol{c} \tag{III.62}$$

and we have the symmetrised system

$$\rho^{(\mathrm{ref})}S_{\alpha\beta}\frac{\partial w_\beta}{\partial t} + \rho^{(\mathrm{ref})}\sum_{l=1}^{3} v_l^{(\mathrm{ref})}S_{\alpha\beta}\frac{\partial w_\beta}{\partial x_l} + \rho^{(\mathrm{ref})}\sqrt{\theta^{(\mathrm{ref})}}\sum_{i=1}^{3}\sum_{\beta\in I_M}\boldsymbol{\mathcal{A}}^{(i)}_{\alpha\beta}\frac{\partial w_\beta}{\partial x_i} = \sum_{\beta\in I_M} S_{\alpha\beta}P_\beta, \tag{III.63}$$

which remains linear on the left hand side.



Euler $:= \left\{w^{(0)}, w^{(0)}_{i_1}, w^{(1)}\right\}$

NSF $:= \left\{w^{(0)}, w^{(0)}_{i_1}, w^{(1)}, w^{(0)}_{i_1 i_2}, w^{(1)}_{i_1}\right\}$

G13 $:= \left\{w^{(0)}, w^{(0)}_{i_1}, w^{(1)}, w^{(0)}_{i_1 i_2}, w^{(1)}_{i_1}\right\}$

G26 $:= \left\{w^{(0)}, w^{(0)}_{i_1}, w^{(1)}, w^{(0)}_{i_1 i_2}, w^{(1)}_{i_1}, w^{(2)}, w^{(1)}_{i_1 i_2}, w^{(0)}_{i_1 i_2 i_3}\right\}$

G45 $:= \left\{w^{(0)}, w^{(0)}_{i_1}, w^{(1)}, w^{(0)}_{i_1 i_2}, w^{(1)}_{i_1}, w^{(2)}, w^{(1)}_{i_1 i_2}, w^{(0)}_{i_1 i_2 i_3}, w^{(2)}_{i_1}, w^{(1)}_{i_1 i_2 i_3}\right\}$

$\vdots$

Figure III.1.: General representation of moment coefficients

III.2.2. Abstract boundary conditions

Rotational invariance. The moment system we derived is rotationally invariant [24]. The coordinate transformation $\widetilde{\xi}_i = T_{ij}(\boldsymbol{n})\xi_j$ transforms the velocity variable from the Cartesian coordinate system $i = (x, y, z)$ to the normal-tangential coordinate system $i = (n, t_1, t_2)$. Correspondingly, we rewrite the indices set from I_M to $I_M^{(\boldsymbol{n})}$. This gives rise to the definition of transformation matrix $\widetilde{T}_{\alpha\beta}$ using which we transform the basis function $\psi_\alpha(\widetilde{\boldsymbol{\xi}}) = \sum_{\beta\in I_M} \widetilde{T}_{\alpha\beta}(\boldsymbol{n})\psi_\beta(\boldsymbol{\xi})$, $\alpha \in I_M^{(\boldsymbol{n})}$. For the dual basis we have $\varphi_\alpha(\widetilde{\boldsymbol{\xi}}) = \sum_{\beta\in I_M} \widehat{T}_{\alpha\beta}(\boldsymbol{n})\varphi_\beta(\boldsymbol{\xi})$, $\alpha \in I_M^{(\boldsymbol{n})}$. Note that $\widetilde{T}_{\alpha\beta}$ is not orthogonal, but we have $\widehat{T}_{\alpha\beta}(\boldsymbol{n})^{-1} = \widetilde{T}_{\alpha\beta}(\boldsymbol{n})^T$. The distribution function is required to be invariant under the coordinate rotation in the sense that the coefficients must be transformed according to

$$f(\boldsymbol{c}) \stackrel{!}{=} \sum_{\beta\in I_M^{(\boldsymbol{n})}} w_\beta^{(\boldsymbol{n})}\varphi_\beta(T\boldsymbol{\xi})f^{(\text{ref})}(\boldsymbol{\xi}) = \sum_{\gamma\in I_M}\sum_{\beta\in I_M^{(\boldsymbol{n})}} w_\beta^{(\boldsymbol{n})}\widehat{T}_{\beta\gamma}(\boldsymbol{n})\varphi_\gamma(\boldsymbol{\xi})f^{(\text{ref})}(\boldsymbol{\xi}) \tag{III.64}$$

which gives

$$w_\alpha^{(\boldsymbol{n})} = \sum_{\beta\in I_M} \widetilde{T}_{\alpha\beta}(\boldsymbol{n})w_\beta, \quad \alpha \in I_M^{(\boldsymbol{n})} \tag{III.65}$$

as a requirement for the coefficients transformed into the normal direction. As a result, the equation for the coefficients are rotationally invariant meaning that the projected matrix

$$\mathcal{A}_{\alpha\beta}^{(\boldsymbol{n})} := n_i\mathcal{A}_{\alpha\beta}^{(i)} = \frac{m}{\rho^{(\text{ref})}}\int_{\mathbb{R}^3} n_i\xi_i\varphi_\alpha(\boldsymbol{\xi})\varphi_\beta(\boldsymbol{\xi})f^{(\text{ref})}\,\mathrm{d}\boldsymbol{c} \tag{III.66}$$

can be computed form the transport matrix in x-direction

$$\mathcal{A}_{\alpha\beta}^{(\boldsymbol{n})} = \sum_{\gamma,\delta\in I_M} \boldsymbol{T}_{\alpha\gamma}(\boldsymbol{n})\mathcal{A}_{\gamma\delta}^{(x)}\boldsymbol{T}_{\delta\beta}(\boldsymbol{n}), \tag{III.67}$$

that is, the equations are already fixed by considering a single direction.

Odd and even components. An important formulation is the grouping of polynomials into odd or even with respect to the normal component ξ_n using the transport matrix in normal direction. This will be useful in the formulation of boundary conditions. Every polynomial is either odd or even and we can split the set of polynomials into odd and even parts

$$\begin{aligned}
&\left\{\psi_\alpha\begin{pmatrix}\xi_n\\ \xi_{t_i}\end{pmatrix}\right\}_{\alpha\in I_M}\\
&= \left\{\frac{1}{2}\left(\psi_\alpha\begin{pmatrix}\xi_n\\ \xi_{t_i}\end{pmatrix} - \psi_\alpha\begin{pmatrix}-\xi_n\\ \xi_{t_i}\end{pmatrix}\right)\right\}_{\alpha\in I_M^{(\boldsymbol{n},o)}} \bigcup \quad \left\{\frac{1}{2}\left(\psi_\alpha\begin{pmatrix}\xi_n\\ \xi_{t_i}\end{pmatrix} + \psi_\alpha\begin{pmatrix}-\xi_n\\ \xi_{t_i}\end{pmatrix}\right)\right\}_{\alpha\in I_M^{(\boldsymbol{n},e)}}\\
&= \left\{\sum_{\beta\in I_M}\widetilde{T}_{\alpha\beta}(\boldsymbol{n})\psi_\beta(\boldsymbol{\xi})\right\}_{\alpha\in I_M^{(\boldsymbol{n},o)}} \bigcup \quad \left\{\sum_{\beta\in I_M}\widetilde{T}_{\alpha\beta}(\boldsymbol{n})\psi_\beta(\boldsymbol{\xi})\right\}_{\alpha\in I_M^{(\boldsymbol{n},e)}}\\
&=: \left\{\psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi})\right\}_{\alpha\in I_M^{(\boldsymbol{n},o)}} \bigcup \quad \left\{\psi_\alpha^{(\boldsymbol{n},e)}(\boldsymbol{\xi})\right\}_{\alpha\in I_M^{(\boldsymbol{n},e)}}
\end{aligned} \tag{III.68}$$

with index sets $I_M^{(\boldsymbol{n},o)} \cup I_M^{(\boldsymbol{n},e)} = I_M^{(\boldsymbol{n})}$ and $|I_M^{(\boldsymbol{n},o)}| = p$ and $|I_M^{(\boldsymbol{n},e)}| = q$. In a rotationally invariant basis with increasing polynomial degree (starting with the zeroth degree) the number of odd basis polynomials cannot be larger than the number of even ones, so q is smaller than p is impossible and for multi-dimensional cases we strictly have $p < q$.

The radial part $p_a^{(n)}(\boldsymbol{\xi})$ in (III.41) is always even, hence only the tensor indices decide if a basis polynomial is odd or not. If x corresponds to the normal direction, for a second order tensor splitting of the essential indices is $\{xy, xz\} \cup \{xx, yy, yz\}$, and for a third order tensor the splitting of the essential indices is $\{xxx, xyy, xyz\} \cup \{xxy, xxz, yyy, yyz\}$.

Moreover, the groupings of the odd and even basis polynomials $\varphi^{(\boldsymbol{n},o/e)}$ imply the odd and even groupings of the expansion coefficients in (III.46). Consequently, this leads to odd and even coefficients

$$\rho^{(\text{ref})} w_\alpha^{(\boldsymbol{n},o)} = m \int_{\mathbb{R}^3} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f(\boldsymbol{c}) \, \mathrm{d}\boldsymbol{c} \ \in \mathbb{R}^p, \quad \alpha \in I_M^{(\boldsymbol{n},o)} \tag{III.69}$$

$$\rho^{(\text{ref})} w_\alpha^{(\boldsymbol{n},e)} = m \int_{\mathbb{R}^3} \psi_\alpha^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) f(\boldsymbol{c}) \, \mathrm{d}\boldsymbol{c} \ \in \mathbb{R}^q, \quad \alpha \in I_M^{(\boldsymbol{n},e)}. \tag{III.70}$$

Combining this into a single ordered vector of variables we find

$$\begin{bmatrix} w_\alpha^{(\boldsymbol{n},o)} \\ w_\alpha^{(\boldsymbol{n},e)} \end{bmatrix} = \begin{bmatrix} \left(\sum_{\beta \in I_M} \widetilde{T}_{\alpha\beta}(\boldsymbol{n}) w_\beta \right)_{\alpha \in I_M^{(\boldsymbol{n},o)}} \\ \left(\sum_{\beta \in I_M} \widetilde{T}_{\alpha\beta}(\boldsymbol{n}) w_\beta \right)_{\alpha \in I_M^{(\boldsymbol{n},e)}} \end{bmatrix} = \sum_{\beta,\gamma \in I_M} P_{\alpha\beta} \widetilde{T}_{\beta\gamma}(\boldsymbol{n}) w_\gamma, \tag{III.71}$$

where $P_{\alpha\beta}$ is a permutation matrix which maps the list of indices $\{\beta\}_{\beta \in I_M}$ into the following order $\{\alpha\}_{\alpha \in I_M^{(o)}} \cup \{\alpha\}_{\alpha \in I_M^{(e)}}$ independent of the underlying coordinate system.

The Cartesian coordinate system allows us to identify odd and even components with respect to x-coordinate. Since ξ_x is odd and $\xi_{y/z}$ is even, the components of the matrix $\mathcal{A}_{\alpha\beta}^{(i)}$ in $i = x/y/z$ directions in (III.63) after permutation have the form

$$\begin{bmatrix} \mathbf{0} & \mathbb{A}^{(oe)} \\ \mathbb{A}^{(oe)^T} & \mathbf{0} \end{bmatrix}, \quad \begin{bmatrix} \mathbb{B}^{(y,o)} & \mathbf{0} \\ \mathbf{0} & \mathbb{B}^{(y,e)} \end{bmatrix}, \quad \begin{bmatrix} \mathbb{B}^{(z,o)} & \mathbf{0} \\ \mathbf{0} & \mathbb{B}^{(z,e)} \end{bmatrix} \tag{III.72}$$

with the sub-matrices

$$\mathbb{A}_{\alpha\beta}^{(oe)} = \sum_{\gamma,\delta \in I_M} P_{\alpha\gamma} \mathcal{A}_{\gamma\delta}^{(x)} P_{\delta\beta} = \int_{\mathbb{R}^3} \xi_x \varphi_\alpha^{(o)}(\boldsymbol{\xi}) \varphi_\beta^{(e)}(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{c}, \quad \alpha \in I_M^{(\boldsymbol{n},o)}, \beta \in I_M^{(\boldsymbol{n},e)} \tag{III.73}$$

$$\mathbb{B}_{\alpha\beta}^{(y,o/e)} = \sum_{\gamma,\delta \in I_M} P_{\alpha\gamma} \mathcal{A}_{\gamma\delta}^{(y)} P_{\delta\beta} = \int_{\mathbb{R}^3} \xi_y \varphi_\alpha^{(o/e)}(\boldsymbol{\xi}) \varphi_\beta^{(o/e)}(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{c}, \quad \alpha, \beta \in I_M^{(\boldsymbol{n},o/e)} \tag{III.74}$$

and analogously for z. These matrices do not depend on the coordinate system and consequently the integration could be taken over ξ_n and $\varphi^{(\boldsymbol{n},o/e)}$.

Hyperbolicity. The symmetrised system (III.63) is globally hyperbolic. Waves with with variations only in the normal direction n satisfy

$$\frac{\partial \widetilde{w}_\alpha}{\partial t} + v_x^{(\mathrm{ref})} \frac{\partial \widetilde{w}_\alpha}{\partial x} + \sum_{\beta \in I_M} \widetilde{\mathcal{A}}^{(\boldsymbol{n})}_{\alpha\beta} \frac{\partial \widetilde{w}_\beta}{\partial \boldsymbol{n}} = \frac{1}{\rho^{(\mathrm{ref})}} \widetilde{P}_\alpha \tag{III.75}$$

with transformed variables

$$\widetilde{w}_\alpha = \sum_{\beta \in I_M} S^{\frac{1}{2}}_{\alpha\beta} w^\beta, \quad \widetilde{\mathcal{A}}^{(\boldsymbol{n})}_{\alpha\beta} = \sqrt{\theta^{(\mathrm{ref})}} \sum_{\gamma,\delta \in I_M} S^{-\frac{1}{2}}_{\alpha\gamma} \mathcal{A}^{(\boldsymbol{n})}_{\gamma\delta} S^{-\frac{1}{2}}_{\delta\beta}, \quad \widetilde{P}_\alpha = \sum_{\beta \in I_M} S^{\frac{1}{2}}_{\alpha\beta} P_\beta \tag{III.76}$$

where the symmetric system matrix $\boldsymbol{\mathcal{A}}^{(\boldsymbol{n})}$ has the block structure as in (III.72). Its eigenvalues are formed from constants $c_j \in \mathbb{R}$ and read

$$\lambda_j^{\pm} = \pm\sqrt{\theta^{(\mathrm{ref})}} c_j \in \mathbb{R}, \quad \text{where } j = 1, \cdots, p, \quad \lambda_k = 0, \quad \text{where } k = 1, \cdots, q-p \tag{III.77}$$

depending on the number of odd p and even q variables. Consequently the speeds of propagation of the system in (III.75) are of the form

$$a_j^{(s)} = v_n^{(\mathrm{ref})} \pm \sqrt{\theta^{(\mathrm{ref})}} c_j \quad \text{where } j = 1, \cdots, p, \quad a_k^{(s)} = v_n^{(\mathrm{ref})}, \quad \text{where } k = 1, \cdots, q-p. \tag{III.78}$$

L^2-entropy law. The symmetrised system in (III.63) allows us to formulate the following L^2-type entropy balance law in the form

$$\frac{\partial \eta(\boldsymbol{W})}{\partial t} + \nabla_x \cdot \boldsymbol{h}(\boldsymbol{W}) = \sigma(\boldsymbol{W}) \tag{III.79}$$

with

$$\eta(\boldsymbol{W}) = \sum_{\alpha,\beta \in I_M} w_\alpha S_{\alpha\beta} w_\beta, \quad \boldsymbol{h}(\boldsymbol{W}) = \sum_{\alpha,\beta \in I_M} w_\alpha \boldsymbol{\mathcal{A}}_{\alpha\beta} w_\beta, \quad \sigma(\boldsymbol{W}) = \sum_{\alpha,\beta \in I_M} w_\alpha S_{\alpha\beta} P_\beta \tag{III.80}$$

as entropy function, its flux and its production. Writing this balance law for a bounded domain Ω gives

$$\frac{\partial}{\partial t} \int_\Omega \eta(\boldsymbol{W}) \, \mathrm{d}\boldsymbol{x} = \int_\Omega \sigma(\boldsymbol{W}) \, \mathrm{d}\boldsymbol{x} - \oint_\Omega \boldsymbol{n} \cdot \boldsymbol{h}(\boldsymbol{W}) \, \mathrm{d}\boldsymbol{n} \tag{III.81}$$

where the right-hand side should be non-positive. This implies restrictions for the production term $P_{\alpha\beta}$ but also for the boundary values $\boldsymbol{W}|_{\partial\Omega}$. The entropy η for the non-linear Boltzmann collision operator is $f\,lnf$ which is not an L^2-norm. However, the entropy $\eta = f^2$ is an L^2-norm and yields H-Theorem for the linearised Boltzmann collision but does not yield H-Theorem for the full non-linear Boltzmann collision operator. Therefore, the non-linear production terms $\sigma(\boldsymbol{W})$ in (III.81) requires special investigation.

Entropy-stable boundary conditions. If we ignore the production terms of the L^2-entropy (III.81), we require that

$$\oint_{\partial\Omega} \boldsymbol{n}\cdot\boldsymbol{h}(\boldsymbol{W})\,\mathrm{d}\boldsymbol{n} \geq 0 \tag{III.82}$$

holds for the stability of the boundary conditions imposed on $\boldsymbol{W}|_{\partial\Omega}$. Using (III.67) and (III.80) we rewrite the entropy flux term as follows

$$\boldsymbol{n}\cdot\boldsymbol{h}(W) = \sum_{\alpha,\beta\in I_M} w_\alpha \mathcal{A}^{(\boldsymbol{n})}_{\alpha\beta} w_\beta = \sum_{\alpha,\beta\in I_M}\sum_{\gamma,\delta\in I_M} w_\alpha \mathcal{A}^{(x)}_{\gamma\delta} T_{\gamma\alpha}(\boldsymbol{n}) T_{\delta\beta}(\boldsymbol{n}) w_\beta \tag{III.83}$$

$$= \sum_{\gamma,\delta\in I_M^{(\boldsymbol{n})}} w^{(\boldsymbol{n})}_\gamma \mathcal{A}^{(x)}_{\gamma\delta} w^{(\boldsymbol{n})}_\delta \tag{III.84}$$

which can be written

$$\boldsymbol{n}\cdot\boldsymbol{h}(\boldsymbol{W}) = 2\sum_{\alpha,\beta\in I_M^{(\boldsymbol{n},o/e)}} w^{(\boldsymbol{n},o)}_\alpha \mathbb{A}^{(oe)}_{\alpha\beta} w^{(\boldsymbol{n},e)}_\beta \tag{III.85}$$

with sub-matrix $\mathbb{A}_{\alpha\beta}\in\mathbb{R}^{p\times q}$ in (III.72). To satisfy stability (III.82) at the boundary we require

$$\sum_{\alpha,\beta\in I_M^{(\boldsymbol{n},o/e)}} w^{(\boldsymbol{n},o)}_\alpha \mathbb{A}^{(oe)}_{\alpha\beta} w^{(\boldsymbol{n},e)}_\beta \Big|_{\mathrm{BC}} \geq 0 \tag{III.86}$$

such that the boundary conditions are entropy-stable. Thermodynamically, this expression can be interpreted as a sum of products of thermodynamic fluxes $w^{(\boldsymbol{n},o)}_\alpha$ and forces $\sum_\beta \mathbb{A}_{\alpha\beta} w^{(\boldsymbol{n},e)}_\beta$. In fact, following the idea to specify thermodynamic fluxes at the boundary as a linear function of forces, we choose the conditions in the form

$$w^{(\boldsymbol{n},o)}_\alpha = \sum_{\beta,\gamma\in I_M^{(\boldsymbol{n},(o/e))}} L_{\alpha\beta}\mathbb{A}_{\beta\gamma} w^{(\boldsymbol{n},e)}_\gamma \tag{III.87}$$

with a symmetric positive definite matrix $L_{\alpha\beta}$. This expression can be considered to give boundary conditions of Onsager-type. Inserting these into (III.86) gives a sum of non-negative quadratic forms and a bounded L^2-entropy. The number of boundary conditions in (III.87) is p and corresponds to the number of characteristic waves going into the system for $v_n^{(\mathrm{ref})} = 0$ according to (III.78). In particular, all odd variables is equivalent to prescribing incoming characteristic variables [11].

Derivation of stable kinetic boundary conditions. For the derivation of the boundary conditions, we use the normal-tangent coordinate system (that is ξ_n, ξ_{t_i}) instead of the Cartesian coordinate system (ξ_x, ξ_y, ξ_z) to represent the variables at the boundary. Moreover, we require that the normal component n be pointing out of the gas. We will consider the case $v_n^{(\mathrm{ref})} = 0$ for the following derivation. In general, the incoming part of the distribution function can be

prescribed as a function of the incoming part

$$f(\boldsymbol{\xi}) = \begin{cases} \mathcal{K}_{\text{BC}}[f(\xi_n > 0)] \left(\begin{bmatrix} \xi_n \\ \xi_{t_i} \end{bmatrix} \right) & \xi_n < 0 \\ f \left(\begin{bmatrix} \xi_n \\ \xi_{t_i} \end{bmatrix} \right) & \xi_n > 0 \end{cases} \tag{III.88}$$

where $\mathcal{K}_{\text{BC}}$ is some convolution kernel [28]. To obtain boundary conditions for moments, the following Boltzmann-equation-projection style naive integration

$$w_\alpha|_{\text{BC}} = \int_{\mathbb{R}^3} \psi_\alpha(\boldsymbol{\xi}) f(\boldsymbol{\xi})|_{\text{BC}} \, \mathrm{d}\boldsymbol{\xi} \tag{III.89}$$

leads to an over-specification at the boundary and consequently ill-posed boundary conditions. We recall that according to (III.78) we only need p boundary conditions for the p incoming waves entering the domain. There are several methods to derive them namely specular reflections, prescribed influx function, Onsager-type, and Maxwell accommodation model.

Specular reflection. As the name indicates, in the case of specular reflection at the boundary the distribution function f is fully reflected

$$f(\boldsymbol{\xi})|_{BC} = \begin{cases} f \left(\begin{bmatrix} -\xi_n \\ \xi_{t_i} \end{bmatrix} \right) & \xi_n < 0 \\ f \left(\begin{bmatrix} \xi_n \\ \xi_{t_i} \end{bmatrix} \right) & \xi_n > 0, \end{cases} \tag{III.90}$$

hence, the distribution function at the boundary remains even. Splitting the distribution function into the odd and even parts with respect to ξ_n, the boundary conditions read

$$\begin{aligned} f^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) &:= \frac{f \left(\begin{bmatrix} \xi_n \\ \xi_{t_i} \end{bmatrix} \right) - f \left(\begin{bmatrix} -\xi_n \\ \xi_{t_i} \end{bmatrix} \right)}{2} \overset{!}{=} 0 \\ f^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) &:= \frac{f \left(\begin{bmatrix} \xi_n \\ \xi_{t_i} \end{bmatrix} \right) + f \left(\begin{bmatrix} -\xi_n \\ \xi_{t_i} \end{bmatrix} \right)}{2} \overset{!}{=} \text{anything} \end{aligned} \tag{III.91}$$

where we prescribe the boundary conditions only at the odd part and have no control over even part of the distribution. This corresponds to the fact that we only prescribe the incoming "half" of the distribution at the boundary. Splitting the distribution in (III.58) yields

$$f(\boldsymbol{\xi}) = \sum_{\alpha \in I_M^{(\boldsymbol{n},o)}} w_\alpha^{(\boldsymbol{n},o)} \varphi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}) + \sum_{\alpha \in I_M^{(\boldsymbol{n},e)}} w_\alpha^{(\boldsymbol{n},e)} \varphi_\alpha^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}) \tag{III.92}$$

using odd and even variables as in (III.71). Because we cannot prescribe any values for $w_\alpha^{(\boldsymbol{n},e)}$ at the boundary, we can conclude that

$$w_\alpha^{(\boldsymbol{n},o)}|_{\text{BC}} = \int_{\mathbb{R}^3} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f(\boldsymbol{\xi})|_{\text{BC}} \overset{!}{=} 0, \tag{III.93}$$

as the boundary conditions which yields the correct number of p conditions.

