TECHNISCHE
UNIVERSITÄT
DARMSTADT

Mesh and model adaptation for hyperbolic balance laws

Vom Fachbereich Mathematik

der Technischen Universität Darmstadt

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

genehmigte Dissertation

von

Hrishikesh Joshi, M.Sc.

aus Pune, Indien

Referent : Prof. Dr. Jan Giesselmann
Koreferent : Prof. Dr. Siegfried Müller
Tag der Einreichung: 21.10.2022
Tag der mündlichen Prüfung: 09.12.2022

Darmstadt
D17

Mesh and model adaptation for hyperbolic balance laws

Accepted doctoral thesis by Hrishikesh Joshi

Referent: Prof. Dr. Jan Giesselmann
Koreferent: Prof. Dr. Siegfried Müller

Date of submission: 21.10.2022
Date of thesis defense: 09.12.2022

Darmstadt – D17

Bitte zitieren Sie dieses Dokument als:
URN: urn:nbn:de:tuda-tuprints-230482
URL: https://tuprints.ulb.tu-darmstadt.de/23048

Dieses Dokument wird bereitgestellt von tuprints,
E-Publishing-Service der TU Darmstadt
http://tuprints.ulb.tu-darmstadt.de
tuprints@ulb.tu-darmstadt.de

To my parents
and my brother
for their love and support

Acknowledgments

I would like to thank and express my gratitude to my supervisor Prof. Dr. Jan Giesselmann for his invaluable advice, feedback and guidance and for providing support and encouragement during the course of the thesis. I also thank Prof. Dr. Siegfried Müller for his assistance and for the countless fruitful discussions.

I extend my gratitude to my colleagues Aleksey, Bogdan, Seshadri, Seva, Neelabja for making it a pleasure to be at work. I also greatly appreciate the assistance provided by Aleksey for software development.

Finally, I would like to thank my parents and my brother Aniket for the unwavering support and love.

Abstract

The objective of mesh and model adaptation is to locally choose the mesh and the model to be employed from a model hierarchy in order to reduce the computational resources needed. We do this by balancing the two sources of error, namely discretization and modelling error. In this work, we devise a mesh and model adaptation strategy for a class of model hierarchies consisting of two levels of model complexity. In particular, the fine model, also referred to as the complex system, consists of a system of hyperbolic balance laws with stiff reaction terms and the coarse model, also referred to as the simple system, consists of a system of hyperbolic conservation laws. The governing equations of the simple system are derived by making a simplifying assumption that the system is in equilibrium, i.e. the speed of the reaction is infinitely fast. Furthermore, the complex system is assumed to have an entropy structure, i.e. it is assumed to be equipped with a strictly convex entropy and entropy flux. The structure of the model hierarchy allows us to show that the simple system is analogously equipped with a strictly convex entropy and entropy flux. The relative entropy stability framework is employed to derive a posteriori error estimates with identifiable contributions of discretization and modelling error estimates. Furthermore, since the use of two different models in the computational domain gives rise to cell boundaries across which the model employed differs, we propose a coupling to be employed at these cell boundaries. In addition, mesh and model coarsening distances are defined, which provide complementary information to the defined error indicators. The defined error indicators and coarsening distances are employed to propose a mesh and model adaptation strategy. The efficacy of the mesh and model adaptation strategy is demonstrated by conducting simulations for chemically reacting fluid mixtures in one space dimension.

Zusammenfassung

Das Ziel der Gitter- und Modelladaption ist die lokale Auswahl des zu verwendenden Gitters und Modells aus einer Modellhierarchie auszuwählen, um die benötigten Rechenressourcen zu reduzieren. Wir tun dies, indem wir die beiden Fehlerquellen ausgleichen, nämlich Diskretisierungs- und Modellierungsfehler. In dieser Arbeit entwickeln wir eine Strategie zur Gitter- und Modelladaption für eine Klasse von Modellhierarchien, die aus zwei Ebenen der Modellkomplexität bestehen. Insbesondere besteht das feine Modell, auch als komplexes System bezeichnet, aus einem System hyperbolischer Bilanzgesetzen mit steifen Reaktionstermen, und das grobe Modell, auch als einfaches System bezeichnet, besteht aus einem System hyperbolischer Erhaltungsgesetze. Die Gleichungen des einfachen Systems werden unter der vereinfachenden Annahme hergeleitet, dass sich das System im Gleichgewicht befindet, d.h. die Reaktionsgeschwindigkeit ist unendlich schnell. Für das komplexe System wird eine Entropiestruktur angenommen, d.h. es wird angenommen, dass es mit einer streng konvexen Entropie und einem Entropiefluss ausgestattet ist. Die Struktur der Modellhierarchie ermöglicht es uns zu zeigen, dass das einfache System analog dazu mit einer streng konvexen Entropie und einem Entropiefluss ausgestattet ist. Der Rahmen der relativen Entropie-Stabilität wird zur Herleitung von a posteriori-Fehler Schätzungen mit identifizierbaren Beiträgen von Diskretisierungs- und Modellierungsfehlern abzuleiten. Da die Verwendung von zwei verschiedenen Modellen im Rechengebiet zu Zellgrenzen führt, an denen sich das verwendete Modell unterscheidet, schlagen wir eine Kopplung vor, die an diesen Zellgrenzen eingesetzt wird. Darüber hinaus werden Gitter- und Modellvergröberungsabstände definiert, die ergänzende Informationen zu den definierten Fehlerindikatoren liefern. Die definierten Fehlerindikatoren und Vergröberungsabstände werden verwendet, um eine Gitter- und Modelladaptive Strategie vorzuschlagen. Die Wirkung der Gitter und Modellanpassungsstrategie wird durch die Durchführung von Simulationen für chemisch reagierende Flüssigkeitsgemische in einer Raumdimension demonstriert.

Contents

Chapter 1

Introduction

Hyperbolic balance laws are an important class of equations, which have an extensive scope of applications arising in continuum physics, cf. [17],[60],[87], [93]. The applications, in numerous cases, are characterised by a wide range of length and time scales. As a result, simulating these by numerical methods can be computationally expensive. The high computational expenses, in the cases where the source terms are stiff and non-linear, is due to the necessity of employing an implicit time stepping method, cf. [9], [76],[77], or small time steps in the case of explicit time stepping, cf. [22],[49]. In some cases, the system of equations can be simplified given some constraints hold, leading to a system of conservation laws. This gives rise to a model hierarchy consisting of two levels of complexity; a system of hyperbolic balance laws and a system of hyperbolic conservation laws. Throughout the thesis, we refer to the system of hyperbolic balance laws as the complex system and the system of hyperbolic conservation laws as the simple system. Furthermore, models of different levels of complexity are referred to as models of different refinement levels. Employing the simple model is also colloquially referred to as employing a coarse model. Additionally, it is assumed that the complex system is endowed with a strictly convex entropy and entropy flux. This assumption constraints the system of balance laws to be compatible with the second law of thermodynamics. The assumed constraints employed to simplify the system of balance laws allow us to show that the model hierarchy has certain geometric and entropy properties, consequently allowing us to show that the simple system inherits a strictly convex entropy and entropy flux satisfying the standard compatibility conditions, cf. [26],[47].

Given a model hierarchy as described above, one way to reduce the computational resources required for the numerical simulations is to a priori decompose the computational domain and employ the complex system in a sub-domain and the simple system everywhere else, see for example [7], [8] and [23]. This approach requires some a priori knowledge about the problem at hand in order to determine the decomposition. Even in the case that the a priori decomposition is done based on some knowledge about the problem, the numerical accuracy of the simulation might be sensitive to the location of the interfaces, i.e. the cell boundaries across which the models employed differ. Furthermore, in transient simulations, the time evolution may be such that the sub-domains where the simplifying assumption holds may change. Hence, a static decomposition of the computational domain will not always be appropriate.

Another approach is employing a so called dynamic heterogeneous model adaptation technique, where the model refinement is determined locally in space and time with the objective of keeping the reduction in the accuracy of the resulting model adaptive simulation small. Devising a dynamic heterogeneous model adaptation technique poses two challenges. The first is estimating the decrease in accuracy due to locally employing a coarse model. The second being the fact that employing two different models in a decomposed computational domain gives rise to interfaces, consequently requiring coupling to be specified in the form of numerical fluxes at the interfaces. The coupling employed and the location of the interfaces should be such that the presence of an interface, which is inherently artificial, does not produce artefacts and significantly affect the accuracy of the simulations. It would also be appropriate for the model adaptation to be combined with mesh adaptation in order to reduce the computational resources needed. In this thesis, we propose and numerically test a dynamic mesh and model adaptation approach for the model hierarchies of the general form stated above.