Prescribed influx boundary functions. Prescribing the influx distribution $f^{(\text{in})}$ in the place of the incoming half of the distribution

$$f(\boldsymbol{\xi})|_{\text{BC}} = \begin{cases} f^{(\text{in})}(\boldsymbol{\xi}) & \xi_n < 0 \\ f(\boldsymbol{\xi}) & \xi_n > 0, \end{cases} \tag{III.94}$$

which forms the building block for Maxwell accommodation model described below. Subtracting the even part $f^{(\boldsymbol{n},e)}(\boldsymbol{\xi})$ of the distribution $f(\boldsymbol{\xi})$ on each side yields only the odd part of the distribution

$$f^{(\boldsymbol{n},o)}(\boldsymbol{\xi})|_{\text{BC}} = \begin{cases} f_{\text{in}}(\boldsymbol{\xi}) - f^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) & \xi_n < 0 \\ f^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) & \xi_n > 0, \end{cases} \tag{III.95}$$

which is motivated from the specular reflection case above. Projection gives

$$w_\alpha^{(\boldsymbol{n},o)}|_{\text{BC}} = \int_{\mathbb{R}^3} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f(\boldsymbol{\xi})|_{\text{BC}} \, \mathrm{d}\boldsymbol{\xi} \tag{III.96}$$

$$= \int_{\mathbb{R}^3, c_n<0} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) \left(f^{(\text{in})}(\boldsymbol{\xi}) - f^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) \right) \, \mathrm{d}\boldsymbol{\xi} + \int_{\mathbb{R}^3, c_n>0} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} \tag{III.97}$$

that can be rearranged as

$$w_\alpha^{(\boldsymbol{n},o)}|_{\text{BC}} = 2 \int_{\mathbb{R}^3, c_n<0} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) \left(f^{(\text{in})}(\boldsymbol{\xi}) - f^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) \right) \, \mathrm{d}\boldsymbol{\xi} \tag{III.98}$$

which replaces the specular reflection case in (III.93). We identify an inhomogeneity

$$g_\alpha^{(\boldsymbol{n},\text{in})} = 2 \int_{\mathbb{R}^3, c_n<0} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f^{(\text{in})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} \tag{III.99}$$

and use the expansion of the even part of the distribution

$$f^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) = \sum_{\beta \in I_M^{\boldsymbol{n},e}} w_\beta^{(\boldsymbol{n},e)} \varphi_\beta^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) f^{(\text{ref})}(\boldsymbol{\xi}) \tag{III.100}$$

to find

$$w_\alpha^{(\boldsymbol{n},o)}|_{\text{BC}} = -2 \sum_{\beta \in I_M^{(\boldsymbol{n},e)}} \left(\int_{\mathbb{R}^3, c_n<0} \psi_\alpha^{(\boldsymbol{n},o)} \varphi_\beta^{(\boldsymbol{n},e)} f^{(\text{ref})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} \right) w_\beta^{(\boldsymbol{n},e)} + g_\alpha^{(\boldsymbol{n},\text{in})} \tag{III.101}$$

$$= -2 \sum_{\beta \in I_M^{(\boldsymbol{n},e)}} M_{\alpha\beta} w_\beta^{(\boldsymbol{n},e)} + g_\alpha^{(\boldsymbol{n},\text{in})} \tag{III.102}$$

where

$$M_{\alpha\beta} := \int_{\mathbb{R}^3, c_n<0} \psi_\alpha^{(\boldsymbol{n},o)} \varphi_\beta^{(\boldsymbol{n},e)} f^{(\text{ref})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi}. \tag{III.103}$$

As a consequence, we have odd moments at the boundary are given as a function of the even moments. Similar to the specular reflection case, now we have p boundary conditions for p incoming waves. Note that the integral of $M_{\alpha\beta}$ is rotationally invariant and the integral can be evaluated in the non-transformed (x, y, z)-based coordinate system.

Kinetic Onsager-type boundary conditions. The homogeneous part of (III.102) has almost the shape of the Onsager-type conditions in (III.87). To identify the matrix $\mathbb{A}_{\alpha\beta}$, we follow [11] and introduce the expansion

$$\xi_n \varphi_\beta^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) \approx \sum_{\gamma \in I^{(\boldsymbol{n},o)}} \varphi_\gamma^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) \int_{\mathbb{R}^3} \widehat{\xi}_n \psi_\gamma^{(\boldsymbol{n},o)}(\widehat{\boldsymbol{\xi}}) \varphi_\beta^{(\boldsymbol{n},e)} f^{(\mathrm{ref})} \, \mathrm{d}\widehat{\boldsymbol{\xi}} \tag{III.104}$$

which holds true when integrated against $\psi_\alpha^{(\boldsymbol{n},o)}$ for $\alpha \in I_M^{(\boldsymbol{n},o)}$, but is an approximation otherwise. Re-arranging the integral in (III.102) we find

$$\int_{\mathbb{R}^3, \xi_n < 0} \psi_\alpha^{(\boldsymbol{n},o)} \varphi_\beta^{(\boldsymbol{n},e)} f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{\xi} \tag{III.105}$$

$$= \int_{\mathbb{R}^3, \xi_n < 0} \frac{1}{\xi_n} \psi_\alpha^{(\boldsymbol{n},o)} \xi_n \varphi_\beta^{(\boldsymbol{n},e)} f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{\xi} \tag{III.106}$$

$$\approx \sum_{\gamma \in I_M^{(\boldsymbol{n},o)}} \left(\int_{\mathbb{R}^3, \xi_n < 0} \frac{1}{\xi_n} \psi_\alpha^{(\boldsymbol{n},o)} \varphi_\gamma^{(\boldsymbol{n},o)} f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{\xi} \right) \left(\int_{\mathbb{R}^3} \xi_n \psi_\gamma^{(\boldsymbol{n},o)} \varphi_\beta^{(\boldsymbol{n},e)} f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{\xi} \right) \tag{III.107}$$

$$= \sum_{\gamma \in I_M^{(\boldsymbol{n},o)}} \left(\int_{\mathbb{R}^3, \xi_n < 0} \frac{1}{\xi_n} \psi_\alpha^{(\boldsymbol{n},o)} \psi_\gamma^{(\boldsymbol{n},o)} f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{\xi} \right) \left(\int_{\mathbb{R}^3} \xi_n \varphi_\gamma^{(\boldsymbol{n},o)} \varphi_\beta^{(\boldsymbol{n},e)} f^{(\mathrm{ref})} \, \mathrm{d}\boldsymbol{\xi} \right) \tag{III.108}$$

where we used the symmetriser matrix for the odd polynomials with

$$\varphi_\beta^{(\boldsymbol{n},o)} = \sum_{\gamma \in I^{(\boldsymbol{n},o)}} S_{\beta\gamma}^{(\boldsymbol{n},o)} \psi_\gamma^{(\boldsymbol{n},o)}$$

which allows us to switch between polynomials $\varphi_\beta^{(\boldsymbol{n},o)}$ and $\psi_\beta^{(\boldsymbol{n},o)}$ easily. The boundary matrix in (III.102) is replaced by

$$M_{\alpha\beta}^{(\mathrm{Onsager})} = \sum_{\gamma \in I_M^{(\boldsymbol{n},o)}} L_{\alpha\gamma} \mathbb{A}_{\gamma\beta} \tag{III.109}$$

with

$$L_{\alpha\beta} := \int_{\mathbb{R}^3, \xi_n < 0} \frac{1}{\xi_n} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) \varphi_\beta^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f^{(\mathrm{ref})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} \tag{III.110}$$

which is a symmetric positive definite matrix. The division by ξ_n is not problematic because the polynomials in the integrand are of odd degree. Ultimately, this gives stable boundary conditions for odd moments. Stability of boundary conditions including the inhomogeneous part require some technical conditions and has been discussed in [29].

Maxwell accommodation model. The most widely used wall boundary condition in kinetic theory is the accommodation model of Maxwell [7, 28] given by

$$f(\boldsymbol{\xi})|_{\mathrm{BC}} = \begin{cases} \chi f^{(\mathrm{in})}(\boldsymbol{\xi}) + (1-\chi) f\left(\begin{bmatrix} -\xi_n \\ \xi_{t_i} \end{bmatrix} \right) & \xi_n < 0 \\ f(\boldsymbol{\xi}) & \xi_n > 0 \end{cases} \tag{III.111}$$

which is a combination of a prescribed influx and specular reflections. The parameter χ is called accommodation coefficient, satisfies $0 \leq \chi \leq 1$ and is a property of the boundary wall. Subtracting again the even part of the distribution $f^{(\boldsymbol{n},e)}(\boldsymbol{\xi})$ on both sides and using the manipulations from above it is straightforward to derive the condition

$$w_\alpha^{(\boldsymbol{n},o)}|_{\text{BC}} = \frac{2\chi}{2-\chi} \int_{\mathbb{R}^3, \xi_n<0} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) \left(f^{(\text{in})}(\boldsymbol{\xi}) - f^{(\boldsymbol{n},e)}(\boldsymbol{\xi}) \right) \, \mathrm{d}\boldsymbol{\xi} \tag{III.112}$$

after projecting with the odd polynomials $\psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi})$. Using the re-definition of the inhomogeneity

$$g_\alpha^{(\text{in},\boldsymbol{n})}(\chi) = \frac{2\chi}{2-\chi} \int_{\mathbb{R}^3, \xi_n<0} \psi_\alpha^{(\boldsymbol{n},o)}(\boldsymbol{\xi}) f^{(\text{in})}(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi} \tag{III.113}$$

the boundary conditions can be written

$$w_\alpha^{(\boldsymbol{n},o)}|_{\text{BC}} = -\frac{2\chi}{2-\chi} \sum_{\beta \in I_M^{(\boldsymbol{n},e)}} M_{\alpha\beta} w_\beta^{(\boldsymbol{n},e)} + g_\alpha^{(\text{in},\boldsymbol{n})}(\chi) \tag{III.114}$$

with the matrix $M_{\alpha\beta}$ as in (III.102). Stability of the accommodation model is obtained by using the matrix in (III.110) and the replacement in (III.109) together with the fact that $(2\chi)/(2-\chi) \geq 0$.

III.2.3. Perturbation kinetics with 13-moment equations

Expansion of the 13-Moment distribution. The 13-Moment case is obtained by the polynomial expansion

$$f^{(W13)}(\boldsymbol{c}) := \left(w^{(0)}\psi^{(0)}(\boldsymbol{c}) + w_i^{(0)}\psi_i^{(0)}(\boldsymbol{c}) + w^{(1)}\psi^{(1)}(\boldsymbol{c}) + w_{ij}^{(0)}\psi_{ij}^{(0)}(\boldsymbol{c}) + w_i^{(1)}\psi_i^{(1)}(\boldsymbol{c}) \right) f^{(\text{ref})}(\boldsymbol{c}) \tag{III.115}$$

where $f^{(\text{ref})}$ is the reference distribution in (III.45) and $\psi_\bullet^{(\bullet)}$ are the orthogonal basis functions and $w_\bullet^{(\bullet)}$ are the expansion coefficients. The coefficient variables

$$\left[w^{(0)}, w_i^{(0)}, w^{(1)}, w_{ij}^{(0)}, w_i^{(1)} \right] \tag{III.116}$$

when written in explicit component form yields the 13 moment variables

$$W = \left[w^0, w_x^{(0)}, w_y^{(0)}, w_z^{(0)}, w^{(1)}, w_{xx}^{(0)}, w_{xy}^{(0)}, w_{xz}^{(0)}, w_{yy}^{(0)}, w_{yz}^{(0)}, w_x^{(1)}, w_y^{(1)}, w_z^{(1)} \right]. \tag{III.117}$$

13-Moment basis functions. The orthogonal polynomial basis functions are given by

$$\psi^{(0)}(\boldsymbol{c}) = 1 \tag{III.118}$$

$$\psi_i^{(0)}(\boldsymbol{c}) = c_i \tag{III.119}$$

$$\psi^{(1)}(\boldsymbol{c}) = \frac{3}{2} - \frac{1}{2}\boldsymbol{c}^2 \tag{III.120}$$

$$\psi_{ij}^{(0)}(\boldsymbol{c}) = c_{\langle i} c_{j \rangle}, \tag{III.121}$$

$$\psi_i^{(1)}(\boldsymbol{c}) = c_i \left(\frac{5}{2} - \frac{1}{2}\boldsymbol{c}^2 \right) \tag{III.122}$$

where the index variables i, j denote directions (x, y, z) of a tensor.

Mapping between 13-Moment coefficients and macroscopic variables. The distribution (III.115) provides the mapping of the 13 coefficients (III.116) to the variables $(\rho, v_i, \theta, \sigma_{ij}, q_i)$. These can be calculated using (I.5).

$$w^{(0)} = \frac{\rho}{\rho^{(\text{ref})}} \tag{III.123}$$

$$w_i^{(0)} = \frac{\rho \Delta v_i}{\rho^{(\text{ref})}\sqrt{\theta^{(\text{ref})}}} \tag{III.124}$$

$$w^{(1)} = -\frac{3}{2}\frac{\rho\Delta\theta}{\rho^{(\text{ref})}\theta^{(\text{ref})}} - \frac{1}{2}\frac{\rho\Delta v_i^2}{\rho^{(\text{ref})}\theta^{(\text{ref})}} \tag{III.125}$$

$$w_{ij}^{(0)} = \frac{\sigma_{ij}}{\rho^{(\text{ref})}\theta^{(\text{ref})}} + \frac{\rho\Delta v_{\langle i}\Delta v_{j\rangle}}{\rho^{(\text{ref})}\theta^{(\text{ref})}} \tag{III.126}$$

$$w_i^{(1)} = -\frac{q_i}{\rho^{(\text{ref})}\sqrt{\theta^{(\text{ref})}}^3} - \frac{\sigma_{ij}\Delta v_j}{\rho^{(\text{ref})}\sqrt{\theta^{(\text{ref})}}^3} - \frac{5}{2}\frac{\rho\Delta\theta\Delta v_i}{\rho^{(\text{ref})}\sqrt{\theta^{(\text{ref})}}^3} - \frac{1}{2}\frac{\rho\Delta v_i\Delta v^2}{\rho^{(\text{ref})}\sqrt{\theta^{(\text{ref})}}^3} \tag{III.127}$$

where $\Delta v_i := v_i = v_i^{(\text{ref})}$ and $\Delta\theta := \theta - \theta^{(\text{ref})}$.

13-Moment system. Projecting the Equation (III.115) on the Boltzmann equation yields the following evolution equations for the 13 moment coefficients

$$\frac{\partial w^{(0)}}{\partial t} + v_k^{(r)}\frac{\partial w^{(0)}}{\partial x_k} + \sqrt{\theta^{(\text{ref})}}\frac{\partial w_j^{(0)}}{\partial x_j} = 0 \tag{III.128a}$$

$$\frac{\partial w_i^{(0)}}{\partial t} + v_k^{(r)}\frac{\partial w_i^{(0)}}{\partial x_k} + \sqrt{\theta^{(\text{ref})}}\frac{\partial w^{(0)}}{\partial x_i} - \frac{2}{3}\sqrt{\theta^{(\text{ref})}}\frac{\partial w^{(1)}}{\partial x_i} + \sqrt{\theta^{(\text{ref})}}\frac{\partial w_{ij}^{(0)}}{\partial x_j} = 0 \tag{III.128b}$$

$$\frac{\partial w^{(1)}}{\partial t} + v_k^{(\text{ref})}\frac{\partial w^{(1)}}{\partial x_k} - \sqrt{\theta^{(\text{ref})}}\frac{\partial w_j^{(0)}}{\partial x_j} + \sqrt{\theta^{(\text{ref})}}\frac{\partial w_j^{(1)}}{\partial x_j} = 0 \tag{III.128c}$$

$$\frac{\partial w_{ij}^{(0)}}{\partial t} + v_k^{(\text{ref})}\frac{\partial w_{ij}^{(0)}}{\partial x_k} + 2\sqrt{\theta^{(\text{ref})}}\frac{\partial w_{\langle j}^{(0)}}{\partial x_i\rangle} - \frac{4}{5}\sqrt{\theta^{(\text{ref})}}\frac{\partial w_{\langle j}^{(1)}}{\partial x_j\rangle} = Q_{ij}^{(0)} \tag{III.128d}$$

$$\frac{\partial w_i^{(1)}}{\partial t} + v_k^{(\text{ref})}\frac{\partial w_i^{(1)}}{\partial x_k} + \frac{5}{3}\sqrt{\theta^{(\text{ref})}}\frac{\partial w^{(1)}}{\partial x_i} - \sqrt{\theta^{(\text{ref})}}\frac{\partial w_{ij}^{(0)}}{\partial x_j} = Q_i^{(1)} \tag{III.128e}$$

where the right-hand side productions terms are evaluated based on the ES-BGK model with relaxation time τ and correction for the Prandtl number [7],

$$Q_{ij}^{(0)} = -\frac{1}{\tau}\left(w_{ij}^{(0)} - w_{ij}^{(0)}|_{\text{eq}}\right) \qquad Q_i^{(1)} = -\frac{1}{\tau}\left(w_i^{(1)} - w_i^{(1)}|_{\text{eq}}\right). \tag{III.129}$$

The relaxation time becomes Knudsen number when the above moment system is written in dimensionless form. By neglecting the non-equilibrium variables — $\sigma_{ij} = 0$, $q_i = 0$ — the equilibrium values can be computed from Equations (III.123) - (III.125). The left hand side of the above system contains only linear differential operators whose coefficients depend only on the constant reference values.

Because pressure tensor and heat flux are exactly reproduced by the expansion, the first three coefficient equations in (III.128) are fully equivalent to conservation laws of mass, momentum, and energy equations

$$\frac{D\rho}{Dt} + \rho\frac{\partial v_i}{\partial x_i} = 0 \tag{III.130}$$

$$\rho\frac{Dv_i}{Dt} + \rho\frac{\partial \theta}{\partial x_i} + \theta\frac{\partial \rho}{\partial x_i} + \frac{\partial \sigma_{ij}}{\partial x_j} = 0 \tag{III.131}$$

$$\frac{3}{2}\rho\frac{D\theta}{Dt} + p\frac{\partial v_i}{\partial x_i} + \sigma_{ij}\frac{\partial v_j}{\partial x_i} + \frac{\partial q_i}{\partial x_i} = 0 \tag{III.132}$$

for any choice of the reference parameters $\rho^{(\text{ref})}$, $v^{(\text{ref})}$, and $\theta^{(\text{ref})}$. The non-linear mappings between the classical fields and moment coefficients turns the non-linear conservation laws into linear equation. However, in the case of 13 coefficients the equations for the coefficients $w_{ij}^{(0)}$ and $w_i^{(1)}$ are not equivalent to the transfer equations of pressure tensor and heat flux because the higher order moments are closed by the distribution yielding non-trivial expressions depending on the reference parameters $\rho^{(\text{ref})}$, $v_i^{(\text{ref})}$, and $\theta^{(\text{ref})}$. Consequently, the first order Chapman-Enskog expansion is affected as well by the reference state. If we define a dimensionless reference deviation ϵ such that

$$\frac{\|\Delta \boldsymbol{v}\|}{\sqrt{\theta^{(\text{ref})}}} = \mathcal{O}(\epsilon), \quad \frac{\Delta\theta}{\theta^{(\text{ref})}} = \mathcal{O}(\epsilon), \tag{III.133}$$

it is possible to obtain the laws of Navier-Stokes and Fourier in the form

$$\sigma_{ij} = -2\text{Kn}\frac{\partial v_{\langle i}}{\partial x_{j\rangle}} + \mathcal{O}(\epsilon^2\text{Kn}) + \mathcal{O}(\text{Kn}^2) \tag{III.134}$$

$$q_i = -\frac{15}{4}\text{Kn}\frac{\partial \theta}{\partial x_i} + \mathcal{O}(\epsilon^2\text{Kn}) + \mathcal{O}(\text{Kn}^2) \tag{III.135}$$

from the theory of 13 moment coefficient equations for $\epsilon,\ \text{Kn} \to 0$. These relations show that the usual second order accuracy of the Chapman-Enskog expansion has an extra term proportional to ϵKn^2. To retain the second order accuracy for the expansion, the reference value must be chosen such that $\epsilon \sim \sqrt{\text{Kn}}$. Note that larger moment systems will result in a similar expansion results, but with increase powers of ϵ in the linear term. This remains to be studied [24].

We further note that we use the following production terms in Equation (III.128)

$$Q_{ij}^{(0)} := \frac{1}{\tau}\left(-w^{(0)}w_{ij}^{(0)} + w_{\langle i}^{(0)}w_{j\rangle}^{(0)}\right) \tag{III.136a}$$

$$Q_i^{(1)} := \frac{1}{\tau}\left(-w^{(0)}w_i^{(1)} + \frac{10}{3}w^{(1)}w_i^{(0)} + \frac{1}{3}w_i^{(0)}w_{ij}^{(0)}\right), \tag{III.136b}$$

which are obtained in [30], to simulate weakly non-equilibrium processes.