To carry out a mesh and model adaptive simulation, local indicators are needed to make an informed decision about the mesh and model refinement to be employed. Model adaptation strategies have been previously proposed employing different mathematical approaches such as dual weighted residuals, cf. [5],[11], and Chapman-Enskog expansion, cf. [65],[66], albeit the two approaches have some limitations. The dual weighted residuals approach consists of solving a dual problem to measure the impact on a target functional depending on the model being employed. However, the dual problem may become ill-posed for non-linear hyperbolic conservation laws, for example in the case of Euler equations [54]. In the approach based on a Chapman-Enskog expansion, the first order term in the expansion is employed as a numerical indicator. The approach is heuristic and does not provide error estimates in general for the case of systems of balance laws. However, in the special case of decoupled scalar equations, error estimates can be derived [13]. Mesh adaptation is also an extensively researched field. The mesh adaptive strategies fall under three major categories [38]. One approach is to employ a posteriori error estimates for mesh adaptation, cf. [42]. Another approach is where a local indicator, for example gradient or curvature of the solution, is employed to locally coarsen or refine the mesh, cf. [3],[83]. Finally, perturbation methods such as multiresolution based Runge-Kutta Discontinuous Galerkin method proposed by Müller et al., where the central idea is to preserve the accuracy of a mesh adaptive solution in comparison to a reference simulation, i.e. simulation on a uniformly refined mesh, cf. [37],[38].

We propose a mesh and model refinement strategy based on a posteriori error analysis by employing the relative entropy stability framework, cf. [26],[94], for the model hierarchy at hand. This stability framework requires one of the solutions being compared to be Lipschitz continuous [26]. Since the numerical solution will not generally have the necessary regularity, a reconstruction of the numerical solution needs to be computed. The reconstruction method proposed in [40], in the context of a posteriori error analysis of Discontinuous Galerkin methods for hyperbolic conservation laws, is employed. Giesselmann et. al. proposed a model adaptation strategy employing the relative entropy stability framework for model hierarchy consisting of the Euler equations and Navier-

Stokes Fourier equations in [44]. In the same spirit, we derive error estimates and propose a mesh and model adaptation strategy for model hierarchies consisting of hyperbolic balance and conservation laws.

The coupling that should be employed depends on the application and the objective of the numerical simulation. It is a well studied area of research in the field of hyperbolic conservation laws and a wide range of sophisticated methods have been proposed, cf. [1],[12], [48],[88]. We propose a simple coupling between the complex and the simple system. The coupling is expected to work reasonably well given the placement of the interfaces in space is such that the numerical solution in the cells near the interfaces satisfies some constraints. Greedier placements of the interface gives rise to artefacts producing significant errors and is detrimental to the accuracy of the resulting model adaptive simulation and counterproductive to the objective of model adaptation.

The a posteriori error analysis is independent of the numerical method employed and provides an upper bound for the difference between the mesh and model adaptive solution and the exact solution to the complex system. The error analysis is carried out such that the error estimate consists of identifiable contributions of the two sources of error; namely discretization and modeling errors. The discretization and modeling error indicators can then be employed locally for mesh and model adaptation. We employ explicit strong stability preserving Runge-Kutta Discontinuous Galerkin methods for the numerical simulations, cf. [22], [56], in particular multiresolution based Discontinuous Galerkin method proposed by Müller [37] et. al. is employed.

A pertinent problem which fits into the framework described above is that of chemically reacting flows. It has a broad range of applications in a large number of engineering and scientific fields like combustion [4],[59], atmospheric physics [86] and electro-chemistry [31] to name a few. Chemically reacting flows can be modeled by employing multi-component Euler equations with source terms [10], [74]. Since the problem is grounded in thermodynamics [10], [27], the notion of mathematical entropy and entropy flux follows from the physical entropy and entropy flux. The mesh and model adaptation strategy proposed is tested and its efficacy is demonstrated for chemically reacting flows in one space dimension. Note that, we conduct numerical experiments to demonstrate the effectiveness of the mesh and model adaptive strategy in terms of understanding if the strategy coarsens the mesh and the model in the regions where it would be suitable to do so. We do not quantitatively evaluate the reduction in computational resources since it would necessitate an optimal implementation and is out of the scope of the thesis.

Outline

The outline of the thesis is as follows:

Chapter 2: In this chapter, we abstractly describe the model hierarchy under consideration and specify assumptions that are made on its structure. In particular, we assume that the complex system possesses a compatible entropy structure and that the systems have certain geometric properties. This enables us to derive the error estimates.

Chapter 3: In this chapter, the numerical methods employed to carry out the mesh and model adaptive simulations are described. The Runge-Kutta Discontinuous Galerkin method is briefly presented followed by a procedure to obtain a Lipschitz continuous reconstruction from the Discontinuous Galerkin solution. A mesh and model adaptation strategy is proposed next, where the structure of the error estimates and properties of the reconstruction are taken into account.

Chapter 4: In this chapter, the modeling of chemically reacting flows is discussed. Closures for the equation of state and chemical reactions are provided to ensure a complete description of chemically reacting fluid mixtures. The entropy structure provided by the second law of thermodynamics is described and the compatibility with the abstract framework presented in Chapter 1 is assessed.

Chapter 5: In this chapter, the numerical simulations conducted to evaluate the efficacy of the mesh and model adaptation strategy and coupling are presented and their results are discussed. The test case of disassociation of molecular oxygen into atomic oxygen with nitrogen acting as a catalyst is considered.

Chapter 6: This chapter concludes the thesis, where we provide a brief outlook.

Research funding

Part of this research was funded from 2017-2020 by the German research Foundation (DFG) via the project DFG Gi 1131/1-1: *Dynamical, spatially heterogeneous model adaptation in compressible flows.*

Chapter 2

A posteriori error analysis

In this chapter, the a posteriori error analysis is presented.

Outline

The outline of this chapter is as follows

Section 2.1: In this section, the general ideas behind mesh and model adaptation are discussed.

Section 2.2: The model hierarchy and the governing equations of the models in the model hierarchy, for which we propose a mesh and model adaptation approach, is discussed in Section 2.2.

Section 2.3: In this section, some noteworthy geometric properties, which are to be employed in the error analysis, are derived.

Section 2.4: In this section, a review of the previous research pertaining to the relative entropy framework is carried out and the notions of relative entropy and relative entropy flux are introduced.

Section 2.5: In this section, equipped with the necessary prerequisites, the a posteriori error estimates are derived.

2.1 Ideas behind mesh and model adaptation

Numerical simulations enable study of phenomena arising in a wide range of fields like physics, chemistry, biology and economics to name a few. Ideally, numerical simulations would require adequately small amount of computational resources. Unfortunately, due to the inherent complexity of the phenomena arising in the real world, the phenomena as well need to be modeled with a high degree of complexity. However, in some cases it may be possible to simplify the modeling given suitable conditions are met. This leads to a model hierarchy, where the same phenomena can be described by employing varying levels of complexity. As a result, the computational resources required to carry out the

simulations depend on the complexity of the model chosen. The accuracy of the resulting numerical simulations also depends on the extent to which the simplifying assumptions are met.

The idea behind model adaptation is to decompose the computational domain and locally choose a model from the model hierarchy with the required level of complexity. Note that there is no universal definition of model adaptation and has been interpreted in various ways. First the approach we take is discussed, followed by giving examples of other methods of model adaptation. In this work, we deal with systems of hyperbolic balance and conservation laws, where the numerical solution depends on time and space. Initially, the finest model is employed in the entire computational domain. Thereafter, after each time step the computational domain is spatially decomposed into sub-domains and a model from the model hierarchy with a suitable level of complexity is chosen. The decision about which model to employ is made based on a mesh and model adaptation strategy employing the error estimates we derive. We refer to this model adaptation approach as heterogeneous model adaptation. We also refer to the resulting system of equations, where different models are employed in the sub-domains as the model adaptive system. This model adaptation approach has been visualized in Figure 2.1 for a model hierarchy consisting of two models: a fine model and a coarse model.