13-Moment symmetriser. The zeroth and first order tensors — scalar and vectors — do not require symmetrisation. Therefore,

$$S^{(0)}_{\alpha\beta} = 1, \qquad S^{(1)}_{\alpha\beta} = 1 \qquad \alpha, \beta \in \{x, y\}\,. \tag{III.137}$$

The second order tensors do require symmetrisation, and the symmetriser is given by

$$S^{(2)}_{\alpha\beta} = \begin{bmatrix} 1 & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ \frac{1}{2} & 0 & 1 \end{bmatrix} \qquad \alpha, \beta \in \{xx, xy, yy\}\,. \tag{III.138}$$

Using Equations (III.137) and (III.138), the G13 symmetriser is given by

$$\boldsymbol{S} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & S^{(2)}_{\alpha\beta} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{III.139}$$

13-Moment coordinate transformation matrix. The Transformation matrix of a zeroth order tensor — scalar — is given by

$$\widetilde{T}^{(0)}_{\alpha\beta}(\boldsymbol{n}) = 1, \tag{III.140}$$

and the transformation of the first order tensor — vector — is given by

$$\widetilde{T}^{(1)}_{\alpha\beta}(\boldsymbol{n}) = \begin{bmatrix} n_x & n_y & n_z \\ t^{(1)}_x & t^{(1)}_y & t^{(1)}_z \\ t^{(2)}_x & t^{(2)}_y & t^{(2)}_z \end{bmatrix}, \tag{III.141}$$

and the second-order tensor — matrix — is given by

$$\widetilde{T}^{(2)}_{\alpha\beta}(\boldsymbol{n}) = \begin{bmatrix} 0 & 2n_x n_y & 2n_x^2 & n_y^2 - n_x^2 & 2n_x n_y \\ n_x t^{(1)}_x - n_x t^{(1)}_z & n_y t^{(1)}_x + n_x t^{(1)}_y & n_x t^{(1)}_z + n_x t^{(1)}_x & n_y t^{(1)}_y - n_x t^{(1)}_z & n_x t^{(1)}_y + n_y t^{(1)}_z \\ n_x t^{(2)}_x - n_x t^{(2)}_z & n_y t^{(2)}_x + n_x t^{(2)}_y & n_x t^{(2)}_z + n_x t^{(2)}_x & n_y t^{(2)}_y - n_x t^{(2)}_z & n_x t^{(2)}_y + n_y t^{(2)}_z \\ \left(t^{(1)}_x\right)^2 - \left(t^{(1)}_z\right)^2 & 2t^{(1)}_x t^{(1)}_y & 2t^{(1)}_x t^{(1)}_z & \left(t^{(1)}_y\right)^2 - \left(t^{(1)}_z\right)^2 & 2t^{(1)}_y t^{(1)}_z \\ t^{(1)}_x t^{(2)}_x - t^{(1)}_z t^{(2)}_z & t^{(1)}_y t^{(2)}_x + t^{(1)}_x t^{(2)}_y & t^{(1)}_z t^{(2)}_x + t^{(1)}_x t^{(2)}_z & t^{(1)}_y t^{(2)}_y - t^{(1)}_z t^{(2)}_z & t^{(1)}_z t^{(2)}_y + t^{(1)}_y t^{(2)}_z \end{bmatrix}. \tag{III.142}$$

Transformed components read

$$\begin{bmatrix} w^{(0)}_n & w^{(0)}_{t_1} & w^{(0)}_{t_2} \end{bmatrix}^T = \widetilde{\boldsymbol{T}}^{(1)}(\boldsymbol{n}) \cdot \begin{bmatrix} w^{(0)}_x & w^{(0)}_y & w^{(0)}_z \end{bmatrix}^T \tag{III.143}$$

$$\begin{bmatrix} w^{(0)}_{nn}, w^{(0)}_{nt_1}, w^{(0)}_{nt_2}, w^{(0)}_{t_1t_1}, w^{(0)}_{t_1t_2} \end{bmatrix}^T = \widetilde{\boldsymbol{T}}^{(2)}(\boldsymbol{n}) \cdot \begin{bmatrix} w^{(0)}_{xx}, w^{(0)}_{xy}, w^{(0)}_{xz}, w^{(0)}_{yy}, w^{(0)}_{yz} \end{bmatrix}^T \tag{III.144}$$

Entropy-stable 13-moment boundary conditions. After transforming the 13 moment coefficients into normal-tangential coordinate system at the boundary, they become:

$$w_\alpha^{(\boldsymbol{n},o)} = \left[w_n^{(0)},\ \ w_{nt_1}^{(0)},\ \ w_{nt_2}^{(0)},\ \ w_n^{(1)}\right] \tag{III.145}$$

$$w_\alpha^{(\boldsymbol{n},e)} = \left[w^{(0)},\ \ w_{t_1}^{(0)},\ \ w_{t_2}^{(0)},\ \ w^{(1)},\ \ w_{nn}^{(0)},\ \ w_{t_1t_1}^{(0)},\ \ w_{t_2t_2}^{(0)},\ \ w_{t_1}^{(1)},\ \ w_{t_2}^{(1)}\right]. \tag{III.146}$$

This gives rise to the transport sub-matrix $\mathbb{A}^{(\mathrm{oe})}$ in Equation (III.73) as

$$\begin{bmatrix} 1 & 0 & 0 & -\frac{2}{3} & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{2}{5} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{2}{5} \\ 0 & 0 & 0 & \frac{2}{3} & -\frac{2}{5} & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{III.147}$$

And the classical boundary matrix as in Equation (III.102) is given by

$$M_{\alpha\beta} = \frac{1}{\sqrt{2\pi}} \begin{bmatrix} 1 & 0 & 0 & -\frac{1}{3} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{5} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{5} \\ \frac{1}{2} & 0 & 0 & \frac{7}{6} & -\frac{1}{4} & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{III.148}$$

while the Onsager coupling matrix from Equation (III.110) is given by

$$L_{\alpha\beta} = \frac{1}{\sqrt{2\pi}} \begin{bmatrix} 1 & 0 & 0 & \frac{1}{2} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{2} & 0 & 0 & \frac{9}{4} \end{bmatrix}. \tag{III.149}$$

The Onsager boundary matrix below is obtained from Equations (III.147) and (III.149)

$$M_{\alpha\beta}^{(\mathrm{Onsager})} = \frac{1}{\sqrt{2\pi}} \sum_{\gamma \in I_M^{(\boldsymbol{n},o)}} L_{\alpha\gamma}\mathbb{A}_{\gamma\beta} = \frac{1}{\sqrt{2\pi}} \begin{bmatrix} 1 & 0 & 0 & -\frac{1}{3} & \frac{4}{5} & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{2}{5} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{2}{5} \\ \frac{1}{2} & 0 & 0 & \frac{7}{6} & -\frac{2}{5} & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{III.150}$$

The inhomogeneity g_α as in Equation (III.113) reads

$$g_\alpha^{(\mathrm{in},\boldsymbol{n})} = \frac{2\chi}{2-\chi} \begin{bmatrix} w^{(0),\mathrm{in}} - \frac{1}{3}w^{(1),\mathrm{in}} \\ w_{t_1}^{(0),\mathrm{in}} \\ w_{t_2}^{(0),\mathrm{in}} \\ \frac{1}{2}w^{(0),\mathrm{in}} + \frac{7}{6}w^{(1),\mathrm{in}} \end{bmatrix}. \tag{III.151}$$

Relation to Grad's original moment equations. The expansion (I.6) can be viewed as a perturbation of the reference distribution $f^{(\mathrm{ref})}$. The constant reference parameters density $\rho^{(\mathrm{ref})}$, bulk velocity $\boldsymbol{v}^{(\mathrm{ref})}$ and temperature $\theta^{(\mathrm{ref})}$ can be chosen arbitrarily and any choice will yield

a linear system of moment equations for the choice of expansion coefficients $\boldsymbol{W}$. The actual local values of density, bulk velocity and temperature follow from the integrals

$$\rho = \int_{\mathbb{R}^3} mf \, \mathrm{d}\boldsymbol{c}, \qquad \rho v_i = \int_{\mathbb{R}^3} mc_i f \, \mathrm{d}\boldsymbol{c}, \qquad \frac{3}{2}\rho\theta = \int_{\mathbb{R}^3} \frac{m}{2}(c_i - v_i)^2 f \, \mathrm{d}\boldsymbol{c} \tag{III.152}$$

and depend on the values of W.They coincide with the reference parameters only if $w^{(0)} = 1$, $w_i^{(0)} = w^{(1)} = 0$. It is tempting to choose the local density, velocity, and temperature as reference values, which turns the globally constant distribution expansion (I.6) into the classical Grad distribution. In this case the relations

$$w^{(0)} = 1, w_i^{(0)} = w^{(1)} = 0 \tag{III.153}$$

act as a constraint in order to ensure (III.43) for the expansion. However, the derivation of the expansion coefficients (III.48) and (III.63) then becomes incomplete and must be corrected by spatial and temporal derivatives of ρ, v_i, and θ from the reference distribution. This will yield a non-linear transport operator known as Grad's moment equations. These resulting equations suffer from the loss of hyperbolicity [31].

Advantages over Grad's approach. The classical Grad's moment equations, as discussed above, are restricted to near-equilibrium processes because of loss of hyperbolicity. The approach of perturbing around a constant reference distribution $f^{(\mathrm{ref})}$ also comes with the restriction of small reference deviations. However, there are certain advantages of using globally constant reference distribution in approximation.

- **mathematical structure:** In contrast to Grad's equation, the systems (III.128) exhibit a convincing structure - linear terms on the left hand side and at most quadratic terms on the right hand side (see (III.136)) when using the Boltzmann collision operator.
- **numerical simplicity:** Due the mathematical structure, the equations are very easy to implement and simulate. In contrast, Grad's equations contain non-linear terms which are difficult to discretise and attributed to cause of instabilities.
- **non-linear capability:** The above discussions also demonstrate that non-linear effects, like dissipation and turbulence, are covered provided that the deviations $\Delta\theta$ and $\Delta\boldsymbol{v}$ are small – weakly non-linear processes.

The system (III.128) appears to be very similar to linearised moment equations, which have similar structures, simplicity, and robustness. However, using linearised Grad moment system will not be able to describe even weakly non-linear processes.

The main disadvantage of using globally constant approximation is that simulating strongly non-linear flows will be difficult. The choices of reference values $\rho^{(\mathrm{ref})}$, $\boldsymbol{v}^{(\mathrm{ref})}$, and $\theta^{(\mathrm{ref})}$ must not deviate much from the local values. Fortunately, in many weakly non-linear processes, this is equivalent to low local Mach numbers which is valid for instance in micro-flows. In such processes a state given by initial or boundary conditions will then be an appropriate reference choice.

III.3. Development of generic FEM numerical framework

III.3.1. Generic weak formulation

This subsection is adapted from [32]. Consider the following abstract first-order system

$$\begin{aligned} \boldsymbol{A}^{(x)}\frac{\partial \boldsymbol{W}}{dx} + \boldsymbol{A}^{(y)}\frac{\partial \boldsymbol{W}}{dy} + \boldsymbol{P}\boldsymbol{W} &= \boldsymbol{F}, && \text{in } \Omega \\ \boldsymbol{W}^{(\boldsymbol{n},\boldsymbol{o})} &= \boldsymbol{L}\mathbb{A}\boldsymbol{W}^{(\boldsymbol{n},\boldsymbol{e})} + \boldsymbol{g}, && \text{on } \Gamma \end{aligned} \tag{III.154}$$

For the sake of convenience we define $\boldsymbol{A} := \begin{bmatrix} \boldsymbol{A}^{(x)} \\ \boldsymbol{A}^{(y)} \end{bmatrix}$, $\boldsymbol{\nabla W} := \begin{bmatrix} \frac{d}{dx} \\ \frac{d}{dy} \end{bmatrix} \boldsymbol{W} = \begin{bmatrix} \frac{d\boldsymbol{W}}{dx} \\ \frac{d\boldsymbol{W}}{dy} \end{bmatrix}$.

After substitution the Equation (III.154) becomes

$$\boldsymbol{A}\cdot\boldsymbol{\nabla W} + \boldsymbol{P}\boldsymbol{W} = \boldsymbol{F}. \tag{III.155}$$

Multiplying (III.155) with appropriate test vector $\boldsymbol{V}$ yields

$$\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla W}\right)\,\mathrm{d}\boldsymbol{x} + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}. \tag{III.156}$$

To maintain bi-linearity we split the first term into two halves and we get

$$\begin{aligned} \frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla W}\right)\,\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla W}\right)\,\mathrm{d}\boldsymbol{x} \\ + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}. \end{aligned} \tag{III.157}$$

We apply integration-by-parts technique to the second term

$$\begin{aligned} \frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla W}\right)\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \left(\boldsymbol{\nabla V}^T\cdot\boldsymbol{A}\right)\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int_\Omega \boldsymbol{\nabla}\cdot\left(\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{W}\right)\,\mathrm{d}\boldsymbol{x} \\ + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}. \end{aligned} \tag{III.158}$$

Applying Gauss theorem to transform volume integral to a surface integral yields

$$\begin{aligned} \frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla W}\right)\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \left(\boldsymbol{\nabla V}^T\cdot\boldsymbol{A}\right)\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int_\Gamma \left(\boldsymbol{V}^T\boldsymbol{A}\boldsymbol{W}\right)\cdot\boldsymbol{n}\,\mathrm{d}\boldsymbol{l} \\ + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}. \end{aligned} \tag{III.159}$$

We simplify the boundary integral term in Equation (III.159) by

$$\begin{aligned} \left(\boldsymbol{V}^T\begin{bmatrix}\boldsymbol{A}^{(x)}\\ \boldsymbol{A}^{(y)}\end{bmatrix}\boldsymbol{W}\right)\cdot\boldsymbol{n} &= \begin{bmatrix}\boldsymbol{V}^T\boldsymbol{A}^{(x)}\boldsymbol{W}\\ \boldsymbol{V}^T\boldsymbol{A}^{(y)}\boldsymbol{W}\end{bmatrix}\cdot\begin{bmatrix}n_x\\ n_y\end{bmatrix} \\ &= \boldsymbol{V}^T\boldsymbol{A}^{(x)}n_x\boldsymbol{W} + \boldsymbol{V}^T\boldsymbol{A}^{(y)}n_y\boldsymbol{W} \\ &= \boldsymbol{V}^T\underbrace{\left(\boldsymbol{A}^{(x)}n_x + \boldsymbol{A}^{(y)}n_y\right)}_{:=\boldsymbol{A}^{(\boldsymbol{n})}}\boldsymbol{W}. \end{aligned} \tag{III.160}$$

Substituting the simplified form (III.160) in the boundary integral in (III.159) yields

$$\begin{aligned}\frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\boldsymbol{W}\right)\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \left(\boldsymbol{\nabla}\boldsymbol{V}^T\cdot\boldsymbol{A}\right)\boldsymbol{W}\,\mathrm{d}\boldsymbol{x}\\ + \frac{1}{2}\int_\Gamma \boldsymbol{V}^T\boldsymbol{A}^{(n)}\boldsymbol{W}\,\mathrm{d}\boldsymbol{l}\\ + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}.\end{aligned} \tag{III.161}$$

To exploit the structure of $\boldsymbol{A}$ we transform the variables in normal-tangent coordinate system

$$\begin{aligned}\boldsymbol{V}^T\boldsymbol{A}^{(n)}\boldsymbol{W} &= (\underbrace{\boldsymbol{T}_n^{-1}\boldsymbol{P}^T\overbrace{\boldsymbol{P}\boldsymbol{T}_n}^{:=\boldsymbol{V}^{(n)}}}_{=\boldsymbol{I}}\boldsymbol{V})^T\boldsymbol{A}^{(n)}\underbrace{\boldsymbol{T}_n^{-1}\boldsymbol{P}^T\overbrace{\boldsymbol{P}\boldsymbol{T}_n}^{:=\boldsymbol{W}^{(n)}}}_{=\boldsymbol{I}}\boldsymbol{W}\\ &= (\boldsymbol{T}_n^{-1}\boldsymbol{P}^T\boldsymbol{V}^{(n)})^T\boldsymbol{A}^{(n)}\boldsymbol{T}_n^{-1}\boldsymbol{P}^T\boldsymbol{W}^{(n)}\\ &= \underbrace{\boldsymbol{V}^{(n)^T}}_{=\begin{bmatrix}\boldsymbol{V}^{(n,o)^T}\\ \boldsymbol{V}^{(n,e)^T}\end{bmatrix}^T}\underbrace{\boldsymbol{P}\boldsymbol{T}_n\boldsymbol{A}^{(n)}\boldsymbol{T}_n^{-1}\boldsymbol{P}^T}_{=\begin{bmatrix}\boldsymbol{0} & \bar{\mathbb{A}}\\ \bar{\mathbb{A}}^T & \boldsymbol{0}\end{bmatrix}}\underbrace{\boldsymbol{W}^{(n)}}_{=\begin{bmatrix}\boldsymbol{W}^{(n,o)}\\ \boldsymbol{W}^{(n,e)}\end{bmatrix}}\\ &= \begin{bmatrix}\boldsymbol{V}^{(n,o)^T} & \boldsymbol{V}^{(n,e)^T}\end{bmatrix}\begin{bmatrix}\boldsymbol{0} & \bar{\mathbb{A}}\\ \bar{\mathbb{A}}^T & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{W}^{(n,o)}\\ \boldsymbol{W}^{(n,e)}\end{bmatrix}\\ &= \boldsymbol{V}^{(n,o)^T}\bar{\mathbb{A}}\boldsymbol{W}^{(n,e)} + \boldsymbol{V}^{(n,e)^T}\bar{\mathbb{A}}^T\boldsymbol{W}^{(n,o)}.\end{aligned} \tag{III.162}$$

We substitute the simplified result of (III.162) in the boundary integral in (III.161)

$$\begin{aligned}\frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\boldsymbol{W}\right)\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \left(\boldsymbol{\nabla}\boldsymbol{V}^T\cdot\boldsymbol{A}\right)\boldsymbol{W}\,\mathrm{d}\boldsymbol{x}\\ + \frac{1}{2}\int_\Gamma \left(\boldsymbol{V}^{(n,o)^T}\bar{\mathbb{A}}\boldsymbol{W}^{(n,e)} + \boldsymbol{V}^{(n,e)^T}\bar{\mathbb{A}}^T\boldsymbol{W}^{(n,o)}\right)\,\mathrm{d}\boldsymbol{l}\\ + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}.\end{aligned} \tag{III.163}$$

Inserting the boundary conditions from (III.154) in (III.163) yields

$$\begin{aligned}\frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\boldsymbol{W}\right)\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \left(\boldsymbol{\nabla}\boldsymbol{V}^T\cdot\boldsymbol{A}\right)\boldsymbol{W}\,\mathrm{d}\boldsymbol{x}\\ + \frac{1}{2}\int_\Gamma \boldsymbol{V}^{(n,o)^T}\left(\boldsymbol{L}^{-1}\boldsymbol{W}^{(n,o)} - \boldsymbol{L}^{-1}g\right)\,\mathrm{d}\boldsymbol{l}\\ + \frac{1}{2}\int_\Gamma \boldsymbol{V}^{(n,e)^T}\bar{\mathbb{A}}^T\left(\boldsymbol{L}\mathbb{A}\boldsymbol{W}^{(n,e)} + g\right)\,\mathrm{d}\boldsymbol{l}\\ + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}.\end{aligned} \tag{III.164}$$

For ease of implementation, it is advantageous to rewrite the variables $\boldsymbol{V}^{(n,o/e)}$ and $\boldsymbol{W}^{(n,o/e)}$ in terms of $\boldsymbol{V}$ and $\boldsymbol{W}$ by substituting $\boldsymbol{V}^{(n,o/e)} = \boldsymbol{P}^{(o/e)}\boldsymbol{T}_n\boldsymbol{V}$ and $\boldsymbol{W}^{(n,o/e)} = \boldsymbol{P}^{(o/e)}\boldsymbol{T}_n\boldsymbol{W}$

respectively. After substitution, Equation (III.164) becomes

$$\begin{aligned}\frac{1}{2}\int_\Omega \boldsymbol{V}^T\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\boldsymbol{W}\right)\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \left(\boldsymbol{\nabla}\boldsymbol{V}^T\cdot\boldsymbol{A}\right)\boldsymbol{W}\,\mathrm{d}\boldsymbol{x}\\ + \frac{1}{2}\int_\Gamma \boldsymbol{V}^T\boldsymbol{T}_{\boldsymbol{n}}^T{\boldsymbol{P}^{(o)}}^T\left(\boldsymbol{L}^{-1}\boldsymbol{P}^{(o)}\boldsymbol{T}_{\boldsymbol{n}}\boldsymbol{W} - \boldsymbol{L}^{-1}g\right)\,\mathrm{d}\boldsymbol{l}\\ + \frac{1}{2}\int_\Gamma \boldsymbol{V}^T\boldsymbol{T}_{\boldsymbol{n}}^T{\boldsymbol{P}^{(e)}}^T\bar{\mathbb{A}}^T\left(\boldsymbol{L}\mathbb{A}\boldsymbol{P}^{(e)}\boldsymbol{T}_{\boldsymbol{n}}\boldsymbol{W} + g\right)\,\mathrm{d}\boldsymbol{l}\\ + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x}.\end{aligned} \tag{III.165}$$

Rearranging terms in (III.165), the final weak form reads

$$\begin{aligned}\frac{1}{2}\int_\Omega \boldsymbol{V}^T\boldsymbol{A}\frac{\partial\boldsymbol{W}}{\partial x}\,\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int_\Omega \boldsymbol{V}^T\boldsymbol{A}\frac{\partial\boldsymbol{W}}{\partial y}\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \frac{\partial\boldsymbol{V}^T}{\partial x}\boldsymbol{A}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \frac{\partial\boldsymbol{V}^T}{\partial y}\boldsymbol{A}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x}\\ + \frac{1}{2}\int_\Gamma \boldsymbol{V}^T\boldsymbol{T}_{\boldsymbol{n}}^T{\boldsymbol{P}^{(o)}}^T\boldsymbol{L}^{-1}\boldsymbol{P}^{(o)}\boldsymbol{T}_{\boldsymbol{n}}\boldsymbol{W}\,\mathrm{d}\boldsymbol{l}\\ + \frac{1}{2}\int_\Gamma \boldsymbol{V}^T\boldsymbol{T}_{\boldsymbol{n}}^T{\boldsymbol{P}^{(e)}}^T\bar{\mathbb{A}}^T\boldsymbol{L}\mathbb{A}\boldsymbol{P}^{(e)}\boldsymbol{T}_{\boldsymbol{n}}\boldsymbol{W}\,\mathrm{d}\boldsymbol{l} + \int_\Omega \boldsymbol{V}^T\boldsymbol{P}\boldsymbol{W}\,\mathrm{d}\boldsymbol{x}\\ = \int_\Omega \boldsymbol{V}^T\boldsymbol{F}\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Gamma \boldsymbol{V}^T\left(\boldsymbol{T}_{\boldsymbol{n}}^T{\boldsymbol{P}^{(e)}}^T\mathbb{A}^T - \boldsymbol{T}_{\boldsymbol{n}}^T{\boldsymbol{P}^{(o)}}^T\boldsymbol{L}^{-1}\right)g\,\mathrm{d}\boldsymbol{l}.\end{aligned} \tag{III.166}$$

III.3.2. Implementation in FEniCS

FEniCS. In order to solve the variational formulation we derived in Equation (III.166), we use FEniCS which is an open-source computing platform to solve partial differential equations [33]. Thanks to its high-level user-friendly interface and one-to-one correspondence with mathematical formulations, it is relatively straightforward to implement the above variational formulation. Furthermore, owing to its ability to transform high-level mathematical expression into high-performance finite element code, it is possible to create highly efficient code.

F2ME. Fenics For Moment Equations (F2ME) is a Python-based moment system solver built on the FEniCS platform. The architecture of F2ME is illustrated in Figure III.2. Users can input the relevant system matrices for the corresponding moment system at the beginning of the program. The computational domain is generated using mesh generation software such as Gmsh. The weak formulation in (III.166) is an integral part of F2ME and remains consistent for any arbitrary first-order moment system. The source code for the F2ME program can be found in [34]. A detailed description of the implementation of the 13 moment system within the F2ME framework is discussed in Appendix B. This approach can be adapted for any moment system with minimal changes to the source code.

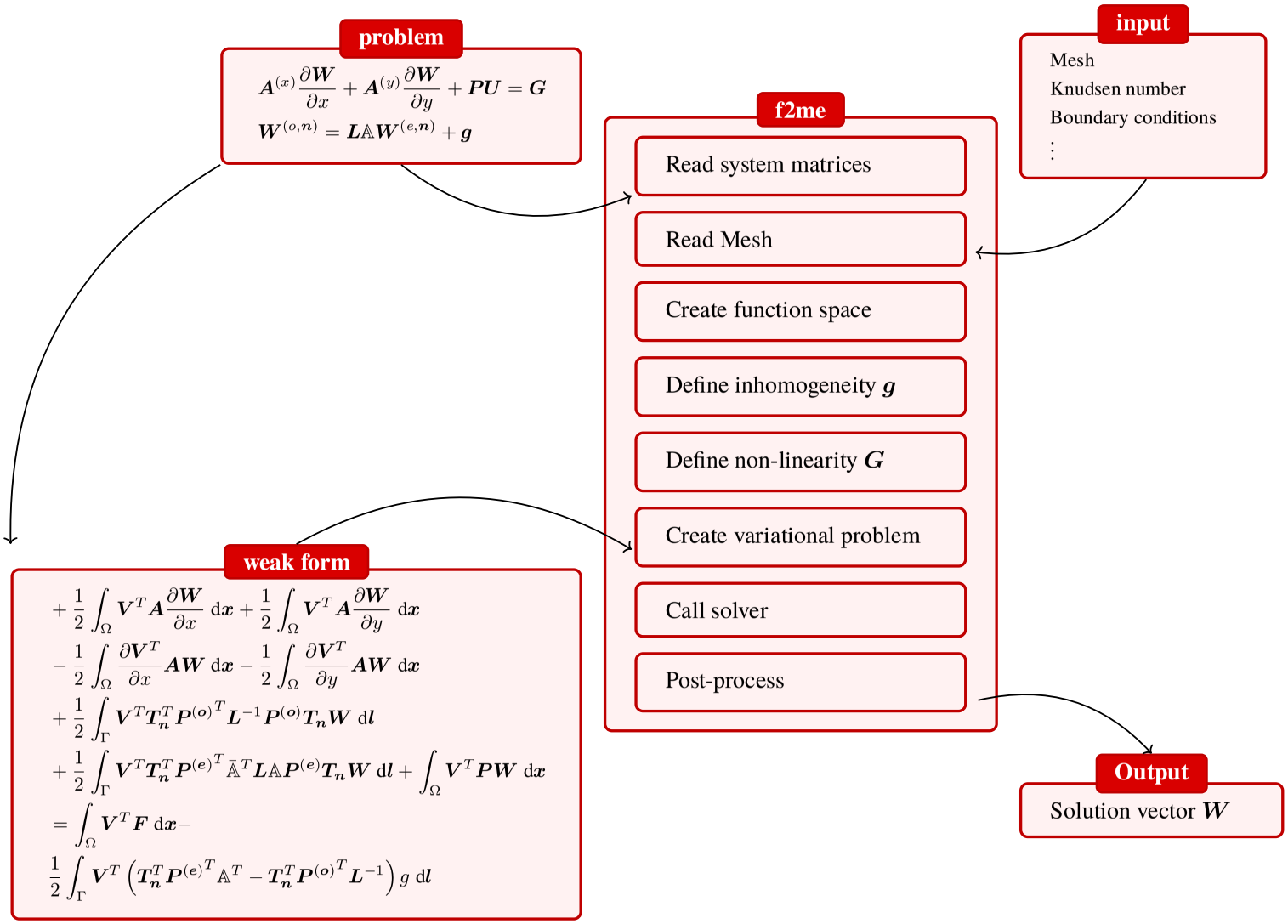


Figure III.2.: F2ME code architecture

III.4. Numerical simulations

III.4.1. Case study: Flow over a cylinder

Case description. [32] We consider a gas flowing around a cylinder of radius R_0. To study the dynamics of flow around the cylinder, it is sufficient to focus on the region around the cylinder. This can be achieved by restricting domain of interest as a region between the inner cylinder and an outer cylinder of radius R_1 which includes the region from the radius R_0 till arbitrary radius R_1.

Geometry. Following the problem description, we consider the geometry as shown in the diagram III.3 with inner circle radius $R_0 = 0.5$ and outer circle radius $R_1 = 2.0$. Γ_0 and Γ_1 denote the inner and outer boundary of the domain respectively. $\hat{n}_0$ and $\hat{n}_0$ are two normal vectors, pointing inwardly, perpendicular to the domain from the respective boundaries. The following quantities can be prescribed at each boundary: temperature θ, tangential velocity $u^{(\mathrm{t})}$, normal velocity $u^{(\mathrm{n})}$, and pressure p. Thanks to the geometry of the problem, there exists an analytical solution to this setup[35].

Simulation setup. In this test case, we perform a convergence study to verify the convergence of the proposed numerical framework. Specifically, we choose R13 moment system as our mathematical model and use the parameters below in the simulation. The Knudsen number is set to 0.1 on the entire domain. The prescribed boundary values are as follows: at the inner boundary $u^{(\mathrm{t})} = -10\sin(\phi)$, $u^{(\mathrm{n})} = 0.0$, $p = 0.0$, $\theta = 1$ and at the outer boundary $u^{(\mathrm{t})} = -1.0\sin(\phi)$, $u^{(\mathrm{n})} = +1.0\cos(\phi)$, $p = -0.27\cos(\phi)$, $\theta = 0.5$. For the convergence study, we run simulations on several meshes of varying mesh sizes and present the results below.

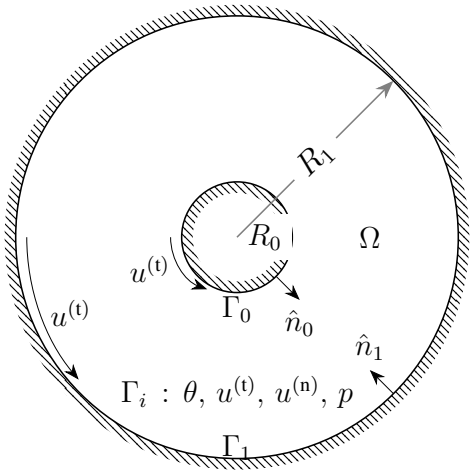


Figure III.3.: Flow over a cylinder - geometry

Results. A comparison of normalised errors for all the variables with respect to different mesh sizes are presented in Figure III.4. The normalised l_2 error norm is calculated as $l_2 = \frac{\|w_{\text{exact}} - w_{\text{approx}}\|_2}{\|w_{\text{exact}}\|_2}$. Similarly, normalised l_∞ error norm is calculated as $l_\infty = \frac{\|w_{\text{exact}} - w_{\text{approx}}\|_\infty}{\|w_{\text{exact}}\|_\infty}$. We note that the convergence orders for all the quantities of interest are between first and second order. This is due to use of P1 Lagrangian element in the simulation. Usage of higher order elements and better discretization of domain may lead to improved convergence rate. We remark that the successful convergence study confirms the convergence of the proposed numerical framework.

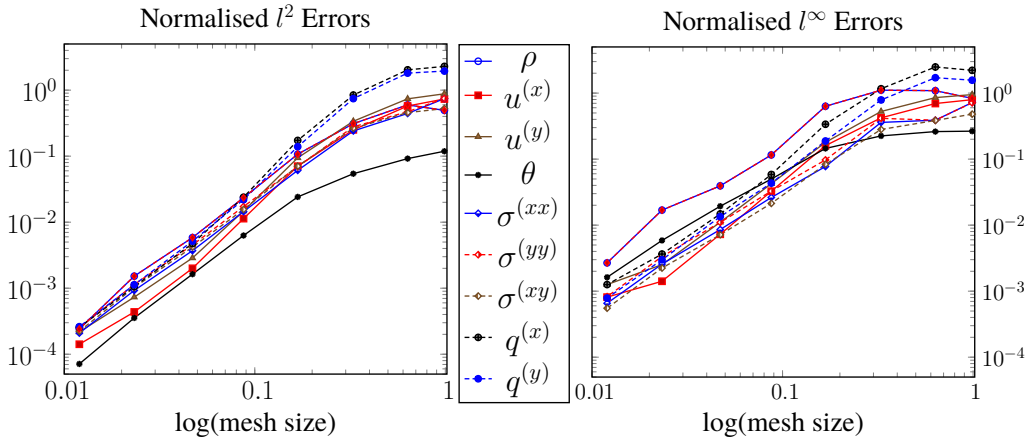


Figure III.4.: Convergence study of R13 moment equations with analytical solution

III.4.2. Case study: Lid-driven cavity flow

Case description. [32] In this test case, we consider a square cavity which has a side length of L unit, is filled with gas, and is covered on top by a lid. We study the dynamics of the gas inside the cavity as a result of sliding the lid.

Geometry. Following the problem description, we consider the geometry as in Figure III.5 with $L = 1$. It has four boundaries, three of which are stationary wall, namely Γ_1, Γ_2, Γ_3 and one of which is a moving lid Γ_0. Similar to the previous test case, the following quantities are prescribed at the boundaries: pressure p, tangential velocity $u^{(\text{t})}$, normal velocity $u^{(\text{n})}$, and temperature θ.

Simulation setup. In this test case, we perform a validation study by comparing solutions obtained by moment method with solutions obtained by standard computational fluid dynamics theory. The underlying motivation for this validation is to demonstrate that moment method theory at the continuum limit converges to classical fluid dynamics theory. For this study, we choose G13 moment system as our mathematical model and use the parameters below in the simulation. The following values are prescribed at the boundaries: at the stationary walls $u^{(\text{t})} = 0$, $u^{(\text{n})} = 0$, $p = 0$, $\theta = 0$ and at the moving lid $u^{(\text{t})} = 1$, $u^{(\text{n})} = 0$, $p = 0$, $\theta = 0$. The Knudsen number is set to very small values close to 1e^{-2} and 1e^{-3} across the entire computational domain to simulate flows at Reynolds numbers of 100 and 1000, respectively.

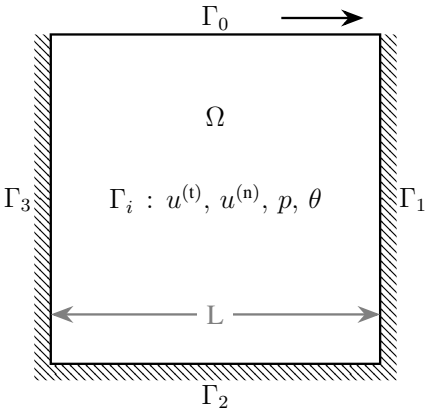
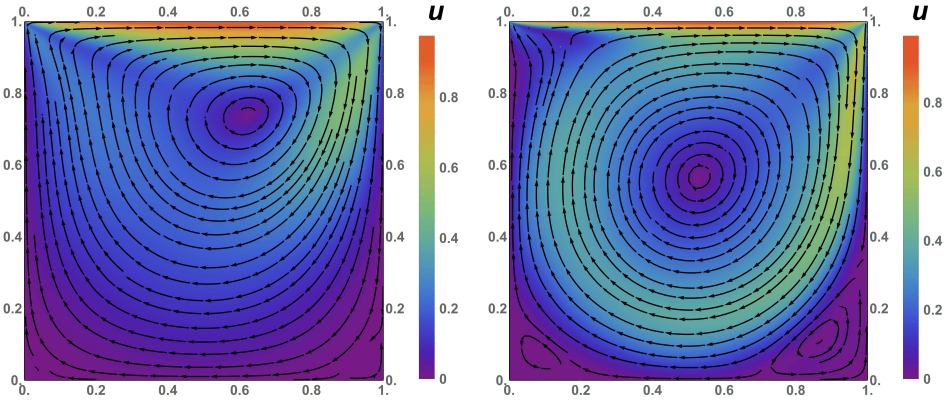
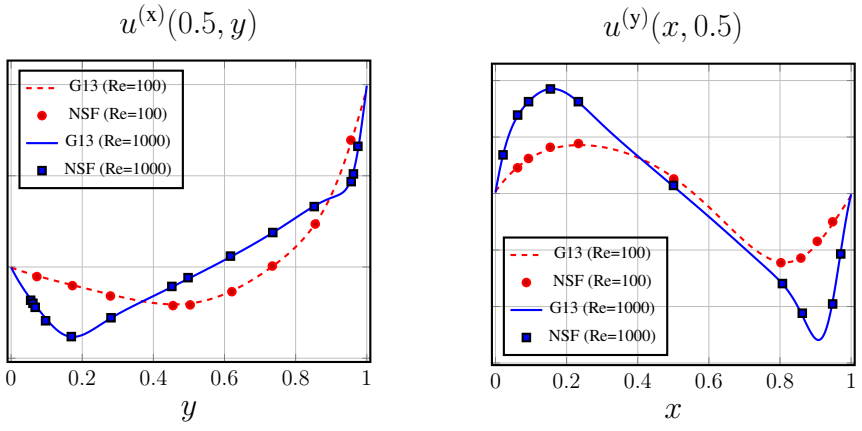


Figure III.5.: Geometry of the lid-driven cavity flow problem



(a) G13 velocity contour profiles and stream lines at Reynold's number 100

(b) G13 velocity contour profiles and stream lines at Reynold's number 1000



(c) Comparison of x component of the velocity along y-axis at $x = 0.5$.

(d) Comparison of y component of the velocity along x-axis at $y = 0.5$.

Figure III.6.: Comparison of velocity profiles predicted by G13 moment theory with classical fluid dynamics equations (NSF)

Results. In this test case, we compared the solutions computed using the G13 moment system with those computed using standard fluid dynamics equations as described in [36]. Figure III.6 shows the comparisons of the x component of velocity along the line $(x = 0.5, y)$ and the y

component of velocity along the line $(x, y = 0.5)$ for two different Reynolds numbers, namely $\text{Re} = 100$ and $\text{Re} = 1000$. We observe good agreement between the solutions predicted by the respective theories at the near continuum limit. This study validates our claim that the proposed general moment system framework at the continuum limit is consistent with classical fluid dynamics theory.

III.4.3. Case study: Non-isothermal lid-driven micro-cavity flow

Case Description. For this test case, we consider a geometry similar to that shown in Figure III.5. The main difference is that we set the length L to 100μm for this case. We use the properties of Argon gas at 293 K as reference values. Furthermore, the cavity's lid moves with a velocity of $10\,\text{m/s}$. The temperature of the lid is kept higher than the gas and wall temperature. We investigate the dynamics of gas flow driven by temperature difference and velocity inside a micro-cavity. We examine three setups that differ only in their temperature differences: $\Delta T = 10\,\text{K}$, $\Delta T = 70\,\text{K}$, and $\Delta T = 170\,\text{K}$.

Simulation Setup. As in the previous case study, we use the G13 moment system as our mathematical model, with the following values applied to respective simulations.

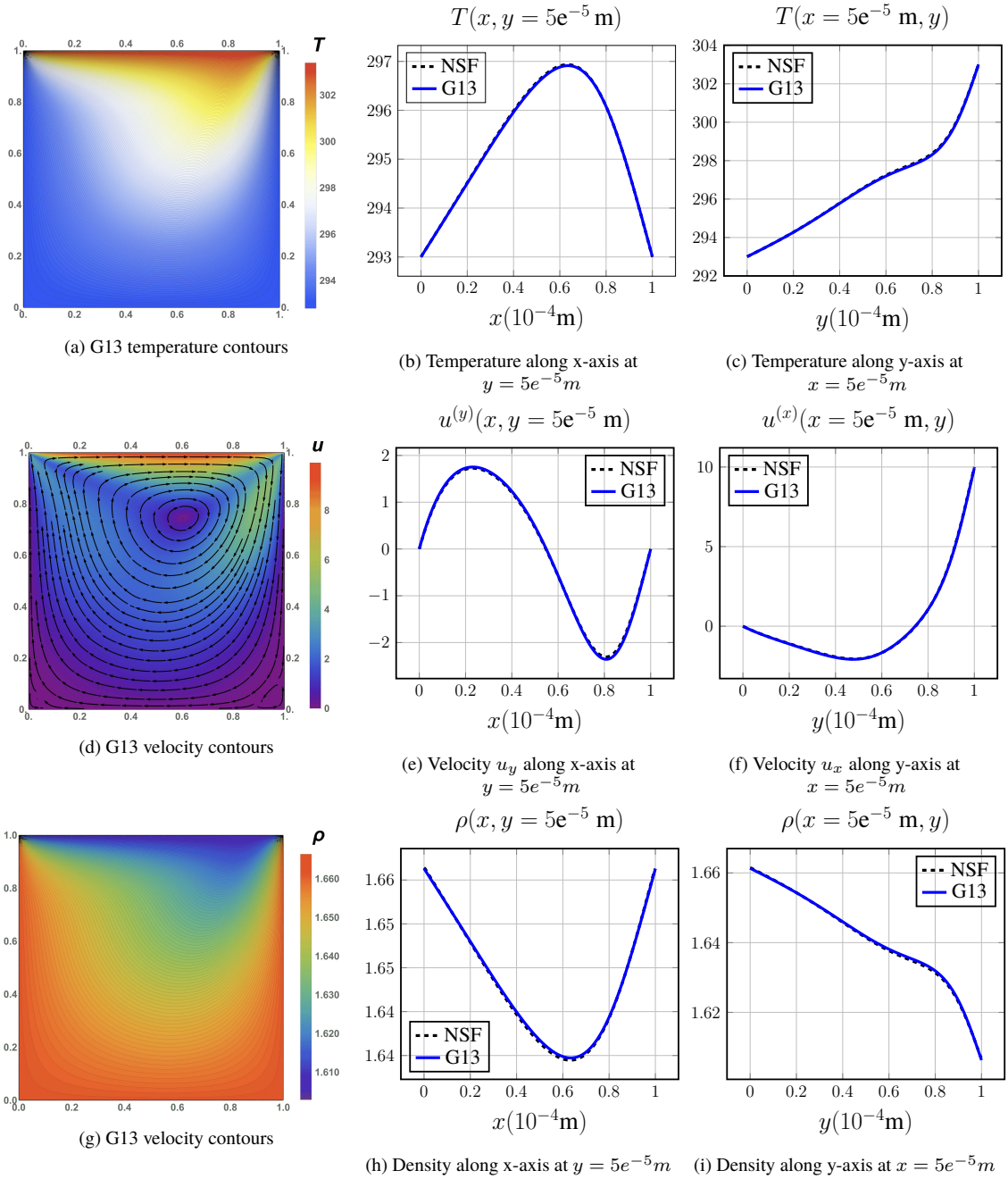
- $\Delta T = 10\,\text{K}$ case: The boundary conditions are as follows: at the stationary walls, $u^{(\text{t})} = 0$, $u^{(\text{n})} = 0$, $p = 0$, and $T = 293\,\text{K}$; at the moving lid, $u^{(\text{t})} = 10\,\text{m/s}$, $u^{(\text{n})} = 0$, $p = 0$, and $T = 303\,\text{K}$. The Knudsen number is set to a very small value of 5e^{-4} across the entire computational domain. The reference density and reference temperature are set to $1.646\,\text{Kg/m}^3$ and $298\,\text{K}$ respectively.

- $\Delta T = 70\,\text{K}$ case: Compared to previous case, the only change is the temperature at the moving lid, which is set to $T = 363\,\text{K}$. The rest of the boundary conditions remain the same as in the previous setup. The Knudsen number is set to 5.5e^{-4} across the entire computational domain. The reference density and reference temperature are set to $1.5625\,\text{Kg/m}^3$ and $320\,\text{K}$ respectively.

- $\Delta T = 170\,\text{K}$ case: In this case, the temperature at the moving lid is set to $T = 463\,\text{K}$, with all other boundary conditions unchanged from the previous setup. The Knudsen number is set to 5.5e^{-4} across the entire computational domain. The reference density and reference temperature are set to $1.46\,\text{Kg/m}^3$ and $360\,\text{K}$ respectively.

The motivation for this case study is to demonstrate the ability of the proposed framework to capture non-linearities in the process. We compare the numerical solutions predicted by G13 moment system with the solutions predicted by classical fluid dynamical equations (Navier-Stokes-Fourier). Furthermore, we use the proprietary software ANSYS Fluent, a well-known fluid dynamics software suite, to solve the NSF equations.

Discussion of results. The comparison of numerical solutions — temperature, velocity, and density — predicted by G13 and NSF is provided in Figures III.7, III.8, and III.9 for temperature differences $\Delta T = 10K$, $\Delta T = 70K$, and $\Delta T = 170K$ respectively. Each figure includes



(a) G13 temperature contours

(b) Temperature along x-axis at $y = 5e^{-5}m$

(c) Temperature along y-axis at $x = 5e^{-5}m$

(d) G13 velocity contours

(e) Velocity u_y along x-axis at $y = 5e^{-5}m$

(f) Velocity u_x along y-axis at $x = 5e^{-5}m$

(g) G13 velocity contours

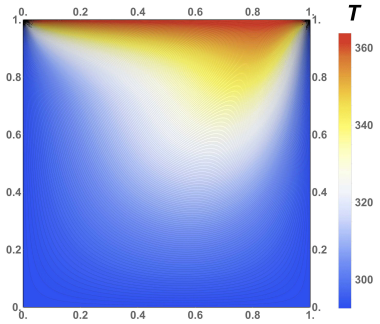
(h) Density along x-axis at $y = 5e^{-5}m$

(i) Density along y-axis at $x = 5e^{-5}m$

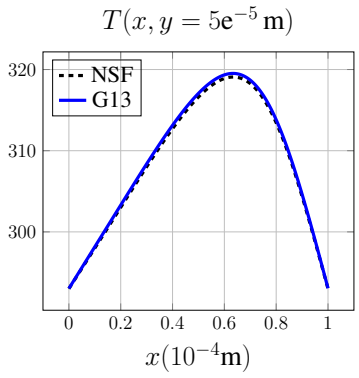
Figure III.7.: Comparison of macroscopic variables between G13 moment system and NSF for $\Delta T = 10K$

several sub-figures. For example, sub-figures III.7b, III.7e, and III.7h illustrate the temperature T, the y-component of velocity $u^{(y)}$, and the density ρ plotted along the line parallel to the x-axis $y = 5 \times 10^{-5}$ m. Sub-figures III.7c, III.7f, and III.7i illustrate the temperature T, the x-component of velocity $u^{(x)}$, and the density ρ plotted along the line parallel to the y-axis at $x = 5 \times 10^{-5}$ m.

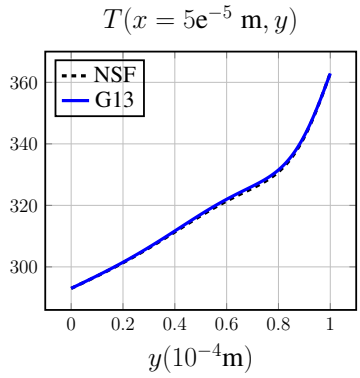
From Figures III.7 and III.8, we observe that when the temperature difference is small, there is very good agreement between the solutions produced by the moment theory and those



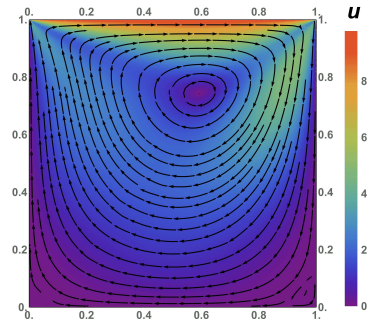
(a) G13 temperature contours



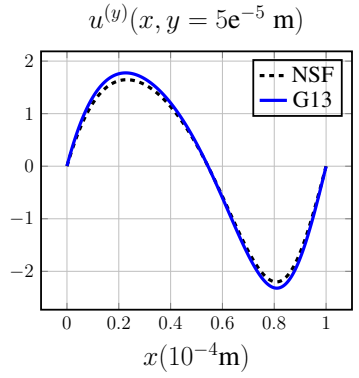
(b) Temperature along x-axis at $y = 5e^{-5}m$



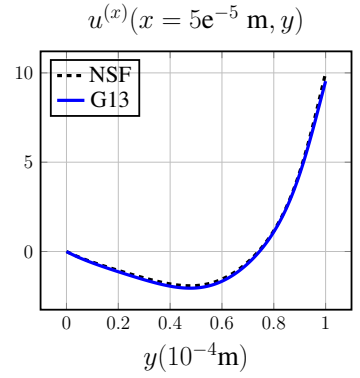
(c) Temperature along y-axis at $x = 5e^{-5}m$



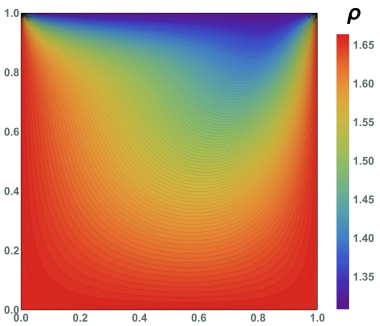
(d) G13 velocity contours



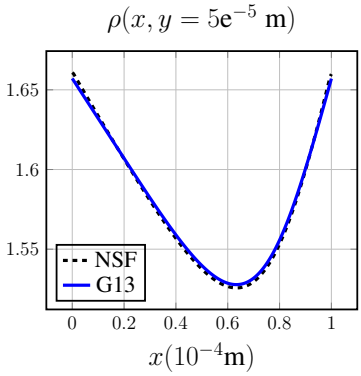
(e) Velocity u_y along x-axis at $y = 5e^{-5}m$



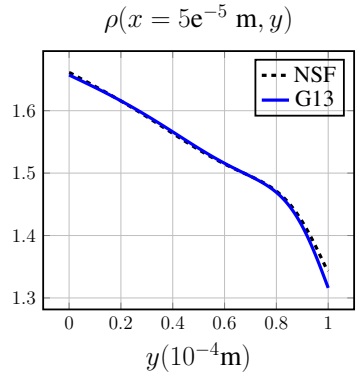
(f) Velocity u_x along y-axis at $x = 5e^{-5}m$



(g) G13 density contours



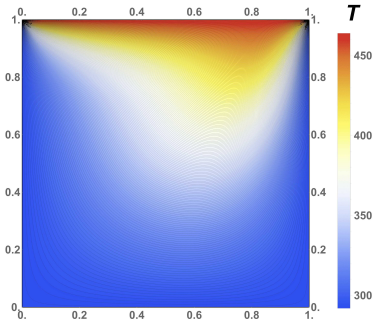
(h) Density along x-axis at $y = 5e^{-5}m$



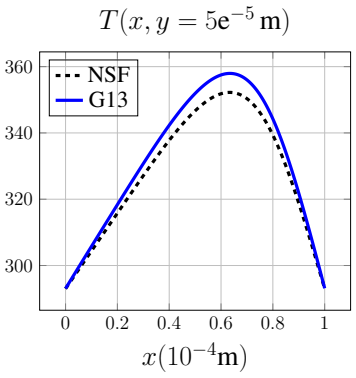
(i) Density along y-axis at $x = 5e^{-5}m$

Figure III.8.: Comparison of macroscopic variables between G13 moment system and NSF for $\Delta T = 70K$

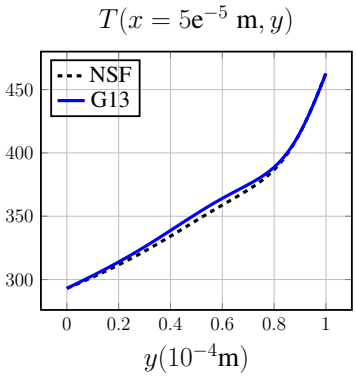
from the classical fluid dynamics theory. However, as the temperature difference increases to $\Delta T = 170\,\mathrm{K}$, the G13 solutions begin to deviate from the solutions obtained using the Navier-Stokes-Fourier equations, as shown in Figure III.9. This discrepancy can be attributed to various factors, including the choice of reference background values, the weak imposition of boundary conditions in FEniCS implementation, and the presence of strong non-linearity. Further investigation is needed to address this issue comprehensively.