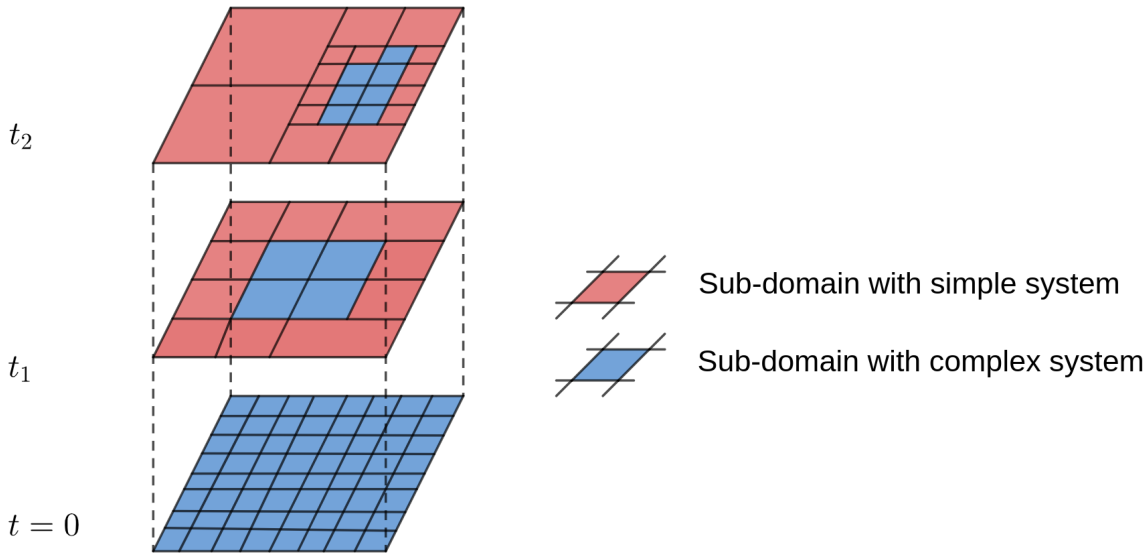

Figure 2.1: Mesh and model adaptation

Model adaptation has been interpreted in other ways. In the work of Munz et. al. [7], [95], the simulation of aeroacoustic problems was accelerated by employing a priori decomposed computational domain and employing either the Navier-Stokes equations, the linearized Euler equations or the far field wave equations in the sub-domains. In the work of Sarna and Torrilhon [92], a hierarchical solution method for the Boltzmann equations was proposed, where successive simulations are carried out by globally employing a finer model in each subsequent simulation from a hierarchy of models. In the the work of Godlewski

et al. [65] on hyperbolic systems with relaxation, model adaptation is interpreted in the same manner as we do, where after each time step the computational domain is subdivided in subdomains based on an error indicator. The central issue of model adaptation is taking into account the mathematical characteristics of the system and the underlying physics, in order to derive error estimates that allow model adaptation strategy without introducing significant errors. This chapter deals with deriving such error estimates for the model hierarchy in consideration.

Furthermore, the numerical methods employed to simulate the model can be adapted to minimize the computational resources needed, in particular by employing a mesh adaptation approach. The objective is to distribute the degrees of freedom with the intent of reducing the computational costs without significantly affecting the accuracy of the numerical solution. It is well known that solutions to hyperbolic conservation laws can be extremely heterogeneous, with the solution being discontinuous in some region to having large gradients in others or being relatively smooth. The degrees of freedom need to be heterogeneously distributed in the computational domain so that huge computational expenses are not incurred. This can be achieved by employing a highly refined grid in regions of strong gradients and discontinuities and a coarser grid in the smooth regions.

Altogether using a heterogeneous mesh and model adaptation approach, where the necessary level of mesh and model refinement is chosen locally in time and space has the potential to lead to significant savings in computational costs.

2.2 Model hierarchy

In this section, we introduce the abstract framework, i.e the governing equations of the models in the model hierarchy. We briefly discuss the difference in the physics for the models in the model hierarchy. The error estimates we derive hold for systems that fulfill the abstract structure discussed in this section.

The objective is to model the time evolution of quantities $\mathbf{U} \in \mathbb{R}^N$, which we refer to, with some abuse of terminology, as quantities of interest. The model hierarchy we investigate consists of two levels of complexity, a fine model and a coarse model. The system of governing equations of the fine model is referred to as the complex system and the system of governing equations of the coarse model is referred to as the simple system. First we discuss the governing equations of the fine model and then turn to the governing equations of the coarse model, which are derived by making some simplifying assumptions.

The complex system, consists of a system of balance laws given by

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = \frac{1}{\varepsilon}\mathbf{R}(\mathbf{U}), \ \mathbf{U} : \mathbb{T} \times \mathbb{R}^+ \to \mathbb{R}^N, \tag{2.1}$$

where $\varepsilon > 0$, $\mathbf{F} : \mathbb{R}^N \to \mathbb{R}^N$ is a given flux function and $\mathbf{R} : \mathbb{R}^N \to \mathbb{R}^N$ is a given source term. The computational domain $\mathbb{T}$ is assumed to be the one dimensional torus, i.e. we assume periodic boundary conditions. Note that in (2.1) $\mathbb{R}^+ := \{x \in \mathbb{R} : x > 0\}$.

The complex system accounts for two processes; convection dynamics and reaction. In this context, by process we mean a mechanism which leads to a change of state. The reaction drives the evolution towards a state of equilibrium, i.e. states for which the source term vanishes. The manifold of equilibrium states, is called the equilibrium manifold and is denoted by $\mathcal{M}$. Note that, once equilibrium is reached, convection may act in the direction normal to the equilibrium manifold, therefore pushing the states out of equilibrium. However, this will immediately be counteracted by the reaction. In the complex system, the convection dynamics is accounted for by the flux function and the reaction by the source term, in which $\frac{1}{\varepsilon}$ characterises the speed of the reaction. A few examples of reaction and its equilibrium in the field of fluid mechanics are: chemical reactions and chemical equilibrium [45], phase change and phase equilibrium [82],[90]. In the model hierarchy we look at, other processes such as viscosity are not accounted for.