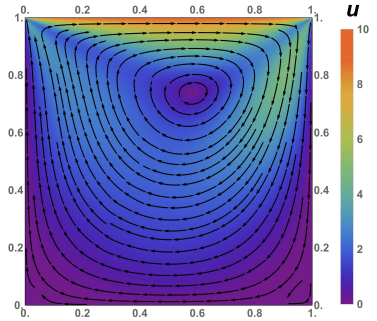
(a) G13 temperature contours



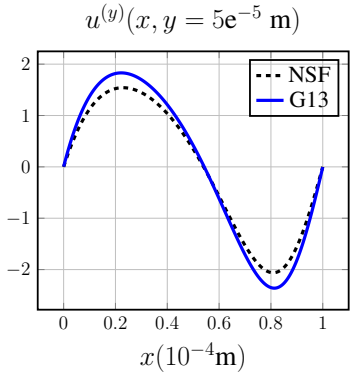
(b) Temperature along x-axis at $y = 5e^{-5}m$



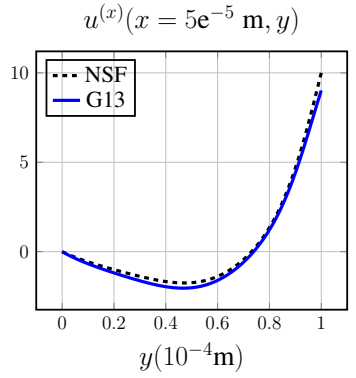
(c) Temperature along y-axis at $x = 5e^{-5}m$



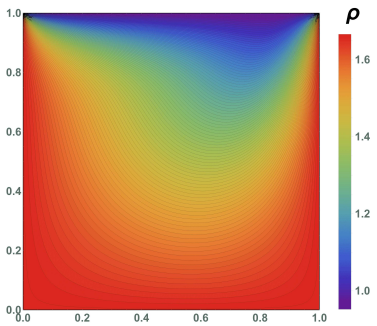
(d) G13 velocity contours



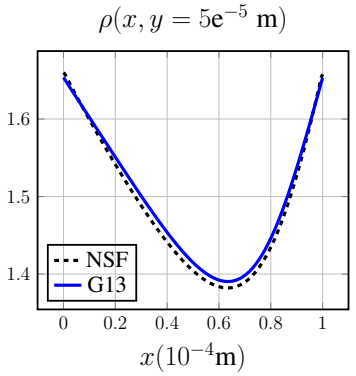
(e) Velocity u_y along x-axis at $y = 5e^{-5}m$



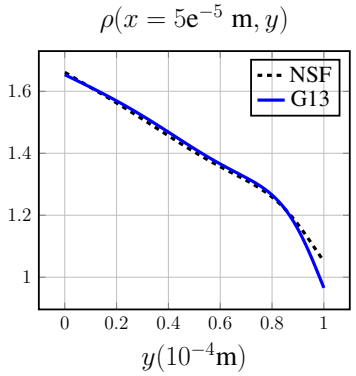
(f) Velocity u_x along y-axis at $x = 5e^{-5}m$



(g) G13 density contours



(h) Density along x-axis at $y = 5e^{-5}m$



(i) Density along y-axis at $x = 5e^{-5}m$

Figure III.9.: Comparison of macroscopic variables between G13 moment system and NSF for $\Delta T = 170K$

III.5. Summary

In this chapter, we explored the advantages of using first-order formulations for higher-order partial differential equations. We derived a general moment system from the Boltzmann equation for mono-atomic gases and examined the properties of these generic systems. The resulting structure is straightforward: it consists of linear operators for the transport part of the equations and terms that are at most quadratic in the evaluation of the Boltzmann collision operator.

We introduced a generic numerical framework based on the finite element method to solve moment systems and provided a detailed implementation of the G13 moment system as a first-order formulation. Our numerical framework was demonstrated through three case studies: the first validated the proposed numerical framework through a convergence study, the second demonstrated the consistency of the moment theory with classical fluid dynamics in the continuum limit, and the third illustrated the capability of our moment formulation to capture weak non-linearities in the process. We observed that as the non-linearity became stronger, the numerical solution predicted by the G13 moment system deviated from the ANSYS solution. This deviations can be attributed to various factors: choice of reference values, presence of strong non-linearity, weak imposition of boundary conditions in FEniCS implementation. Further investigation is needed to address this issue.

Chapter IV Outlook

Summary of findings. This thesis presents two significant contributions to the development of moment methods in kinetic gas theory.

It demonstrates that moment equations can be effectively applied to three-dimensional non-equilibrium gas flows. The successful application of the regularized 13-moment (R13) equations to the Crookes radiometer problem enhanced our understanding of the force driving the radiometer, validated widely accepted hypothesis regarding its working principle, and enabled us to study the design of vanes and their effects on the radiometric force.

Additionally, the development of a generic moment system framework provided a comprehensive procedure to derive arbitrary order of moment systems. The mathematical structure of these equations facilitates the development of a generic finite-element-method based numerical framework. Together, these advancements enhance our ability to model weakly non-linear gas flow processes with a balanced approach to accuracy and computational efficiency.

Significance and Impact. The findings of this research contribute to the broader understanding of non-equilibrium gas dynamics, offering a valuable tool for simulations that bridge the gap between classical fluid dynamics and kinetic theory. The application of moment equations to real-world problems, such as the Crookes radiometer, illustrates their practical utility and potential for further exploration.

Challenges. In the derivation of the generic moment system, our choice of reference distribution to approximate the underlying distribution is globally constant. This enabled us to derive a set of generic linear moment equations, as discussed in Chapter III. When modelling slow gas flow processes, such as micro-channel flows, low local Mach numbers can help us choose appropriate reference values. However, when modelling general gas flow processes, great care must be taken in choosing the reference values to ensure they do not deviate significantly from the true underlying distribution. This is often not trivial in real-world problems and requires further exploration.

Future research directions. The derivation of generic moment systems of arbitrary order primarily focused on mono-atomic gases. The successful application of these systems motivates us to explore a similar generic derivation for polyatomic gases, which has been addressed through the study of the 14- and 17-moment equations for polyatomic gases in [37].

Furthermore, during the development of the generic numerical framework, our focus was initially limited to steady-state equations. The next logical step is to extend this framework to incorporate time dependence, thereby enhancing its applicability to a broader range of gas flow scenarios. Additionally, we have developed moment models of arbitrary order, including 13, and 26 moments, among others. It would be beneficial to implement a model adaptation strategy, applying different mathematical models to various parts of the computational domain based on the required accuracy. This approach will enhance the flexibility and efficiency of our modelling efforts.

Extension to time-dependent moment models. The current numerical framework mainly handles steady-state problems, which works well for many scenarios but limits the range of applications. The generalised finite-element framework that has been developed can be extended to simulate time-dependent problems. From the time term in (III.63), one can see that it leads to a diagonal matrix if written as a first-order system. Therefore, (III.154) can be augmented with a time-dependent iteration. This extension allows for the modelling of unsteady flows, such as transient heating effects, which are common in real-world gas dynamics. Time-dependent simulations have also been studied for specific moment systems, such as the regularised 13-moment equations with Onsager boundary conditions in the linear regime [38]. The main challenge lies in maintaining numerical stability and accuracy when the time dimension is introduced, especially for high-order systems. Still, this is a necessary step to make the framework applicable to a broader class of problems.

Extension to higher-order moment models. Moment models of arbitrary order have been developed, including commonly used sets like 13 and 26 moments. These higher-order models are useful when lower-order ones fail to capture important physical effects, particularly in rarefied or non-equilibrium regimes. In the current setup, the generic finite-element framework requires the system matrices and boundary condition expressions as input. These are pre-computed using an automated algebraic tool, such as Mathematica, and then provided to the Python-based finite-element solver F2ME. To handle higher-order models, it is first necessary to compute the corresponding coefficients in the algebraic software. Moving forward, this process should be automated, so that F2ME can independently generate and solve arbitrary moment models without requiring manual symbolic derivation. However, the effect of boundary conditions in higher-order moment models is not yet fully understood and should be studied in more detail, as it can significantly influence the solution. With these improvements, one can pursue a more systematic exploration of the trade-off between model complexity and accuracy and choose the right level of detail for specific applications.

Model Adaptation Strategy. Not all parts of a flow domain require the same level of modeling detail. For instance, near walls or in shock regions, high-order models may be needed, while in more uniform areas, lower-order models might suffice. To take advantage of this, one can implement a model adaptation strategy that selects the appropriate moment model based on local flow features or error estimates. This helps reduce computational cost without sacrificing accuracy where it's most critical. The idea is to build a solver that adapts dynamically

during the simulation, improving both efficiency and flexibility.

Appendix

A R13 Moment System

A.1. Weak formulation

A.1.1. Weak formulation of the heat flux balance equation

Differential equation. The heat flux balance equation reads

$$\frac{5}{2}\nabla\theta + \nabla\cdot\boldsymbol{\sigma} + \frac{1}{2}\nabla\cdot\boldsymbol{R} + \frac{1}{6}\nabla\Delta = -\frac{1}{\mathrm{Kn}}\frac{2}{3}\boldsymbol{s}. \tag{A.1}$$

Weak integral formulation. We multiply the test function $\boldsymbol{r}$ with the heat flux balance equation and integrate the resulting expression over the computational domain. We get

$$\begin{aligned}\frac{5}{2}\int_\Omega \nabla\theta\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} + \int_\Omega(\nabla\cdot\boldsymbol{\sigma})\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int_\Omega(\nabla\cdot\boldsymbol{R})\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} \\ + \frac{1}{6}\int_\Omega\nabla\Delta\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} = -\frac{1}{\mathrm{Kn}}\frac{2}{3}\int_\Omega\boldsymbol{s}\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x}.\end{aligned} \tag{A.2}$$

Integration by parts. After applying integration-by-parts technique and divergence theorem to each term in the left hand side of the Equation (A.2) we obtain

$$\frac{5}{2}\int_\Omega(\nabla\theta)\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} = -\frac{5}{2}\int_\Omega\theta(\nabla\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} + \frac{5}{2}\int_\Gamma\theta(\boldsymbol{r}\cdot\boldsymbol{n})\,\mathrm{d}\boldsymbol{l}, \tag{A.3}$$

$$\int_\Omega(\nabla\cdot\sigma)\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} = -\int_\Omega\boldsymbol{\sigma}:\nabla\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} + \int_\Gamma(\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{l}, \tag{A.4}$$

$$\frac{1}{2}\int_\Omega(\nabla\cdot\boldsymbol{R})\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} = -\frac{1}{2}\int_\Omega\boldsymbol{R}:\nabla\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int_\Gamma(\boldsymbol{R}\cdot\boldsymbol{n})\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{l}, \tag{A.5}$$

$$\frac{1}{6}\int_\Omega(\nabla\Delta)\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} = -\frac{1}{6}\int_\Omega\Delta(\nabla\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} + \frac{1}{6}\int_\Gamma\Delta(\boldsymbol{r}\cdot\boldsymbol{n})\,\mathrm{d}\boldsymbol{l}. \tag{A.6}$$

Boundary condition substitution. The expressions for the boundary integral terms in Equations (A.3) - (A.6) are provided below:

$$\theta(\boldsymbol{r}\cdot\boldsymbol{n}) = \theta\left(\begin{bmatrix}r_n\\ r_{t_1}\\ r_{t_2}\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}\right) = \theta r_n, \tag{A.7}$$

$$(\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{r} = \left(\begin{bmatrix}\sigma_{nn} & \sigma_{nt_1} & \sigma_{nt_2}\\ \sigma_{t_1n} & \sigma_{t_1t_1} & \sigma_{t_1t_2}\\ \sigma_{t_2n} & \sigma_{t_2t_1} & -(\sigma_{nn}+\sigma_{t_1t_1})\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}\right)\begin{bmatrix}r_n\\r_{t_1}\\r_{t_2}\end{bmatrix} = \sigma_{nn}r_n + \sigma_{nt_1}r_{t_1} + \sigma_{nt_2}r_{t_2}, \tag{A.8}$$

$$(\boldsymbol{R}\cdot\boldsymbol{n})\cdot\boldsymbol{r} = \left(\begin{bmatrix}R_{nn} & R_{nt_1} & R_{nt_2}\\ R_{t_1n} & R_{t_1t_1} & R_{t_1t_2}\\ R_{t_2n} & R_{t_2t_1} & -(R_{nn}+R_{t_1t_1})\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}\right)\begin{bmatrix}r_n\\r_{t_1}\\r_{t_2}\end{bmatrix} = R_{nn}r_n + R_{nt_1}r_{t_1} + R_{nt_2}r_{t_2}, \tag{A.9}$$

$$\Delta(\boldsymbol{r}\cdot\boldsymbol{n}) = \Delta\left(\begin{bmatrix}r_n\\r_{t_1}\\r_{t_2}\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}\right) = \Delta r_n. \tag{A.10}$$

First we substitute the expressions (A.7) - (A.10) in the boundary integral terms in Equations (A.3) - (A.6) and subsequently substitute the resulting equations in Equation (A.2). After re-ordering the terms in the boundary integrals we obtain

$$\begin{aligned}-\frac{5}{2}\int_\Omega \theta(\nabla\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} - \int_\Omega \boldsymbol{\sigma}:\boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \boldsymbol{R}:\boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} - \frac{1}{6}\int_\Omega \Delta(\boldsymbol{\nabla}\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x}\\ + \int_\Gamma\left(\frac{5}{2}\theta + \sigma_{nn} + \frac{1}{2}R_{nn} + \frac{1}{6}\Delta\right)r_n\,\mathrm{d}\boldsymbol{l} + \int_\Gamma\left(\sigma_{nt_1} + \frac{1}{2}R_{nt_1}\right)r_{t_1}\,\mathrm{d}\boldsymbol{l}\\ + \int_\Gamma\left(\sigma_{nt_2} + \frac{1}{2}R_{nt_2}\right)r_{t_2}\,\mathrm{d}\boldsymbol{l} = -\frac{1}{\mathrm{Kn}}\frac{2}{3}\int_\Omega \boldsymbol{s}\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x}.\end{aligned} \tag{A.11}$$

For the sake of preserving the mathematical structure, we normalise the Equation (A.11) by a factor of 2/5 and we get

$$\begin{aligned}-\int_\Omega \theta(\nabla\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} - \frac{2}{5}\int_\Omega \boldsymbol{\sigma}:\boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} - \frac{1}{5}\int_\Omega \boldsymbol{R}:\boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} - \frac{1}{15}\int_\Omega \Delta(\boldsymbol{\nabla}\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x}\\ + \int_\Gamma \underbrace{\left(\theta + \frac{2}{5}\sigma_{nn} + \frac{1}{5}R_{nn} + \frac{1}{15}\Delta\right)}_{\text{Expression 1}} r_n\,\mathrm{d}\boldsymbol{l} + \int_\Gamma \underbrace{\left(\frac{2}{5}\sigma_{nt_1} + \frac{1}{5}R_{nt_1}\right)}_{\text{Expression 2}} r_{t_1}\,\mathrm{d}\boldsymbol{l}\\ + \int_\Gamma \underbrace{\left(\frac{2}{5}\sigma_{nt_2} + \frac{1}{5}R_{nt_2}\right)}_{\text{Expression 3}} r_{t_2}\,\mathrm{d}\boldsymbol{l} = -\frac{1}{\mathrm{Kn}}\frac{4}{15}\int_\Omega \boldsymbol{s}\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x}.\end{aligned} \tag{A.12}$$

The three terms namely Expressions 1, 2, and 3 in Equation (A.12) can be further simplified by appropriate substitution of boundary conditions. Simplification of the Expression 1 involves the reformulation of the boundary condition in Equation (II.12)

$$\theta + \frac{1}{4}\sigma_{nn} + \frac{1}{5}R_{nn} + \frac{1}{15}\Delta = \frac{1}{2}\frac{1}{\tilde{\chi}}s_n + \theta^w. \tag{A.13}$$

Comparing the Equation (A.13) and the Expression 1 in the Equation (A.12) we can conclude that Expression 1 is equivalent to

$$\frac{1}{2}\frac{1}{\tilde{\chi}}s_n + \theta^w + \frac{3}{20}\sigma_{nn}. \tag{A.14}$$

Simplification of the Expressions 2 and 3 involves the addition of the two boundary conditions in Equations (II.10) and (II.11)

$$\sigma_{nt_i} + R_{nt_i} = \frac{12}{5}\tilde{\chi}s_{t_i}, \qquad i = 1, 2. \tag{A.15}$$

Comparing the Equation (A.15) and the Expressions 2 and 3 in Equation (A.12) we can conclude that the Expressions 2 and 3 are equivalent to

$$\frac{12}{25}\tilde{\chi}s_{t_1} + \frac{1}{5}\sigma_{nt_1} \quad \text{and} \quad \frac{12}{25}\tilde{\chi}s_{t_2} + \frac{1}{5}\sigma_{nt_2} \quad \text{respectively.} \tag{A.16}$$

After substituting the Expressions (A.14) and (A.16) in Equation (A.12) we obtain

$$\begin{aligned} -\int_\Omega \theta(\nabla\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} - \frac{2}{5}\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} - \frac{1}{5}\int_\Omega \boldsymbol{R} : \boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} - \frac{1}{15}\int_\Omega \Delta(\boldsymbol{\nabla}\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} \\ + \int_\Gamma \left(\frac{1}{2}\frac{1}{\tilde{\chi}}s_n + \theta^w + \frac{3}{20}\sigma_{nn}\right) r_n\,\mathrm{d}\boldsymbol{l} + \int_\Gamma \left(\frac{12}{25}\tilde{\chi}s_{t_1} + \frac{1}{5}\sigma_{nt_1}\right) r_{t_1}\,\mathrm{d}\boldsymbol{l} \\ + \int_\Gamma \left(\frac{2}{5}\tilde{\chi}s_{t_2} + \frac{1}{5}\sigma_{nt_2}\right) r_{t_2}\,\mathrm{d}\boldsymbol{l} = -\frac{1}{\mathrm{Kn}}\frac{4}{15}\int_\Omega \boldsymbol{s}\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x}. \end{aligned} \tag{A.17}$$

Closure relation substitution. Substituting the closure relations for $\boldsymbol{R}$ and Δ yields

$$\begin{aligned} -\int_\Omega \theta(\nabla\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} - \frac{2}{5}\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} + \frac{24}{25}\mathrm{Kn}\int_\Omega (\boldsymbol{\nabla s})_{\mathrm{stf}} : \boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} + \frac{12}{15}\mathrm{Kn}\int_\Omega (\boldsymbol{\nabla}\cdot\boldsymbol{s})(\boldsymbol{\nabla}\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} \\ + \int_\Gamma \left(\frac{1}{2}\frac{1}{\tilde{\chi}}s_n + \theta^w + \frac{3}{20}\sigma_{nn}\right) r_n\,\mathrm{d}\boldsymbol{l} + \int_\Gamma \left(\frac{12}{25}\tilde{\chi}s_{t_1} + \frac{1}{5}\sigma_{nt_1}\right) r_{t_1}\,\mathrm{d}\boldsymbol{l} \\ + \int_\Gamma \left(\frac{2}{5}\tilde{\chi}s_{t_2} + \frac{1}{5}\sigma_{nt_2}\right) r_{t_2}\,\mathrm{d}\boldsymbol{l} = -\frac{1}{\mathrm{Kn}}\frac{4}{15}\int_\Omega \boldsymbol{s}\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x}. \end{aligned} \tag{A.18}$$

Weak formulation in functional notation. After re-ordering the terms in Equation (A.18), the final weak formulation for the heat flux balance equation in functional notation reads

$$a(\boldsymbol{s}, \boldsymbol{r}) - b(\theta, \boldsymbol{r}) - c(\boldsymbol{\sigma}, \boldsymbol{r}) = l_1(\boldsymbol{r}), \tag{A.19}$$

where

$$\begin{aligned} a(\boldsymbol{s}, \boldsymbol{r}) := +\frac{1}{\mathrm{Kn}}\frac{4}{15}\int_\Omega \boldsymbol{s}\cdot\boldsymbol{r}\,\mathrm{d}\boldsymbol{x} + \frac{24}{25}\mathrm{Kn}\int_\Omega (\boldsymbol{\nabla s})_{\mathrm{stf}} : \boldsymbol{\nabla r}\,\mathrm{d}\boldsymbol{x} \\ + \frac{12}{15}\mathrm{Kn}\int_\Omega (\boldsymbol{\nabla}\cdot\boldsymbol{s})(\boldsymbol{\nabla}\cdot\boldsymbol{r})\,\mathrm{d}\boldsymbol{x} + \int_\Gamma \left(\frac{1}{2}\frac{1}{\tilde{\chi}}s_n r_n + \frac{12}{25}\tilde{\chi}s_{t_1}r_{t_1} + \frac{2}{5}\tilde{\chi}s_{t_2}r_{t_2}\right)\,\mathrm{d}\boldsymbol{l}, \end{aligned} \tag{A.20}$$

$$b(\theta, \boldsymbol{r}) := \int_\Omega \theta(\nabla \cdot \boldsymbol{r}) \, \mathrm{d}\boldsymbol{x}, \tag{A.21}$$

$$c(\boldsymbol{\sigma}, \boldsymbol{r}) := \frac{2}{5} \int_\Omega \boldsymbol{\sigma} : \boldsymbol{\nabla r} \, \mathrm{d}\boldsymbol{x} - \int_\Gamma \left(\frac{3}{20} \sigma_{nn} r_n + \frac{1}{5} \sigma_{nt_1} r_{t_1} + \frac{1}{5} \sigma_{nt_2} r_{t_2} \right) \, \mathrm{d}\boldsymbol{l}, \tag{A.22}$$

$$l_1(\boldsymbol{r}) := - \int_\Gamma \theta^w r_n \, \mathrm{d}\boldsymbol{l}. \tag{A.23}$$

A.1.2. Weak formulation of the energy balance equation

Differential equation. The Equation (II.3) for energy balance reads

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} + \boldsymbol{\nabla} \cdot \boldsymbol{s} = r. \tag{A.24}$$

This equation can be simplified by recognising that $\boldsymbol{\nabla} \cdot \boldsymbol{u} = \dot{m}$ from the Equation (II.1) for mass conservation. Simplified equation reads

$$\boldsymbol{\nabla} \cdot \boldsymbol{s} = r - \dot{m}. \tag{A.25}$$

Weak integral formulation. We multiply the Equation (A.25) with the test function κ and integrate it over the computational domain. Then we get

$$\int_\Omega \kappa(\boldsymbol{\nabla} \cdot \boldsymbol{s}) \, \mathrm{d}\boldsymbol{x} = - \int_\Omega (r - \dot{m}) \kappa \, \mathrm{d}\boldsymbol{x}. \tag{A.26}$$

Weak formulation in functional notation. The weak formulation of the energy balance in functional notation reads

$$b(\boldsymbol{s}, \kappa) = l_2(\kappa), \tag{A.27}$$

where

$$b(\boldsymbol{s}, \kappa) := \int_\Omega \kappa(\boldsymbol{\nabla} \cdot \boldsymbol{s}) \, \mathrm{d}\boldsymbol{x}, \tag{A.28}$$

$$l_2(\kappa) := - \int_\Omega (r - \dot{m}) \kappa \, \mathrm{d}\boldsymbol{x}. \tag{A.29}$$

A.1.3. Weak formulation of the stress balance equation

Differential equation. The Equation (II.4) for stress balance reads

$$\frac{4}{5} (\boldsymbol{\nabla s})_{\mathrm{stf}} + 2(\boldsymbol{\nabla u})_{\mathrm{stf}} + \boldsymbol{\nabla} \cdot \boldsymbol{m} = -\frac{1}{\mathrm{Kn}} \boldsymbol{\sigma}. \tag{A.30}$$

Further normalisation by a factor of 1/2 leads to

$$\frac{2}{5} (\boldsymbol{\nabla s})_{\mathrm{stf}} + (\boldsymbol{\nabla u})_{\mathrm{stf}} + \frac{1}{2} \boldsymbol{\nabla} \cdot \boldsymbol{m} = -\frac{1}{2} \frac{1}{\mathrm{Kn}} \boldsymbol{\sigma}. \tag{A.31}$$

Weak integral formulation. Multiplying the equation (A.31) with the test function $\boldsymbol{\psi}$ gives

$$\begin{aligned}\frac{2}{5}\int_\Omega (\boldsymbol{\nabla s})_{\text{stf}} : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x} + \int_\Omega (\boldsymbol{\nabla u})_{\text{stf}} : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x} \\ + \frac{1}{2}\int_\Omega (\boldsymbol{\nabla}\cdot\boldsymbol{m}) : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x} = -\frac{1}{2}\frac{1}{\mathrm{Kn}}\int_\Omega \boldsymbol{\sigma} : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x}.\end{aligned} \tag{A.32}$$

Integration by parts. Applying the integration-by-parts technique and divergence theorem to each term in the left hand side of the Equation (A.32) give

$$\frac{2}{5}\int_\Omega (\boldsymbol{\nabla s})_{\text{stf}} : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x} = -\frac{2}{5}\int_\Omega \boldsymbol{s}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\ \mathrm{d}\boldsymbol{x} + \frac{2}{5}\int_\Gamma (\boldsymbol{\psi}\cdot\boldsymbol{n})\cdot\boldsymbol{s}\ \mathrm{d}\boldsymbol{l}, \tag{A.33}$$

$$\int_\Omega (\boldsymbol{\nabla u})_{\text{stf}} : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x} = -\int_\Omega \boldsymbol{u}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\ \mathrm{d}\boldsymbol{x} + \int_\Gamma (\boldsymbol{\psi}\cdot\boldsymbol{n})\cdot\boldsymbol{u}\ \mathrm{d}\boldsymbol{l}, \tag{A.34}$$

$$\frac{1}{2}\int_\Omega (\boldsymbol{\nabla}\cdot\boldsymbol{m}) : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x} = -\frac{1}{2}\int_\Omega \boldsymbol{m}\vdots\nabla\boldsymbol{\psi}\ \mathrm{d}\boldsymbol{x} + \frac{1}{2}\int_\Omega (\boldsymbol{m}\cdot\boldsymbol{n}) : \boldsymbol{\psi}\ \mathrm{d}\boldsymbol{l}. \tag{A.35}$$

Boundary condition substitution. The expressions for the boundary integral terms in Equations (A.33) - (A.35) are provided below

$$\boldsymbol{\psi}\cdot\boldsymbol{n} = \begin{bmatrix}\psi_{nn} & \psi_{nt_1} & \psi_{nt_2}\\ \psi_{nt_1} & \psi_{t_1t_1} & \psi_{t_1t_2}\\ \psi_{nt_2} & \psi_{t_1t_2} & -(\psi_{nn}-\psi_{t_2t_2})\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix} = \begin{bmatrix}\psi_{nn}\\ \psi_{nt_1}\\ \psi_{nt_2}\end{bmatrix}, \tag{A.36}$$

$$(\boldsymbol{\psi}\cdot\boldsymbol{n})\cdot\boldsymbol{s} = \begin{bmatrix}\psi_{nn}\\ \psi_{nt_1}\\ \psi_{nt_2}\end{bmatrix}\begin{bmatrix}s_n\\ s_{t_1}\\ s_{t_2}\end{bmatrix} = \psi_{nn}s_n + \psi_{nt_1}s_{t_1} + \psi_{nt_2}s_{t_2}, \tag{A.37}$$

$$(\boldsymbol{\psi}\cdot\boldsymbol{n})\cdot\boldsymbol{u} = \begin{bmatrix}\psi_{nn}\\ \psi_{nt_1}\\ \psi_{nt_2}\end{bmatrix}\begin{bmatrix}u_n\\ u_{t_1}\\ u_{t_2}\end{bmatrix} = \psi_{nn}u_n + \psi_{nt_1}u_{t_1} + \psi_{nt_2}u_{t_2}, \tag{A.38}$$

$$\boldsymbol{m}\cdot\boldsymbol{n} = \begin{bmatrix}m_{nnn} & m_{nnt_1} & m_{nnt_2}\\ m_{nnt_1} & m_{nt_1t_1} & m_{nt_1t_2}\\ m_{nnt_2} & m_{nt_1t_2} & -(m_{nnn}+m_{nt_1t_1})\end{bmatrix}, \tag{A.39}$$

$$(\boldsymbol{m}\cdot\boldsymbol{n}) : \boldsymbol{\psi} = \begin{bmatrix}m_{nnn} & m_{nnt_1} & m_{nnt_2}\\ m_{nnt_1} & m_{nt_1t_1} & m_{nt_1t_2}\\ m_{nnt_2} & m_{nt_1t_2} & -(m_{nnn}+m_{nt_1t_1})\end{bmatrix}\begin{bmatrix}\psi_{nn} & \psi_{nt_1} & \psi_{nt_2}\\ \psi_{nt_1} & \psi_{t_1t_1} & \psi_{t_1t_2}\\ \psi_{nt_2} & \psi_{t_1t_2} & -(\psi_{nn}+\psi_{t_1t_1})\end{bmatrix} \tag{A.40}$$

$$\begin{aligned}&= m_{nnn}\psi_{nn} + m_{nt_1t_1}\psi_{t_1t_1} + 2(m_{nnt_1}\psi_{nt_1} + m_{nnt_2}\psi_{nt_2} + m_{nt_1t_2}\psi_{t_1t_2})\\ &\quad + (m_{nnn}+m_{nt_1t_1})(\psi_{nn}+\psi_{t_1t_1})\end{aligned} \tag{A.41}$$

$$\begin{aligned}&= \frac{6}{4}m_{nnn}\psi_{nn} + 2(m_{nnt_1}\psi_{nt_1} + m_{nnt_2}\psi_{nt_2} + m_{nt_1t_2}\psi_{t_1t_2})\\ &\quad + 2\left(\frac{1}{2}m_{nnn} + m_{nt_1t_1}\right)\left(\frac{1}{2}\psi_{nn} + \psi_{t_1t_1}\right).\end{aligned} \tag{A.42}$$

We substitute the expressions (A.37), (A.38), and (A.42) in the boundary integral terms in Equations (A.33) - (A.35) and subsequently substitute the resulting equations in Equation (A.32). After re-ordering the terms in the boundary integrals we obtain

$$
\begin{aligned}
&-\frac{2}{5}\int_\Omega \boldsymbol{s}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x}-\int_\Omega \boldsymbol{u}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x}-\frac{1}{2}\int_\Omega \boldsymbol{m}\vdots\nabla\boldsymbol{\psi}\,\mathrm{d}\boldsymbol{x}\\
&+\int_\Gamma \underbrace{(\frac{3}{4}m_{nnn}+\frac{2}{5}s_n+u_n)}_{\text{Expression 1}}\psi_{nn}\,\mathrm{d}\boldsymbol{l}+\int_\Gamma \underbrace{(m_{nnt_1}+u_{t_1}+\frac{2}{5}s_{t_1})}_{\text{Expression 2}}\psi_{nt_1}\,\mathrm{d}\boldsymbol{l}\\
&+\int_\Gamma \underbrace{(m_{nnt_2}+u_{t_2}+\frac{2}{5}s_{t_2})}_{\text{Expression 3}}\psi_{nt_2}\,\mathrm{d}\boldsymbol{l}+\int_\Gamma \underbrace{m_{nt_1t_2}}_{\text{Expression 4}}\ \psi_{t_1t_2}\,\mathrm{d}\boldsymbol{l}\\
&+\int_\Omega \left(\underbrace{\frac{1}{2}m_{nnn}+m_{nt_1t_1}}_{\text{Expression 5}}\right)\left(\frac{1}{2}\psi_{nn}+\psi_{t_1t_1}\right)\,\mathrm{d}\boldsymbol{l}=-\frac{1}{2}\frac{1}{\mathrm{Kn}}\int_\Omega \boldsymbol{\sigma}:\boldsymbol{\psi}\,\mathrm{d}\boldsymbol{x}.
\end{aligned}
\tag{A.43}
$$

In the following steps, we simplify the Expressions 1-5 in Equation (A.43) by substituting relevant boundary conditions. We first take the following linear combination of the two boundary conditions in Equations (II.12) and (II.13)

$$\frac{3}{4}m_{nnn}+\frac{3}{20}s_n=\tilde{\chi}\frac{9}{8}\sigma_{nn}. \tag{A.44}$$

Then we reformulate the Equation (II.16) as below:

$$u_n=u_n^w+\epsilon^w\tilde{\chi}((p-p^w)+\sigma_{nn}). \tag{A.45}$$

Comparing the Expression 1 in Equation (A.43) and the reformulated boundary expressions in Equations (A.44) and (A.45) we can conclude that the Expression 1 can be replaced by

$$\tilde{\chi}\frac{9}{8}\sigma_{nn}+\frac{1}{4}s_n+u_n^w+\epsilon^w\tilde{\chi}((p-p^w)+\sigma_{nn}). \tag{A.46}$$

To simplify the Expressions 2 and 3 in Equation (A.43) we reformulate the boundary condition in Equation (II.10) as follows

$$m_{nnt_i}+\frac{1}{5}s_{t_i}+u_{t_i}=\frac{1}{\tilde{\chi}}\sigma_{nt_i}+u_{t_i}^w,\qquad i=1,2. \tag{A.47}$$

Comparing the Expressions 2 and 3 in Equation (A.43) and the reformulated boundary condition in Equation (A.47) we can conclude that the Expressions 2 and 3 can be replaced by

$$\frac{1}{\tilde{\chi}}\sigma_{nt_1}+\frac{1}{5}s_{t_1}+u_{t_1}^w\qquad\text{and}\qquad\frac{1}{\tilde{\chi}}\sigma_{nt_2}+\frac{1}{5}s_{t_2}+u_{t_2}^w\qquad\text{respectively.} \tag{A.48}$$

To simplify Expression 4, we compare the Expression 4 in Equation (A.43) and the boundary condition in Equation (II.14). We conclude that the Expression 4 can be replaced by

$$\tilde{\chi}\left(\frac{1}{2}\sigma_{nn}+\sigma_{t_1t_1}\right). \tag{A.49}$$

Similarly, comparing the Expression 5 in Equation (A.43) and the boundary condition in Equation (II.15), we conclude that the Expression 5 can be replaced by

$$\tilde{\chi}\sigma_{t_1t_2}. \tag{A.50}$$

Substituting the Expressions (A.46), (A.48), (A.49), and (A.50) in the place of Expressions 1-5 in (A.43) we obtain

$$\begin{aligned}
&-\frac{2}{5}\int_\Omega \boldsymbol{s}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x} - \int_\Omega \boldsymbol{u}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x} - \frac{1}{2}\int_\Omega \boldsymbol{m}\vdots\nabla\boldsymbol{\psi}\,\mathrm{d}\boldsymbol{x}\\
&+\int_\Gamma\left(\tilde{\chi}\frac{9}{8}\sigma_{nn}+\frac{1}{4}s_n+u_n^w+\epsilon^w\tilde{\chi}((p-p^w)+\sigma_{nn})\right)\psi_{nn}\,\mathrm{d}\boldsymbol{l}\\
&+\int_\Gamma\left(\frac{1}{\tilde{\chi}}\sigma_{nt_1}+\frac{1}{5}s_{t_1}+u_{t_1}^w\right)\psi_{nt_1}\,\mathrm{d}\boldsymbol{l}+\int_\Gamma\left(\frac{1}{\tilde{\chi}}\sigma_{nt_2}+\frac{1}{5}s_{t_2}+u_{t_2}^w\right)\psi_{nt_2}\,\mathrm{d}\boldsymbol{l}+\tilde{\chi}\int_\Gamma\sigma_{t_1t_2}\psi_{t_1t_2}\,\mathrm{d}\boldsymbol{l}\\
&+\tilde{\chi}\int_\Omega\left(\frac{1}{2}\sigma_{nn}+\sigma_{t_1t_1}\right)\left(\frac{1}{2}\psi_{nn}+\psi_{t_1t_1}\right)\,\mathrm{d}\boldsymbol{l}=-\frac{1}{2}\frac{1}{\mathrm{Kn}}\int_\Omega\boldsymbol{\sigma}:\boldsymbol{\psi}\,\mathrm{d}\boldsymbol{x}.
\end{aligned} \tag{A.51}$$

Closure relation substitution. Substituting closure relation for $\boldsymbol{m}$ in Equation (A.51) yields

$$\begin{aligned}
&-\frac{2}{5}\int_\Omega \boldsymbol{s}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x} - \int_\Omega \boldsymbol{u}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x} + \mathrm{Kn}\int_\Omega(\boldsymbol{\nabla}\boldsymbol{\sigma})_{\mathrm{stf}}\vdots\nabla\boldsymbol{\psi}\,\mathrm{d}\boldsymbol{x}\\
&+\int_\Gamma\left(\tilde{\chi}\frac{9}{8}\sigma_{nn}+\frac{1}{4}s_n+u_n^w+\epsilon^w\tilde{\chi}((p-p^w)+\sigma_{nn})\right)\psi_{nn}\,\mathrm{d}\boldsymbol{l}\\
&+\int_\Gamma\left(\frac{1}{\tilde{\chi}}\sigma_{nt_1}+\frac{1}{5}s_{t_1}+u_{t_1}^w\right)\psi_{nt_1}\,\mathrm{d}\boldsymbol{l}+\int_\Gamma\left(\frac{1}{\tilde{\chi}}\sigma_{nt_2}+\frac{1}{5}s_{t_2}+u_{t_2}^w\right)\psi_{nt_2}\,\mathrm{d}\boldsymbol{l}+\tilde{\chi}\int_\Gamma\sigma_{t_1t_2}\psi_{t_1t_2}\,\mathrm{d}\boldsymbol{l}\\
&+\tilde{\chi}\int_\Omega\left(\frac{1}{2}\sigma_{nn}+\sigma_{t_1t_1}\right)\left(\frac{1}{2}\psi_{nn}+\psi_{t_1t_1}\right)\,\mathrm{d}\boldsymbol{l}=-\frac{1}{2}\frac{1}{\mathrm{Kn}}\int_\Omega\boldsymbol{\sigma}:\boldsymbol{\psi}\,\mathrm{d}\boldsymbol{x}.
\end{aligned} \tag{A.52}$$

Weak formulation in functional notation. After re-ordering the terms in Equation (A.52), the final weak formulation for the stress balance equation in functional notation reads

$$\tilde{c}(\boldsymbol{s},\boldsymbol{\psi})+d(\boldsymbol{\sigma},\boldsymbol{\psi})-e(\boldsymbol{u},\boldsymbol{\psi})+f(p,\boldsymbol{\psi})=l_3(\boldsymbol{\psi}), \tag{A.53}$$

where

$$\tilde{c}(\boldsymbol{s},\boldsymbol{\psi}):=-\frac{2}{5}\int_\Omega\boldsymbol{s}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x}+\int_\Gamma\left(\frac{1}{4}s_n\psi_{nn}+\frac{1}{5}s_{t_1}\psi_{nt_1}+\frac{1}{5}s_{t_2}\psi_{nt_2}\right)\,\mathrm{d}\boldsymbol{l}, \tag{A.54}$$

$$\begin{aligned}
d(\boldsymbol{\sigma},\boldsymbol{\psi}):=&\,\mathrm{Kn}\int_\Omega(\boldsymbol{\nabla}\boldsymbol{\sigma})_{\mathrm{stf}}\vdots\boldsymbol{\nabla}\boldsymbol{\psi})\,\mathrm{d}\boldsymbol{x}+\frac{1}{2}\frac{1}{\mathrm{Kn}}\int_\Omega\boldsymbol{\sigma}:\boldsymbol{\psi}\,\mathrm{d}\boldsymbol{x}\\
&+\int_\Gamma\left(\tilde{\chi}\left(\frac{9}{8}\sigma_{nn}\psi_{nn}+\sigma_{t_1t_2}\psi_{t_1t_2}\right)+\frac{1}{\tilde{\chi}}\left(\sigma_{nt_1}\psi_{nt_1}+\sigma_{nt_2}\psi_{nt_2}\right)\right)\,\mathrm{d}\boldsymbol{l}\\
&+\tilde{\chi}\int_\Omega\left(\frac{1}{2}\sigma_{nn}+\sigma_{t_1t_1}\right)\left(\frac{1}{2}\psi_{nn}+\psi_{t_1t_1}\right)\,\mathrm{d}\boldsymbol{l},
\end{aligned} \tag{A.55}$$

$$e(\boldsymbol{u}, \boldsymbol{\psi}) := \int_\Omega \boldsymbol{u} \cdot (\boldsymbol{\nabla} \cdot \boldsymbol{\psi}) \, \mathrm{d}\boldsymbol{x}, \tag{A.56}$$

$$f(p, \boldsymbol{\psi}) := \epsilon^w \tilde{\chi} \int_\Gamma p \psi_{nn} \, \mathrm{d}\boldsymbol{l}, \tag{A.57}$$

$$l_3(\boldsymbol{\psi}) := -\int_\Gamma (u_n^w \psi_{nn} + u_{t_1}^w \psi_{nt_1} + u_{t_1}^w \psi_{nt_2} - \epsilon^w \tilde{\chi} p^w \psi_{nn}) \, \mathrm{d}\boldsymbol{l}. \tag{A.58}$$

A.1.4. Weak formulation of the momentum balance equation

Differential equation. The momentum balance equation reads

$$\boldsymbol{\nabla} p + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \boldsymbol{b}. \tag{A.59}$$

Weak integral formulation. Multiply the equation with the test function $\boldsymbol{v}$ and integrate it over the computational domain,

$$\int_\Omega \boldsymbol{\nabla} p \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} + \int_\Omega (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x} = \int_\Omega \boldsymbol{b} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}. \tag{A.60}$$

Weak formulation in functional notation. The weak formulation for the momentum balance in functional notation reads

$$e(\boldsymbol{\sigma}, \boldsymbol{v}) + g(p, \boldsymbol{v}) = l_4(\boldsymbol{v}), \tag{A.61}$$

where

$$e(\boldsymbol{\sigma}, \boldsymbol{v}) := \int_\Omega (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}, \tag{A.62}$$

$$g(p, \boldsymbol{v}) := \int_\Omega \boldsymbol{\nabla} p \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}, \tag{A.63}$$

$$l_4(\boldsymbol{v}) := \int_\Omega \boldsymbol{b} \cdot \boldsymbol{v} \, \mathrm{d}\boldsymbol{x}. \tag{A.64}$$

A.1.5. Weak formulation of the mass balance equation

Differential equation. The mass balance equation reads

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = \dot{m}. \tag{A.65}$$

Weak integral formulation. Multiply the mass balance equation with the test function q and integrate it over the computational domain. Then, we get

$$\int_\Omega (\boldsymbol{\nabla} \cdot \boldsymbol{u}) q \, \mathrm{d}\boldsymbol{x} = \int_\Omega \dot{m} q \, \mathrm{d}\boldsymbol{x}. \tag{A.66}$$

Integration by parts. After applying integration-by-parts technique and divergence theorem to the term on the left hand side of the Equation (A.66), we obtain

$$\int_{\Omega} (\boldsymbol{\nabla} \cdot \boldsymbol{u}) q \, \mathrm{d}\boldsymbol{x} = -\int_{\Omega} u \cdot \boldsymbol{\nabla} q \, \mathrm{d}\boldsymbol{x} + \int_{\Gamma} (u \cdot \boldsymbol{n}) q \, \mathrm{d}\boldsymbol{l}. \tag{A.67}$$

Boundary condition substitution. The expression for the boundary integral term in Equation (A.67) is as follows:

$$(\boldsymbol{u} \cdot \boldsymbol{n}) q = q u_n. \tag{A.68}$$

We first substitute the Equation (A.68) in the boundary integral term in Equation (A.67) and subsequently we substitute the resulting equation in Equation (A.66). Then, we obtain

$$-\int_{\Omega} u \cdot \boldsymbol{\nabla} q \, \mathrm{d}\boldsymbol{x} + \int_{\Gamma} q u_n \, \mathrm{d}\boldsymbol{l}. = \int_{\Gamma} \dot{m} q \, \mathrm{d}\boldsymbol{x}. \tag{A.69}$$

We can straightforwardly substitute the boundary expression in (II.16) for u_n and we obtain

$$-\int_{\Omega} u \cdot \boldsymbol{\nabla} q \, \mathrm{d}\boldsymbol{x} + \int_{\Gamma} q(u_n^w + \epsilon^w \tilde{\chi}((p - p^w) + \sigma_{nn})) \, \mathrm{d}\boldsymbol{l} = \int_{\Omega} \dot{m} q \, \mathrm{d}\boldsymbol{x}. \tag{A.70}$$

Re-ordering the Equation (A.70) yields

$$-\int_{\Omega} u \cdot \boldsymbol{\nabla} q \, \mathrm{d}\boldsymbol{x} + \epsilon^w \tilde{\chi} \int_{\Gamma} pq \, \mathrm{d}\boldsymbol{l} + \epsilon^w \tilde{\chi} \int_{\Gamma} \sigma_{nn} q \, \mathrm{d}\boldsymbol{l} = \int_{\Omega} \dot{m} q \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma} (u_n^w - \epsilon^w \tilde{\chi} p^w) q \, \mathrm{d}\boldsymbol{l}. \tag{A.71}$$

Weak formulation in functional notation. The weak formulation for the mass balance in functional notation reads

$$f(\boldsymbol{\sigma}, q) - g(\boldsymbol{u}, q) + h(p, q) = l_5(q), \tag{A.72}$$

where

$$f(\boldsymbol{\sigma}, q) := \epsilon^w \tilde{\chi} \int_{\Gamma} \sigma_{nn} q \, \mathrm{d}\boldsymbol{l} \tag{A.73}$$

$$g(\boldsymbol{u}, q) := \int_{\Omega} u \cdot \boldsymbol{\nabla} q \, \mathrm{d}\boldsymbol{x} \tag{A.74}$$

$$h(p, q) := \epsilon^w \tilde{\chi} \int_{\Gamma} pq \, \mathrm{d}\boldsymbol{l} \tag{A.75}$$

$$l_5(q) := \int_{\Omega} \dot{m} q \, \mathrm{d}\boldsymbol{x} - \int_{\Gamma} (u_n^w - \epsilon^w \tilde{\chi} p^w) q \, \mathrm{d}\boldsymbol{l}. \tag{A.76}$$

A.2. Data for flow-over-a-sphere convergence plots

Table A.1.: Flow-over-a-sphere convergence plot data: L2 error norm for Kn = 0.005

h	8.608e-01	5.315e-01	3.033e-01	1.668e-01	9.580e-02	4.878e-02
θ	1.077e-01	6.304e-02	1.601e-02	2.841e-03	4.173e-04	1.511e-04
s_x	8.414e-01	5.068e-01	1.473e-01	4.388e-02	1.720e-02	8.653e-03
s_y	8.461e-01	5.112e-01	1.483e-01	4.315e-02	1.711e-02	8.675e-03
s_z	8.469e-01	5.304e-01	1.844e-01	8.703e-02	4.692e-02	2.281e-02
p	6.423e-02	5.089e-02	4.055e-02	1.804e-02	7.691e-03	2.994e-03
u_x	9.692e-01	2.942e-01	9.180e-02	2.039e-02	4.218e-03	9.884e-04
u_y	9.499e-01	2.958e-01	8.703e-02	2.044e-02	4.190e-03	9.917e-04
u_z	3.339e-01	1.043e-01	2.554e-02	4.711e-03	7.177e-04	1.305e-04
σ_{xx}	7.010e-01	4.074e-01	1.660e-01	5.230e-02	1.894e-02	7.156e-03
σ_{xy}	7.936e-01	5.422e-01	2.128e-01	7.149e-02	2.657e-02	1.017e-02
σ_{xz}	6.402e-01	3.498e-01	1.371e-01	4.333e-02	1.522e-02	5.618e-03
σ_{yy}	6.877e-01	4.373e-01	1.539e-01	5.227e-02	1.898e-02	7.163e-03
σ_{yz}	6.509e-01	3.528e-01	1.270e-01	4.293e-02	1.514e-02	5.649e-03