At equilibrium, the source term vanishes and the modeling can be simplified. The governing equations of the coarse model, i.e. the simple system are obtained as follows. Projecting the complex system, employing a projection matrix $\mathbb{P} : \mathbb{R}^N \to \mathbb{R}^n$ such that

$$\mathbb{P}\mathbf{R}(\mathbf{U}) = 0, \ \text{rank}\ (\mathbb{P}) = n \tag{2.2}$$

and defining $\mathbf{u} := \mathbb{P}\mathbf{U} : \mathbb{T} \times \mathbb{R}^+ \to \mathbb{R}^n$, we get

$$\partial_t \mathbf{u} + \partial_x \mathbb{P}\mathbf{F}(\mathbf{U}) = 0.$$

Note that the projection matrix $\mathbb{P}$ is assumed to be constant.

The equilibrium states are parameterized by the conserved quantities $\mathbf{u} \in \mathbb{R}^n$ by employing a function M such that

$$M : \mathbb{R}^n \to \mathbb{R}^N, \ \mathbf{R}(M(\mathbf{u})) = 0, \ \mathbb{P}M(\mathbf{u}) = \mathbf{u}. \tag{2.3}$$

We refer to the function M as the Maxwellian.

The gradient of the source term on the equilibrium manifold is assumed to satisfy the following non-degeneracy conditions

$$\dim \text{Ker}\ (\mathrm{D}\,\mathbf{R}(M(\mathbf{u}))) = n, \ \dim \text{Im}\ (\mathrm{D}\,\mathbf{R}(M(\mathbf{u}))) = N - n. \tag{2.4}$$

Formally, in the limit $\varepsilon \to 0$, the conserved quantities $\mathbf{u}$ satisfy the system of conservation laws given by

$$\partial_t \mathbf{u} + \partial_x \mathbf{g}(\mathbf{u}) = 0, \tag{2.5}$$

where $\mathbf{g}(\mathbf{u}) := \mathbb{P}\mathbf{F}(M(\mathbf{u})) : \mathbb{R}^n \to \mathbb{R}^n$.

System (2.5) is referred to as the simple system.

It can be observed that the processes accounted for by the complex system can be modeled by quantities in $\mathbb{R}^N$. On the other hand, the simple system, makes the simplifying

assumption that the evolution of the quantities can be adequately described by evolution on the equilibrium manifold, i.e. the reaction is infinitely fast. As a result the dynamics can be described by quantities in $\mathbb{R}^n$. This results in a model hierarchy of two levels of complexity where the governing equations of the coarser model form a system of equations of a smaller size than that of the fine model.

2.2.1 Entropy structure

Mathematically, the notion of entropy has been employed in the field of hyperbolic conservation laws to show uniqueness of solutions and also to derive admissibility criteria. Extensive research has been conducted for hyperbolic conservation laws in this direction. The interested reader is referred to standard texts such as [26],[47].

For systems of the form (2.1), the notion of entropy was employed by Dafermos in [25] to show stability. Here, stability is interpreted as continuous dependence of the solution on the initial state. It was shown that provided suitable regularity conditions are satisfied the second law of thermodynamics in the form of Clausius-Duhem inequality is sufficient for stability. Furthermore, the global existence of smooth solutions for initial states near equilibrium was shown in [97]. For further results for systems of the form (2.1), the reader is referred to the cited works and the references within.

The relative entropy framework we employ also relies on the notion of entropy. In this section, the entropy structure for the model hierarchy at hand is discussed.

The complex system is assumed to be equipped with a strictly convex entropy $H : \mathbb{R}^N \to \mathbb{R}$ and an entropy flux $Q : \mathbb{R}^N \to \mathbb{R}$ satisfying the structure discussed here.

The entropy-entropy flux pair (H, Q) satisfies the compatibility condition

$$\mathrm{D}\, Q(\mathbf{U}) = \mathrm{D}\, H\,(\mathbf{U}) \cdot \mathrm{D}\, \mathbf{F}(\mathbf{U}), \quad \forall\ \mathbf{U} \in \mathbb{R}^N. \tag{2.6}$$

Furthermore, due to the fact that the Hessian of the entropy flux is symmetric, we have

$$(\mathrm{D}\, \mathbf{F}(\mathbf{U}))^T\, \mathrm{D}^2\, H(\mathbf{U}) = \mathrm{D}^2\, H\,(\mathbf{U})\, \mathrm{D}\, \mathbf{F}(\mathbf{U}). \tag{2.7}$$

As a result, smooth solutions of (2.1) satisfy the following entropy balance law

$$\partial_t H\,(\mathbf{U}) + \partial_x Q(\mathbf{U}) = \frac{1}{\varepsilon}\, \mathrm{D}\, H\,(\mathbf{U})\, \mathbf{R}(\mathbf{U}). \tag{2.8}$$

For the complex system, it is assumed that

$$\frac{1}{\varepsilon}\, \mathrm{D}\, H\,(\mathbf{U}) \cdot \mathbf{R}(\mathbf{U}) \leq 0. \tag{2.9}$$

holds. Inequality (2.9) can be viewed as compliance with the second law of thermodynamics.