Table A.2.: Flow-over-a-sphere convergence plot data: l_∞ error norm for Kn = 0.005

h	8.608e-01	5.315e-01	3.033e-01	1.668e-01	9.580e-02	4.878e-02
θ	4.350e-01	2.736e-01	8.797e-02	2.223e-02	5.345e-03	1.860e-03
s_x	8.260e-01	5.260e-01	3.262e-01	2.602e-01	9.498e-02	5.618e-02
s_y	7.945e-01	6.038e-01	2.815e-01	1.943e-01	9.808e-02	5.323e-02
s_z	8.506e-01	5.768e-01	2.998e-01	2.552e-01	1.892e-01	1.244e-01
p	2.237e-01	2.352e-01	2.859e-01	1.723e-01	7.468e-02	3.799e-02
u_x	6.173e-01	7.716e-01	4.815e-01	1.998e-01	6.481e-02	2.040e-02
u_y	5.615e-01	7.547e-01	4.621e-01	2.042e-01	6.913e-02	2.062e-02
u_z	1.074e+00	5.217e-01	2.306e-01	9.017e-02	2.444e-02	6.470e-03
σ_{xx}	9.004e-01	6.912e-01	3.919e-01	2.080e-01	9.160e-02	5.222e-02
σ_{xy}	1.042e+00	9.948e-01	5.369e-01	3.129e-01	1.336e-01	7.661e-02
σ_{xz}	6.423e-01	5.198e-01	3.329e-01	1.471e-01	8.546e-02	3.481e-02
σ_{yy}	8.138e-01	7.454e-01	3.811e-01	2.027e-01	9.126e-02	4.767e-02
σ_{yz}	7.325e-01	5.435e-01	3.088e-01	1.540e-01	6.698e-02	3.157e-02

Table A.3.: Flow-over-a-sphere convergence plot data: L2 error norm for Kn = 0.2

h	8.608e-01	5.315e-01	3.033e-01	1.668e-01	9.580e-02	4.878e-02
θ	2.053e-02	9.981e-03	4.102e-03	2.119e-03	1.230e-03	7.661e-04
s_x	3.947e-01	1.687e-01	5.278e-02	1.835e-02	7.402e-03	3.719e-03
s_y	3.929e-01	1.708e-01	5.228e-02	1.842e-02	7.409e-03	3.718e-03
s_z	4.100e-01	1.868e-01	6.029e-02	2.183e-02	9.104e-03	4.596e-03
p	8.830e-01	5.129e-01	1.527e-01	3.461e-02	8.194e-03	3.281e-03
u_x	1.069e+00	5.501e-01	1.315e-01	3.681e-02	9.968e-03	2.734e-03
u_y	1.068e+00	5.460e-01	1.305e-01	3.694e-02	9.846e-03	2.775e-03
u_z	2.795e-01	1.407e-01	3.928e-02	1.017e-02	2.223e-03	5.371e-04
σ_{xx}	8.615e-01	4.298e-01	1.209e-01	2.846e-02	5.872e-03	1.771e-03
σ_{xy}	7.404e-01	3.901e-01	1.113e-01	2.628e-02	6.325e-03	2.286e-03
σ_{xz}	9.008e-01	4.411e-01	1.242e-01	2.964e-02	5.690e-03	1.520e-03
σ_{yy}	8.591e-01	4.300e-01	1.184e-01	2.858e-02	5.858e-03	1.772e-03
σ_{yz}	8.931e-01	4.445e-01	1.225e-01	2.969e-02	5.686e-03	1.521e-03

Table A.4.: Flow-over-a-sphere convergence plot data: l_∞ error norm for Kn = 0.2

h	8.608e-01	5.315e-01	3.033e-01	1.668e-01	9.580e-02	4.878e-02
θ	1.322e-01	8.663e-02	6.205e-02	4.400e-02	3.206e-02	2.371e-02
s_x	3.617e-01	1.938e-01	7.932e-02	3.197e-02	9.930e-03	5.116e-03
s_y	3.780e-01	2.091e-01	7.352e-02	3.091e-02	1.136e-02	5.131e-03
s_z	4.916e-01	2.352e-01	8.971e-02	3.736e-02	1.606e-02	7.480e-03
p	8.432e-01	5.548e-01	2.240e-01	7.416e-02	2.122e-02	6.576e-03
u_x	1.302e+00	7.817e-01	3.433e-01	2.237e-01	1.246e-01	4.282e-02
u_y	1.381e+00	7.666e-01	2.862e-01	2.246e-01	1.091e-01	5.764e-02
u_z	1.294e+00	8.221e-01	3.338e-01	1.752e-01	7.222e-02	2.741e-02
σ_{xx}	9.223e-01	5.190e-01	3.199e-01	1.193e-01	3.486e-02	1.540e-02
σ_{xy}	8.051e-01	7.105e-01	4.454e-01	2.193e-01	6.192e-02	2.292e-02
σ_{xz}	7.252e-01	4.472e-01	2.311e-01	1.293e-01	2.895e-02	1.072e-02
σ_{yy}	9.120e-01	6.504e-01	3.160e-01	1.367e-01	4.143e-02	1.283e-02
σ_{yz}	7.764e-01	4.924e-01	2.366e-01	1.008e-01	3.109e-02	9.204e-03

A.3. Data for thickness comparison in radiometer simulation

Table A.5.: Radiometric force data on a vane with different thicknesses.

Kn	Force Component	Circle (C)		Diamond (D)		Rectangle (R)	
		$w = 0.01$	$w = 0.1$	$w = 0.01$	$w = 0.1$	$w = 0.01$	$w = 0.1$
0.01	$F_z^{(\text{hot+cold})}$	1.081e-04	1.433e-04	1.137e-04	1.505e-04	1.159e-04	1.591e-04
	$F_z^{(\text{transition})}$	-2.262e-06	-6.281e-05	-2.067e-06	-6.726e-05	-2.377e-06	-7.019e-05
	$F_z^{(\text{net})}$	1.059e-04	8.050e-05	1.116e-04	8.325e-05	1.135e-04	8.889e-05
0.02	$F_z^{(\text{hot+cold})}$	2.946e-04	3.569e-04	3.036e-04	3.671e-04	3.104e-04	3.895e-04
	$F_z^{(\text{transition})}$	-1.460e-06	-8.873e-05	-1.206e-06	-9.518e-05	-1.540e-06	-9.896e-05
	$F_z^{(\text{net})}$	2.931e-04	2.682e-04	3.024e-04	2.719e-04	3.088e-04	2.905e-04
0.04	$F_z^{(\text{hot+cold})}$	7.518e-04	8.333e-04	7.554e-04	8.338e-04	7.760e-04	8.847e-04
	$F_z^{(\text{transition})}$	-3.960e-08	-8.025e-05	3.123e-07	-8.639e-05	4.770e-09	-8.861e-05
	$F_z^{(\text{net})}$	7.518e-04	7.531e-04	7.557e-04	7.474e-04	7.760e-04	7.961e-04
0.08	$F_z^{(\text{hot+cold})}$	1.696e-03	1.781e-03	1.660e-03	1.738e-03	1.714e-03	1.832e-03
	$F_z^{(\text{transition})}$	2.337e-06	-2.647e-05	2.595e-06	-2.991e-05	2.531e-06	-2.801e-05
	$F_z^{(\text{net})}$	1.698e-03	1.755e-03	1.663e-03	1.708e-03	1.717e-03	1.804e-03
0.16	$F_z^{(\text{hot+cold})}$	2.946e-03	3.019e-03	2.835e-03	2.913e-03	2.919e-03	3.010e-03
	$F_z^{(\text{transition})}$	5.732e-06	5.711e-05	5.469e-06	5.508e-05	6.055e-06	6.140e-05
	$F_z^{(\text{net})}$	2.952e-03	3.076e-03	2.841e-03	2.968e-03	2.925e-03	3.071e-03
0.32	$F_z^{(\text{hot+cold})}$	3.210e-03	3.259e-03	3.077e-03	3.148e-03	3.114e-03	3.160e-03
	$F_z^{(\text{transition})}$	6.816e-06	1.012e-04	6.116e-06	9.956e-05	7.133e-06	1.055e-04
	$F_z^{(\text{net})}$	3.217e-03	3.361e-03	3.083e-03	3.248e-03	3.122e-03	3.265e-03
0.64	$F_z^{(\text{hot+cold})}$	1.994e-03	2.018e-03	1.918e-03	1.959e-03	1.901e-03	1.919e-03
	$F_z^{(\text{transition})}$	4.015e-06	6.848e-05	3.507e-06	6.783e-05	4.225e-06	7.029e-05
	$F_z^{(\text{net})}$	1.998e-03	2.087e-03	1.922e-03	2.027e-03	1.905e-03	1.989e-03
1.28	$F_z^{(\text{hot+cold})}$	8.053e-04	8.138e-04	7.813e-04	7.961e-04	7.562e-04	7.665e-04
	$F_z^{(\text{transition})}$	1.512e-06	2.761e-05	1.312e-06	2.761e-05	1.595e-06	2.817e-05
	$F_z^{(\text{net})}$	8.069e-04	8.415e-04	7.826e-04	8.237e-04	7.578e-04	7.947e-04
2.56	$F_z^{(\text{hot+cold})}$	2.616e-04	2.635e-04	2.576e-04	2.614e-04	2.402e-04	2.470e-04
	$F_z^{(\text{transition})}$	4.824e-07	8.868e-06	4.302e-07	9.053e-06	4.950e-07	8.968e-06
	$F_z^{(\text{net})}$	2.621e-04	2.724e-04	2.580e-04	2.705e-04	2.407e-04	2.559e-04
5.12	$F_z^{(\text{hot+cold})}$	7.946e-05	7.964e-05	8.042e-05	8.112e-05	7.012e-05	7.412e-05
	$F_z^{(\text{transition})}$	1.563e-07	2.660e-06	1.462e-07	2.930e-06	1.451e-07	2.639e-06
	$F_z^{(\text{net})}$	7.962e-05	8.230e-05	8.057e-05	8.405e-05	7.026e-05	7.676e-05

A.4. Data for length comparison in radiometer simulation

Table A.6.: Radiometric force data on a vane with different aspect ratios.

Kn	Force Component	R1 l=0.5, d=2.0, w=0.01	R2 l=1.0, d=1.0, w=0.01	R3 l=2.0, d=0.5, w=0.01
0.01	$F_z^{(\text{hot+cold})}$	1.496e-04	1.159e-04	1.416e-04
	$F_z^{(\text{transition})}$	-3.293e-06	-2.377e-06	-3.005e-06
	$F_z^{(\text{net})}$	1.463e-04	1.135e-04	1.386e-04
0.02	$F_z^{(\text{hot+cold})}$	4.014e-04	3.104e-04	3.759e-04
	$F_z^{(\text{transition})}$	-2.309e-06	-1.540e-06	-1.898e-06
	$F_z^{(\text{net})}$	3.991e-04	3.088e-04	3.740e-04

0.04	$F_z^{(\text{hot+cold})}$	9.954e-04	7.760e-04	9.181e-04
	$F_z^{(\text{transition})}$	-2.819e-07	4.770e-09	2.035e-07
	$F_z^{(\text{net})}$	9.951e-04	7.760e-04	9.183e-04
0.08	$F_z^{(\text{hot+cold})}$	2.089e-03	1.714e-03	1.885e-03
	$F_z^{(\text{transition})}$	3.451e-06	2.531e-06	3.843e-06
	$F_z^{(\text{net})}$	2.093e-03	1.717e-03	1.889e-03
0.16	$F_z^{(\text{hot+cold})}$	3.109e-03	2.919e-03	2.774e-03
	$F_z^{(\text{transition})}$	7.703e-06	6.055e-06	7.453e-06
	$F_z^{(\text{net})}$	3.117e-03	2.925e-03	2.782e-03
0.32	$F_z^{(\text{hot+cold})}$	2.737e-03	3.114e-03	2.518e-03
	$F_z^{(\text{transition})}$	7.439e-06	7.133e-06	6.808e-06
	$F_z^{(\text{net})}$	2.744e-03	3.122e-03	2.525e-03
0.64	$F_z^{(\text{hot+cold})}$	1.419e-03	1.901e-03	1.399e-03
	$F_z^{(\text{transition})}$	3.817e-06	4.225e-06	3.465e-06
	$F_z^{(\text{net})}$	1.423e-03	1.905e-03	1.403e-03
1.28	$F_z^{(\text{hot+cold})}$	5.154e-04	7.562e-04	5.458e-04
	$F_z^{(\text{transition})}$	1.346e-06	1.595e-06	1.262e-06
	$F_z^{(\text{net})}$	5.168e-04	7.578e-04	5.471e-04
2.56	$F_z^{(\text{hot+cold})}$	1.573e-04	2.402e-04	1.782e-04
	$F_z^{(\text{transition})}$	3.974e-07	4.950e-07	4.187e-07
	$F_z^{(\text{net})}$	1.577e-04	2.407e-04	1.786e-04
5.12	$F_z^{(\text{hot+cold})}$	4.503e-05	7.012e-05	5.593e-05
	$F_z^{(\text{transition})}$	1.104e-07	1.451e-07	1.428e-07
	$F_z^{(\text{net})}$	4.515e-05	7.026e-05	5.608e-05

Appendix

B Generic Moment Systems

B.1. Implementation of G13 moment system in FEniCS

In this section, the Python implementation of the G13 moment system is explained snippet by snippet. By going through the following sections, one will see that it is easy to adapt or extend the implementation to any order of moment system. This is why we have tried to strike a balance between conciseness and comprehensibility in the following program. Furtermore, as we discover new things, software evolves over time. Therefore, for the most recent versions, please refer to the online repository on GitLab [34].

Listing B.1: Importing libraries. The following packages are required for successfully running the program:

- `dolfin`: This core library provides numerous user interfaces for accessing the underlying features of FEniCS.
- `numpy`: This library is necessary for converting system matrices and other structures to the `numpy.array` datatype expected by `dolfin`.
- `math`: This library is used for accessing mathematical constants.
- `sys`: This library is needed for reading and writing files from custom paths and executing custom commands.
- `yaml`: Although not strictly necessary, this library is used to sanitize user input.

Each package plays a crucial role in ensuring the proper functionality and flexibility of the program.

```
1  # Import packages
2  import sys
3  import dolfin as df
4  import numpy as np
5  import yaml  # version higher than 5.0
6  import math
7  df.parameters['ghost_mode'] = 'shared_facet'  # for parallel program
```

Listing B.1: Snippet to import required libraries

Listing B.2: Read user input. User must input an `yaml` type file which contains details about mesh location, finite element types, Knudsen number, stabilisation, and boundary conditions. A sample file is provided in Section B.2.

```
8  if len(sys.argv) > 1:
9    user_given_input_file = sys.argv[1]
10 else:
11   print("Provide an input file")
12   quit()
13 with open(user_given_input_file, 'r') as input_file:
14   input_file = yaml.load(input_file, Loader=yaml.FullLoader)
```

Listing B.2: Snippet to read user input

```
15 my_current_mesh = input_file["MESH_PARAMS"]['mesh_list'][0]
16 mesh_number = int(list(my_current_mesh)[-4])
17 mesh = df.Mesh()
18 hdf = df.HDF5File(mesh.mpi_comm(), my_current_mesh, "r")
19 hdf.read(mesh, "/mesh", False)
20 dim = mesh.topology().dim()
21 subdomains = df.MeshFunction("size_t", mesh, dim)
22 hdf.read(subdomains, "/subdomains")
23 boundaries = df.MeshFunction("size_t", mesh, dim-1)
24 hdf.read(boundaries, "/boundaries")
```

Listing B.3: Snippet to read mesh data

```
25 number_of_moments = input_file["GENERAL_PARAMS"]['number_of_moments']
26 problem_type = input_file["GENERAL_PARAMS"]['problem_type']
27 ve = df.VectorElement("Lagrange", mesh.ufl_cell(), 1, dim=number_of_moments)
28 se = df.FiniteElement("R", mesh.ufl_cell(), 0)
29 mel = df.MixedElement([ve, se])
30 V = df.FunctionSpace(mesh, ve)
31 W = df.FunctionSpace(mesh, se)
32 MFS = df.FunctionSpace(mesh, ve*se)
33 (v, dd) = df.TestFunctions(MFS)
34 v = df.as_vector(v)
35 if problem_type == 'nonlinear':
36     u_mixed = df.Function(MFS)
37     (u, cc) = df.split(u_mixed)
38 elif problem_type == 'linear':
39     (u, cc) = df.TrialFunctions(MFS)
40 u = df.as_vector(u)
41 nv = df.FacetNormal(mesh)
42 ds = df.Measure("ds", domain=mesh, subdomain_data=boundaries)
43 dS = df.Measure("dS", domain=mesh, subdomain_data=boundaries)
```

Listing B.4: Snippet to create function space

Listing B.2: Read mesh data. The snippet shows the code for reading the mesh data from a `h5` file. It also stores the data of subdomains which contain tags of boundaries. This data will be used in the latter snippets when we construct the finite element function space.

Listing B.4: Create function space. This snippet gathers information regarding the moment system: the number of moment coefficients and whether the production terms are linear or nonlinear. Depending on the number of moment variables, it defines finite elements, as well as test and trial function spaces.

```
44  moment_order = input_file["GENERAL_PARAMS"]['moment_order']
45  Kn = input_file["GENERAL_PARAMS"]['Kn']
46  A_x = np.array([...]) #input the system matrix A_x
47  A_y = np.array([... ]) #input the  system matrix A_y
48  Block_A = np.array([...]) #input the \mathbb{A} matrix
49
50
51  def L_matrix(bc_id):
52    eps_w = input_file["BOUNDARY_PARAMS"]['bc'][bc_id]['eps_w']
53    chi2 = math.sqrt(2/math.pi) * input_file["BOUNDARY_PARAMS"]['bc'][
54      bc_id]['chi']
55    L_matrix = np.array([
56      [eps_w, 0.0, 0.0],
57      [0.0, chi2, 0.0],
58      [0.0, 0.0, 2*chi2]
59    ])
60    return L_matrix
61
62
63  P_Coeff = np.array([...])/Kn # input the linear production term matrix
64  P_even = np.array([...]) # input the even permuation matrix
65  P_odd = np.array([...]) # input the odd permutation matrix
66  Symm = np.array([...]) # input the symmetriser matrix
67  T_n = np.array([...]) # input the transformation matrix
68  SA_x = df.as_matrix(np.dot(Symm, A_x))
69  SA_y = df.as_matrix(np.dot(Symm, A_y))
70  SP_Coeff = df.as_matrix(np.dot(Symm, P_Coeff))
71  Block_A_trans = np.transpose(Block_A)
72
73
74  def L_inverse(bc_id):
75    L_inverse = np.linalg.inv(L_matrix(bc_id))
76    return L_inverse
```

Listing B.5: Snippet to read system matrices

Listing B.5: Read system matrices. This snippet demonstrates how to read and process system matrices required for the moment system solver. It begins by extracting general parameters such as the moment order and Knudsen number from the input file. Key matrices like

A_x, A_y, and other coefficient matrices are defined as numpy arrays. For precise definition of these matrices, we refer to Section B.3. The snippet includes functions to calculate the boundary condition matrices L and its inverse based on boundary parameters. The system matrices are symmetrized by applying transformations, and symmetrized matrices are converted into `dolfin`-compatible formats. This setup ensures that the required system matrices are correctly formatted and ready for further computations.

```
77  def bc1(bc_id):
78    BC1 = np.dot( np.dot( np.dot( np.transpose(T_n), np.transpose(P_odd)),
      ↪  L_inverse(bc_id)), np.dot( P_odd, T_n))
79    BC1 = df.as_matrix(BC1)  # UFL conversion
80    return BC1
81  def bc2(bc_id):
82    BC2 = np.dot( np.dot( np.dot( np.dot( np.dot( np.transpose(T_n),
      ↪  np.transpose(P_even)), Block_A_trans), L_matrix(bc_id)), Block_A),
      ↪  np.dot( P_even, T_n))
83    BC2 = df.as_matrix(BC2)  # UFL conversion
84    return BC2
85  BC1_rhs = np.dot( np.transpose(T_n), np.dot( np.transpose(P_even),
    ↪  Block_A_trans))
86  BC1_rhs = df.as_matrix(BC1_rhs)  # UFL conversion
87
88  def bc2_rhs(bc_id):
89    BC2_rhs = np.dot( np.transpose(T_n), np.dot( np.transpose(P_odd),
      ↪  L_inverse(bc_id)))
90    BC2_rhs = df.as_matrix(BC2_rhs)  # UFL conversion
91    return BC2_rhs
```

Listing B.6: Snippet to define homogeneous boundary conditions

Listing B.6: Define homogeneous boundary condition matrices. This snippet defines functions, `bc1` and `bc2` `bc2_rhs`, that calculate homogeneous boundary condition matrices for the moment system. These functions implement the surface integral expressions in the left-hand side and the right-hand side of the Equation III.166. These resulting matrices are converted into a format compatible with FEniCS using `df.as_matrix`, ensuring they are ready for use in the `dolfin` framework.

Listing B.7: Define inhomogeneity. This snippet defines a function `inhomogeneity` that calculates the inhomogeneous boundary condition vector for a given boundary condition ID (`bc_id`). The function uses expressions involving user-defined reference variables such as wall temperature (`theta_w`), tangential and normal wall velocities (`u_t_w` and `u_n_w`), and wall pressure (`p_w`). These variables are converted into expressions suitable for FEniCS and used to compute the inhomogeneous boundary condition vector `G_rhs_list`. The result is transformed into a vector format compatible with FEniCS using `df.as_vector`, which is returned for further use in the `dolfin` framework.

```
92  def inhomogeneity(bc_id):
93    phi_local = df.Expression("atan2(x[1],x[0])", degree=2)
94    chi2 = math.sqrt(2/math.pi) * input_file["BOUNDARY_PARAMS"]['bc'][
      ↪  bc_id]['chi']
95    eps_w = input_file["BOUNDARY_PARAMS"]['bc'][bc_id]['eps_w']
96    theta_w = ref_R * input_file["BOUNDARY_PARAMS"]['bc'][bc_id]['theta_w']
97    u_t_w = input_file["BOUNDARY_PARAMS"]['bc'][bc_id]['u_t_w']
98    u_n_w = input_file["BOUNDARY_PARAMS"]['bc'][bc_id]['u_n_w']
99    p_w = input_file["BOUNDARY_PARAMS"]['bc'][bc_id]['p_w']
100   u_n_w = df.Expression("{}".format(u_n_w), degree=2, phi=phi_local)
101   u_t_w = df.Expression("{}".format(u_t_w), degree=2, phi=phi_local)
102   p_w = df.Expression("{}".format(p_w), degree=2, phi=phi_local)
103   theta_w = df.Expression("{}".format(theta_w), degree=2, phi=phi_local)
104   G_rhs_list = [
105           eps_w * (u[0] - (2/3) * u[3] +  u[4]),
106           -chi2 * (u_t_w/math.sqrt(ref_theta)),
107           2 * chi2 * ((theta_w - ref_theta)/ref_theta)
108           ]
109   return df.as_vector(G_rhs_list)  # UFL form conversion
```

Listing B.7: Snippet to define inhomogeneous boundary conditions

```
118 bc = 0.0
119 df.ds = ds
120 bc -= sum([(0.5 * df.inner(v, ( (BC1_rhs - bc2_rhs(bc_id)) *
    ↪  inhomogeneity(bc_id)))) * df.ds( bc_id) for bc_id in
    ↪  input_file["BOUNDARY_PARAMS"]["bc"].keys()])
```

Listing B.8: Snippet to apply boundary conditions

Listing B.8: Apply inhomogeneous boundary condition. This snippet combines Listing B.6 and Listing B.7 to apply the user-defined inhomogeneous boundary conditions.

```
110 a = 0.0
111 df.ds = ds
112 a += (+0.5 * ((df.inner(v, (SA_x * df.Dx(u, 0)))) + (df.inner( v, (SA_y *
    ↪  df.Dx(u, 1)))))) * df.dx
113 a += (-0.5 * ((df.inner(df.Dx(v, 0), (SA_x * u))) + (df.inner( df.Dx(v, 1),
    ↪  (SA_y * u))))) * df.dx
114 a += sum([(0.5 * df.inner(v, ((bc1(bc_id) + bc2(bc_id)) * u))) * df.ds(
    ↪  bc_id) for bc_id in input_file["BOUNDARY_PARAMS"]["bc"].keys()])
115 a += (df.inner(v, (SP_Coeff * u))) * df.dx
116 a += (df.inner(v[0], cc)) * df.dx
117 a += (df.inner(u[0], dd)) * df.dx
```

Listing B.9: Snippet to create bilinear form

Listing B.9: Creating the bilinear form. This snippet defines the bilinear form for the variational formulation, encompassing all terms on the left-hand side of the Equation III.166. It

constructs the form by assembling contributions from various components, including terms involving the system matrices `SA_x` and `SA_y`, boundary condition matrices `bc1` and `bc2`, and specific coefficients such as `SP_Coeff`. The bilinear form is built using inner products and integrals over the domain and boundary surfaces, making it suitable for use in the finite element method within the FEniCS framework.

```
121  def Collision_nonlinearity():
122    nl = 0.0
123    nl = np.array(
124      [
125        df.Constant(0.0),
126        df.Constant(0.0),
127        df.Constant(0.0),
128        df.Constant(0.0),
129        -(u[0]+1) * u[4] + ((2.0/3.0) * u[1]**2 - ( 1.0/3.0) * u[2]**2),
           ↪  -(u[0]+1) * u[5] + (u[1] * u[2]),
130        -(u[0]+1) * u[6] + ((2.0/3.0) * u[2]**2 - ( 1.0/3.0) * u[1]**2),
131        -(2/3) * (u[0]+1) * u[7] + (((10/9) * u[3] * u[1]) + ( (1/3) * (u[4]
           ↪  * u[1] + u[5] * u[2]))),
132        -(2/3) * (u[0]+1) * u[8] + (((10/9) * u[3] * u[2]) + ( (1/3) * (u[5]
           ↪  * u[1] + u[6] * u[2])))
133          ]
134      )
135    nl = (1.0 / Kn) * nl
136    nl = np.dot(Symm, nl)
137    nl = df.as_vector(nl)
138    return nl
```

Listing B.10: Snippet to define nonlinear production terms

Listing B.10: Define nonlinear production terms. This snippet defines a function, which calculates nonlinear production terms for the moment system. It initializes a zero array and then populates it with user-defined production term for 13-moment system as in the Equation III.136, incorporating nonlinear terms based on the system's state variables. The terms are then scaled by the Knudsen number and transformed using a symmetry matrix. Finally, the resulting vector is converted to a format compatible with FEniCS and integrated over the domain to contribute to the bilinear form of the variational problem.

Listing B.11: Apply Stabilisation. As discussed in Equations II.36 -II.40 in Section II.3, the snippet implements the GLS stabilisation.

Listing B.12: Create implicit variational form. This snippet constructs the implicit variational form for the moment system. It initializes both the left-hand side (LHS) and right-hand side (RHS) of the variational problem. The LHS is assigned the previously defined bilinear form 'a'. The RHS accumulates contributions from Listing B.8 , Lagrangian constraints and

```
139  df.dS = dS
140  h_msh = df.CellDiameter(mesh)
141  h_avg = ( h_msh("+") + h_msh("-") )/2.0
142  gls_const = input_file["STABILISATION_PARAMS"]["gls"]["const"]
143  gls_lhs =  gls_const * h_msh * df.inner(SA_x * df.Dx(u,0)+ SA_y *
     ↪  df.Dx(u,1) + SP_Coeff * u, SA_x * df.Dx(v,0) + SA_y * df.Dx(v,1) +
     ↪  SP_Coeff * v ) * df.dx
144  gls_rhs =  gls_const * h_msh * df.inner(Collision_nonlinearity(), SA_x *
     ↪  df.Dx(v,0) + SA_y * df.Dx(v,1)) * df.dx
```

Listing B.11: Snippet to apply stabilisation

nonlinear production terms computed by Listing B.10. The final implicit variational form 'F' is obtained by subtracting the RHS from the LHS.

```
145  lhs = 0.0
146  rhs = 0.0
147  if problem_type == "linear":
148      lhs = a + df.lhs(gls_lhs)
149      rhs = bc + df.rhs(gls_lhs)
150  elif problem_type == "nonlinear":
151      lhs = a
152      rhs = bc + df.inner(v,Collision_nonlinearity()) * df.dx + dd *
         ↪  df.Constant(1.0) * df.dx
153  else:
154      print("Undefined problem_type paramter is passed")
155      quit()
156  F = lhs - rhs
```

Listing B.12: Snippet to create implicit variational form

Listing B.13: Solvers for linear and nonlinear problems. This snippet outlines the process for solving the variational problem depending on the type specified by the user-input `problem_type`. For linear problems, it directly solves the linear system using the `df.solve` function. For nonlinear problems, it sets up and solves the nonlinear variational problem using a Newton method. It defines the Jacobian of the nonlinear form 'F', configures solver parameters (absolute and relative tolerances, maximum iterations, and relaxation parameter), and employs a nonlinear solver with the MUMPS linear solver. If no valid `problem_type` is provided, it prints an error message.

Listing B.14: Post-processing. The solution variable `u` returned by solver function in Listing B.13 is in $w_{\alpha\beta}^{(\bullet)}$. This snippet maps moment variables to macroscopic variables using linear transformation mapping. Furthermore, it outputs solution variables to `HDF5` file formats which can be used for further post-processing.

```
if problem_type == "linear":
  u_function = df.Function(MFS)
  df.solve(lhs == rhs, u_function, [])
  u = u_function
elif problem_type == "nonlinear":
  du_mixed = df.TrialFunction(MFS)
  Jac = df.derivative(F, u_mixed, du_mixed)
  abs_tol = input_file["SOLVER_PARAMS"]['newton_abs_tol']
  rel_tol = input_file["SOLVER_PARAMS"]['newton_rel_tol']
  max_itr = input_file["SOLVER_PARAMS"]['newton_max_itr']
  step_size = input_file["SOLVER_PARAMS"]['newton_relaxation_parameter']
  problem = df.NonlinearVariationalProblem(F, u_mixed, [], Jac)
  solver = df.NonlinearVariationalSolver(problem)
  solver.parameters ['newton_solver']['linear_solver'] = 'mumps'
  solver.parameters ['newton_solver']['absolute_tolerance'] = abs_tol
  solver.parameters ['newton_solver']['relative_tolerance'] = rel_tol
  solver.parameters ['newton_solver']['maximum_iterations'] = max_itr
  solver.parameters ['newton_solver']['relaxation_parameter'] =step_size
  solver.solve()
  u = u_mixed
else:
  print("Unknown problem type. Choose linear or nonlinear.")
```

Listing B.13: Snippet for solver

```
physical_variables = u.split(deepcopy=True)
def linear_transformation(sol_w):
  physical_variables[0].split()[0].vector()[:] =
  ↪  sol_w[0].split()[0].vector()[:]
  physical_variables[0].split()[1].vector()[:] = ref_vx +
  ↪  sol_w[0].split()[1].vector()[:] * math.sqrt(ref_theta)
  physical_variables[0].split()[2].vector()[:] = ref_vy +
  ↪  sol_w[0].split()[2].vector()[:] * math.sqrt(ref_theta)
  physical_variables[0].split()[3].vector()[:] = np.divide( ref_theta -
  ↪  sol_w[0].split()[3].vector()[:] * (2/3) * ref_theta, ref_R)
  return physical_variables

physical_variables = linear_transformation(w_variables)

def write_func(field_name, file_path, file_marker_list):
    xdmffile_u = df.XDMFFile(df.MPI.comm_world, '{0}/{1}_{2}.xdmf'
    ↪  .format(file_path, file_marker_list[0], file_marker_list[1]))
    xdmffile_u.write(field_name)
    xdmffile_u.close()

for i in range(input_file["GENERAL_PARAMS"]["number_of_moments"]):
    write_func( w_variables[i], "results_output_folder", [i,"w"])

print("Program terminated successfully")
```

Listing B.14: Snippet for post-processing

B.2. Sample YAML input file

```yaml
GENERAL_PARAMS:
  problem_type: 'nonlinear'
  moment_order: 'grad13w'
  number_of_moments: 9
  Kn: 0.0005
  ref_R: 208.1
  ref_rho: 1.646
  ref_theta: 298
  ref_vx: 0.0
  ref_vy: 0.0
MESH_PARAMS:
  mesh_list:
    - ../../../mesh/lid06.h5
STABILISATION_PARAMS:
  enable: True
  stab_type: gls
  cip:
    DELTA_T: 1.0
    DELTA_P: 0.01
    DELTA_U: 1.0
    ht: 3
    hp: 3
    hu: 3
  gls:
    h_power: 1
    const: 0.0001
SOLVER_PARAMS:
  newton_abs_tol: 0.0000000001
  newton_rel_tol: 0.00000000001
  newton_max_itr: 5000
  newton_relaxation_parameter: 0.9
BOUNDARY_PARAMS:
  chi: 1.0
  epsilon_w: 0.0000001
  bc:
    3000: #moving wall
      theta_w: 303
      u_t_w: -10.0
      u_n_w: 0.0
      p_w: 0.0
      eps_w: 0.000000000000001
      chi: 1.0
    3100: #stationary wall
      theta_w: 293
      u_t_w: 0.0
      u_n_w: 0.0
      p_w: 0.0
      eps_w: 0.000000000000001
      chi: 1.0
```

Listing B.15: Sample user-input `yaml` file to simulate nonisothermal lid-driven cavity flow

B.3. 13-Moment matrices

In this section we provide the definition of the matrices used in G13 program described in the Section B.1.

System matrices. The entries of the general system matrices $\boldsymbol{A}^{(x)}$ and $\boldsymbol{A}^{(y)}$ can be derived using Equation (III.60). In the 13-moment case, the following entries are obtained by writing Equation (III.128) in explicit x, y component forms. Since we are interested in studying the steady-state, we exclude the term with time derivative. Furthermore, we assumed the background velocity $v^{(\mathrm{ref})}$ zero. The remaining terms yield the following matrices:

$$\boldsymbol{A}^{(x)} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -\frac{2}{3} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & \frac{4}{3} & 0 & 0 & 0 & 0 & 0 & -\frac{8}{15} & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{2}{5} \\ 0 & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 & \frac{4}{15} & 0 \\ 0 & 0 & 0 & \frac{5}{3} & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix} \tag{B.1}$$

$$\boldsymbol{A}^{(y)} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -\frac{2}{3} & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 & \frac{4}{15} \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{2}{5} & 0 \\ 0 & 0 & \frac{4}{3} & 0 & 0 & 0 & 0 & 0 & -\frac{8}{15} \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{5}{3} & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \tag{B.2}$$

Linear collision matrix. As we remarked in Equation (III.136), we use the collisions terms obtained by solving the full Boltzmann collision integral. The following matrix $\boldsymbol{P}$ is the result of the linearised Boltzmann collision operator.

$$\text{P\_Coeff} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{3} \end{bmatrix} \tag{B.3}$$

Symmetrisation matrix. The 13-moment symmetriser is given by Equation (III.139).

$$\text{Symm} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{2}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{5} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{5} \end{bmatrix} \tag{B.4}$$

Transport sub-matrix. The following $\mathbb{A}$ matrix can be computed from Equation (III.73). The 13-moment sub-matrix is given in Equation (III.147).

$$\mathbb{A} = \begin{bmatrix} 1 & 0 & -\frac{2}{3} & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -\frac{2}{5} \\ 0 & 0 & \frac{2}{3} & -\frac{2}{5} & 0 & 0 \end{bmatrix} \tag{B.5}$$

Odd and Even matrices. The following matrices can be constructed from the odd and even variables at the boundary, defined in Equations (III.145) and (III.146).

$$\text{P\_odd} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{B.6}$$

$$\text{P\_even} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{B.7}$$

Transformation matrix. The 13-moment transformation matrix is described in Equations (III.140), (III.141), and (III.142). Please note that the `nv` in the following refers to the symbol used in the Python program to denote the normal vector in Dolfin. Refer the previous section

for more details.

$$\text{T\_n} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \text{nv}[0] & \text{nv}[1] & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\text{nv}[1] & \text{nv}[0] & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{nv}[0]^2 & 2\text{nv}[0] * \text{nv}[1] & \text{nv}[1]^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\text{nv}[0] * \text{nv}[1] & \text{nv}[0]^2 - \text{nv}[1]^2 & \text{nv}[0] * \text{nv}[1] & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{nv}[1]^2 & -2\text{nv}[0] * \text{nv}[1] & \text{nv}[0]^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \text{nv}[0] & \text{nv}[1] \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\text{nv}[1] & \text{nv}[0] \end{bmatrix} \tag{B.8}$$

Onsager coupling matrix. The 2D 13-moment Onsager coupling matrix in Equation (III.149) reads

$$\text{L\_matrix} = \begin{bmatrix} 1 & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{9}{4} \end{bmatrix}. \tag{B.9}$$

However, for numerical stability reasons we eliminate the density coefficient from the boundary conditions by assuming $u_n = 0$. That means no inflow at the boundary. This can be achieved by multiplying the following matrix

$$\boldsymbol{R} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\frac{1}{2} & 0 & 0 & 1 \end{bmatrix} \tag{B.10}$$

with Equation (III.114), we get

$$w^{(\boldsymbol{n},o)}|_{\text{BC}} = -\frac{2\chi}{2-\chi}\boldsymbol{R}\sum_{\beta\in I_M^{(\boldsymbol{n},e)}}\left(L_{\alpha\beta}\mathbb{A}_{\alpha\beta}w_\beta^{(\boldsymbol{n},e)}\right) + \boldsymbol{R}g_\alpha^{(\text{in},\boldsymbol{n})}. \tag{B.11}$$

From (B.11) we get the modified coupling matrix (in 2D)

$$\text{L\_matrix}^{(\text{simplified})} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \tag{B.12}$$

and the modified 2D inhomogeneity becomes

$$g^{(\text{simplified})} = \frac{2\chi}{2-\chi}\begin{bmatrix} 0 \\ w_{t1} \\ \frac{4}{3}w^{(1),\text{in}} \end{bmatrix}. \tag{B.13}$$

Since we are inverting the boundary coupling matrix in our weak formulation, we cannot have a zero on the diagonal of the $\boldsymbol{L}$ matrix. We adapt the following work around in our implementation. We need to add small ϵ in the first entry to numerical error. We subtract the same ϵ from the inhomogeneity vector.

The one-to-one non-linear mappings between moment coefficients and macroscopic variables are discussed in Equations (III.123) - (III.127), Below we consider only the linear mappings

$$w^{(0)} = \frac{\rho}{\rho^{(\text{ref})}} \tag{B.14}$$

$$w_i^{(0)} = \frac{\Delta v_i}{\sqrt{\theta^{(\text{ref})}}}, \qquad i = 1, 2 \tag{B.15}$$

$$w^{(1)} = -\frac{3}{2}\frac{\Delta\theta}{\theta^{(\text{ref})}} \tag{B.16}$$

By substituting the above mappings in Equation (B.13), we obtain the implementation-ready inhomogeneity vector

$$g^{(\text{simplified})} = \frac{2\chi}{2-\chi}\begin{bmatrix} 0 \\ \frac{\Delta v_t}{\sqrt{\theta^{(\text{ref})}}} \\ -2\frac{\Delta\theta}{\theta^{(\text{ref})}} \end{bmatrix}. \tag{B.17}$$

Bibliography

[1] Zhenning Cai, Yuwei Fan, and Yanli Wang. "Burnett spectral method for the spatially homogeneous Boltzmann equation." In: *Computers and Fluids* 200 (2020), p. 104456. ISSN: 0045-7930. DOI: https://doi.org/10.1016/j.compfluid.2020.104456.

[2] Irene M. Gamba and Sergej Rjasanow. "Galerkin–Petrov approach for the Boltzmann equation." In: *Journal of Computational Physics* 366 (2018), pp. 341–365. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.04.017.

[3] M. TORRILHON and H. STRUCHTRUP. "Regularized 13-moment equations: shock structure calculations and comparison to Burnett models." In: *Journal of Fluid Mechanics* 513 (2004), pp. 171–198. DOI: 10.1017/S0022112004009917.

[4] Henning Struchtrup and Manuel Torrilhon. "Higher-order effects in rarefied channel flows." In: *Phys. Rev. E* 78 (4 Oct. 2008), p. 046301. DOI: 10.1103/PhysRevE.78.046301. URL: https://link.aps.org/doi/10.1103/PhysRevE.78.046301.

[5] Henning Struchtrup et al. "Evaporation boundary conditions for the R13 equations of rarefied gas dynamics." In: *Physics of Fluids* 29.9 (Sept. 2017), p. 092004. DOI: 10.1063/1.4989570.

[6] M. Yu. Timokhin et al. "Different variants of R13 moment equations applied to the shock-wave structure." In: *Physics of Fluids* 29.3 (Mar. 2017), p. 037105. DOI: 10.1063/1.4977978.

[7] H. Struchtrup. *Macroscopic Transport Equations for Rarefied Gas Flows*. Springer, 2005.

[8] Manuel Torrilhon. "Modeling Nonequilibrium Gas Flow Based on Moment Equations." In: *Annual Review of Fluid Mechanics* 48.Volume 48, 2016 (2016), pp. 429–458. ISSN: 1545-4479. DOI: https://doi.org/10.1146/annurev-fluid-122414-034259.

[9] Lambert Theisen and Manuel Torrilhon. "fenicsR13: A Tensorial Mixed Finite Element Solver for the Linear R13 Equations Using the FEniCS Computing Platform." In: *ACM Trans. Math. Softw.* 47.2 (Apr. 2021). ISSN: 0098-3500. DOI: 10.1145/3442378.

[10] Manuel Torrilhon. "Slow gas microflow past a sphere: Analytical solution based on moment equations." In: *Physics of Fluids* 22.7 (July 2010), p. 072001. ISSN: 1070-6631. DOI: 10.1063/1.3453707.

[11] Jonas Bünger, Neeraj Sarna, and Manuel Torrilhon. *Stable Boundary Conditions and Discretization for PN Equations*. 2021. arXiv: 2004.02497 [math.NA]. URL: https://arxiv.org/abs/2004.02497.

[12] Henning Struchtrup and Manuel Torrilhon. "H theorem, regularization, and boundary conditions for linearized 13 moment equations." In: *Physical review letters* 99 1 (2007). URL: https://api.semanticscholar.org/CorpusID:19648097.

[13] Neeraj Sarna and Manuel Torrilhon. “Entropy stable Hermite approximation of the linearised Boltzmann equation for inflow and outflow boundaries.” In: *Journal of Computational Physics* 369 (2018), pp. 16–44. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.04.050.

[14] Armin Westerkamp and Manuel Torrilhon. “Stabilization techniques in finite element discretizations for moment approximations.” In: *AIP Conference Proceedings* 1628.1 (Dec. 2014), pp. 1016–1023. ISSN: 0094-243X. DOI: 10.1063/1.4902705.

[15] Lambert Theisen. *FEniCSR13*. Accessed: 2024-08-29. 2023. URL: https://github.com/lambooo/fenicsr13.

[16] E. Christhuraj. 2024. URL: https://github.com/19ec94/r13crookes.

[17] Jane Wess. “Crookes’s radiometers: a train of thought manifest.” In: *Notes and Records of the Royal Society* 64.4 (2010), pp. 457–470. DOI: 10.1098/rsnr.2010.0034.

[18] J. Clerk Maxwell. “On Stresses in Rarefied Gases Arising from Inequalities of Temperature.” In: *Proceedings of the Royal Society of London* 27 (1878), pp. 304–308. ISSN: 03701662. URL: http://www.jstor.org/stable/113680.

[19] Andrew Ketsdever et al. “Radiometric phenomena: From the 19th to the 21st century.” In: *Vacuum* 86.11 (2012). Vacuum Gas Dynamics: Theory, experiments and practical applications, pp. 1644–1662. ISSN: 0042-207X. DOI: https://doi.org/10.1016/j.vacuum.2012.02.006. URL: https://www.sciencedirect.com/science/article/pii/S0042207X12000723.

[20] Yu. Anikin. “Numerical study of radiometric forces via the direct solution of the Boltzmann kinetic equation.” In: *Computational Mathematics and Mathematical Physics* 51 (2011). DOI: 10.1134/S0965542511070049.

[21] Li-Hsin Han et al. “Light-Powered Micromotor: Design, Fabrication, and Mathematical Modeling.” In: *Microelectromechanical Systems, Journal of* 20 (May 2011), pp. 487–496. DOI: 10.1109/JMEMS.2011.2105249.

[22] Songze Chen, Kun Xu, and Cunbiao Lee. “The dynamic mechanism of a moving Crookes radiometer.” In: *Physics of Fluids* 24.11 (Nov. 2012). ISSN: 1070-6631. URL: https://doi.org/10.1063/1.4765353.

[23] Natalia E. Gimelshein et al. “Shear Force in Radiometric Flows.” In: *AIP Conference Proceedings* 1333.1 (May 2011), pp. 661–666. ISSN: 0094-243X. URL: https://doi.org/10.1063/1.3562722.

[24] Jonas Bünger et al. “Structured derivation of moment equations and stable boundary conditions with an introduction to symmetric, trace-free tensors.” In: *Kinetic and Related Models* 16.3 (2023), pp. 458–494. ISSN: 1937-5093. DOI: 10.3934/krm.2022035.

[25] I. Mueller and T. Ruggeri. *Rational extended thermodynamics*. Springer Tracts in Natural Philosophy. Springer New York, 2013. ISBN: 9781461222101. URL: https://books.google.de/books?id=ZePSBgAAQBAJ.

[26] Harold Grad. “On the kinetic theory of rarefied gases.” In: *Communications on Pure and Applied Mathematics* 2.4 (1949), pp. 331–407. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160020403.

[27] Jon Applequist. “Maxwell–Cartesian spherical harmonics in multipole potentials and atomic orbitals.” In: *Theoretical Chemistry Accounts* 107 (Feb. 2002), pp. 103–115. DOI: 10.1007/s00214-001-0301-2.

[28] D. C. Pack. “Theory and Application of the Boltzmann Equation. By CARLO CERCIGNANI.” In: *Journal of Fluid Mechanics* 81.4 (1977), pp. 793–794. DOI: 10.1017/S0022112077212353.

[29] Sarna Neeraj and Torrilhon Manuel. “On Stable Wall Boundary Conditions for the Hermite Discretization of the Linearised Boltzmann Equation.” In: *Journal of Statistical Physics* 170 (2018). URL: https://doi.org/10.1007/s10955-017-1910-z.

[30] Andrea Hanke and Manuel Torrilhon. “Representation Theory Based Algorithm to Compute Boltzmann’s Bilinear Collision Operator in the Irreducible Spectral Burnett Ansatz Efficiently.” In: *Journal of scientific computing* 95.3 (2023), p. 78. ISSN: 0885-7474. URL: https://publications.rwth-aachen.de/record/958502.

[31] Zhenning Cai and Manuel Torrilhon. “On the Holway-Weiss debate: Convergence of the Grad-moment-expansion in kinetic gas theory.” In: *Physics of Fluids* 31.12 (Dec. 2019), p. 126105. ISSN: 1070-6631. URL: https://doi.org/10.1063/1.5127114.

[32] Edilbert Christhuraj and Manuel Torrilhon. “Generic moment systems and FEM-based numerical solutions for the Boltzmann equation.” In: *AIP Conference Proceedings* 2996.1 (Feb. 2024), p. 070001. ISSN: 0094-243X. DOI: 10.1063/5.0187559. URL: https://doi.org/10.1063/5.0187559.

[33] J. Scroggs et al. e-print arXiv:2102.11901v5. 2022.

[34] E. Christhuraj. 2022. URL: https://gitlab.com/19ec94/f2me.

[35] Armin Westerkamp and Manuel Torrilhon. “Slow rarefied gas flow past a cylinder: Analytical solution in comparison to the sphere.” In: *AIP Conference Proceedings* 1501.1 (Nov. 2012), pp. 207–214. ISSN: 0094-243X. URL: https://doi.org/10.1063/1.4769505.

[36] U. Ghia, K. Ghia, and C. Shin. In: *Journal of Computational Physics* 48 (1982), pp. 387–411.

[37] Vladimir Djordjić, Milana Pavić-Čolić, and Manuel Torrilhon. “14- and 17-moment systems for polytropic gases - Comparison regarding transport coefficients.” In: *AIP Conference Proceedings* 2996.1 (Feb. 2024), p. 040011. ISSN: 0094-243X. URL: https://doi.org/10.1063/5.0187771.

[38] Bo Lin et al. “Time-dependent regularised 13-moment equations with Onsager boundary conditions in the linear regime.” In: *Journal of Fluid Mechanics* 1009 (2025), A60. DOI: 10.1017/jfm.2025.215.