From hereon, if a system of equations is equipped with a strictly convex entropy and entropy flux and the pair satisfies a compatibility condition such as in (2.6), this is referred to as the entropy structure of the system.

Many systems of the form (2.1) that are based in thermodynamics satisfy the entropy structure, such as combustion [98], thermomechanics [24],[61], multiphase flows [31] to name a few.

Entropy consistency

The restriction of the entropy-entropy flux pair to the equilibrium manifold, i.e. the composition of the Maxwellian with the entropy and entropy flux of the complex system induces an entropy structure for the simple system. This is a consequence of the model hierarchy having the structure discussed in Section 2.2 and the complex system having entropy structure discussed above. We refer to the fact that the simple system fulfills the entropy structure as entropy consistency.

Entropy consistency for the simple system does not trivially follow but needs to be shown. In the work of Tzavaras [94], it was proved that the induced entropy for the simple system is strictly convex and it was assumed that the entropy and entropy flux pair satisfies an equivalent compatibility property. We employ a different approach, where we employ geometric properties in Section 2.3 to show entropy consistency. In this section the entropy structure for the simple system is outlined and is later proven in Section 2.3.2.

The induced strictly convex entropy and entropy pair is denoted by (η, q) and given by

$$\eta(\mathbf{u}) := H(M(\mathbf{u})),\ q(\mathbf{u}) := Q(M(\mathbf{u})). \tag{2.10}$$

The entropy-entropy flux pair (η, q) satisfies the compatibility condition

$$D_u q(\mathbf{u}) = D_u \eta(\mathbf{u}) \cdot D_u \mathbf{g}(\mathbf{u}). \tag{2.11}$$

And as a result, the pair also satisfies the commutativity property

$$(\mathrm{D_u}\,\mathbf{g}(\mathbf{u}))^T\,\mathrm{D_u^2}\,\eta(\mathbf{u}) = \mathrm{D_u^2}\,\eta(\mathbf{u})\,\mathrm{D_u}\,\mathbf{g}(\mathbf{u}). \tag{2.12}$$

Hence smooth solutions to (2.5) satisfy the entropy conservation law

$$\partial_t \eta(\mathbf{u}) + \partial_x q(\mathbf{u}) = 0. \tag{2.13}$$

Remark 2.2.1. Note that, from henceforth, unless otherwise indicated, the subscript u is dropped when denoting the Jacobian and the Hessian with respect to $\mathbf{u}$ for functions of $\mathbf{u}$.

2.3 Geometrical Structure

Based on the abstract framework discussed in Section 2.2 and the entropy structure discussed in Section 2.2.1, some noteworthy geometric properties can be derived. These

properties can be physically motivated and the error analysis relies heavily on having this geometric structure.

Some of the geometric properties we discuss were introduced by Tzavaras in [94]. These have been reproduced since they play an important role in the error analysis. We further derive some properties, which enable us to show entropy consistency and are to be employed in the error analysis. Where deemed necessary, for ease of use, the properties have been expressed in index notation.

2.3.1 Geometric properties

Lemma 2.3.1. *Let the linear projection* $\mathbb{P} : \mathbb{R}^N \to \mathbb{R}^n$ *satisfy*

$$dim\ Im\ \left(\mathbb{P}^T\right) = n,\ \ dim\ Ker\ (\mathbb{P}) = N - n. \tag{2.14}$$

Then $\mathbb{R}^N$ *can be decomposed as*

$$\mathbb{R}^N = Im\ (\mathbb{P}^T) \bigoplus Ker\ (\mathbb{P}) \tag{2.15}$$

i.e. any vector $\mathbf{U} \in \mathbb{R}^N$ *can be decomposed as*

$$\mathbf{U} = \mathbb{P}^T\mathbf{w} + \mathbf{V},\ \ \mathbf{w} \in \mathbb{R}^n,\ \ \mathbf{V} \in Ker\ (\mathbb{P}), \tag{2.16}$$

where

$$\mathbf{w} = \left(\mathbb{P}\mathbb{P}^T\right)^{-1}\mathbb{P}\mathbf{U},\ \ \mathbf{V} = \left(\mathrm{I} - \mathbb{P}^T\left(\mathbb{P}\mathbb{P}^T\right)^{-1}\mathbb{P}\right)\mathbf{U}. \tag{2.17}$$

Proof. We know that

$$\mathrm{Im}\ (\mathbb{P}^T) \perp\ \mathrm{Ker}\ (\mathbb{P}),$$

which allows $\mathbb{R}^N$ to be decomposed as a direct sum as in (2.15). Hence any vector $\mathbf{V} \in \mathbb{R}^N$ can be decomposed as per (2.16).

Note that

$$\mathbb{P}\mathbf{U} = \mathbb{P}\mathbb{P}^T\mathbf{w},$$

hence

$$\mathbf{w} = \left(\mathbb{P}\mathbb{P}^T\right)^{-1}\mathbb{P}\mathbf{U},$$

which gives the decomposition

$$\mathbf{w} = \left(\mathbb{P}\mathbb{P}^T\right)^{-1}\mathbb{P}\mathbf{U},\ \ \mathbf{V} = \left(\mathrm{I} - \mathbb{P}^T\left(\mathbb{P}\mathbb{P}^T\right)^{-1}\mathbb{P}\right)\mathbf{U}.$$

Note that $\mathbb{P}\mathbb{P}^T$ is invertible because the assumption is that the projection matrix $\mathbb{P}$ has full rank. □

Lemma 2.3.2. *The entropy gradient is orthogonal to the gradient of the source term and the null space of the projection matrix $\mathbb{P}$ on the equilibrium manifold, i.e.*

$$Im\ (\mathrm{D}\,\mathbf{R}(M(\mathbf{u}))) = Ker\ (\mathbb{P})\quad \forall\ \mathbf{u} \in \mathbb{R}^n, \tag{2.18}$$

$$\mathrm{D}\,H\,(M(\mathbf{u})) \perp Ker\ (\mathbb{P})\quad \forall\ \mathbf{u} \in \mathbb{R}^n. \tag{2.19}$$

Proof. Recalling (2.2)

$$\mathbb{P}\mathbf{R}(\mathbf{U}) = 0.$$

Differentiating with respect to $\mathbf{U}$ and multiplying from the right by a vector $\mathbf{A} \in \mathbb{R}^N$, we have

$$\mathbb{P}\,\mathrm{D}\,\mathbf{R}(\mathbf{U})\mathbf{A} = 0.$$

Hence, we can infer that Im $(\mathrm{D}\,\mathbf{R}(\mathbf{U})) \subset$ Ker $(\mathbb{P})$.

Due to (2.4) and (2.14), we have

$$\text{Im}\ (\mathrm{D}\,\mathbf{R}\,(M\,(\mathbf{u}))) = \text{Ker}\ (\mathbb{P}). \tag{2.20}$$

Next, we show that (2.19) holds. The entropy structure is such that

$$\frac{1}{\varepsilon}\,\mathrm{D}\,H\,(\mathbf{U}) \cdot \mathbf{R}(\mathbf{U}) \leq 0 \tag{2.21}$$

holds. Furthermore, on the equilibrium manifold

$$\mathrm{D}\,H\,(M(\mathbf{u})) \cdot \mathbf{R}(M(\mathbf{u})) = 0.$$

Let

$$\phi(t) := \mathrm{D}\,H\,(M(\mathbf{u}) + t\mathbf{A}) \cdot \mathbf{R}(M(\mathbf{u}) + t\mathbf{A}) \leq 0,\ \forall\ t \in \mathbb{R},\ \mathbf{A} \in \mathbb{R}^N.$$

$\phi(0) = 0$ is an extremum, which implies

$$\phi'(0) \quad = (\mathrm{D}^2\,H\,(M(\mathbf{u}))\,\mathbf{A}) \cdot \mathbf{R}\,(M(\mathbf{u})) + \mathrm{D}\,H\,(M(\mathbf{u})) \cdot \mathrm{D}\,\mathbf{R}\,(M(\mathbf{u}))\,\mathbf{A} \quad = 0$$

The first term vanishes since the source term vanishes for states on the equilibrium manifold. As a result, we get

$$\mathrm{D}\,H\,(M(\mathbf{u})) \perp \text{Im}\ (\mathrm{D}\,\mathbf{R}\,(M(\mathbf{u})))\,.$$

Employing (2.18), we get the desired orthogonality condition

$$\mathrm{D}\,H\,(M(\mathbf{u})) \perp \text{Ker}\ (\mathbb{P}).$$

□

Remark 2.3.3. For any vector $\mathbf{V} \in \text{Ker}\ (\mathbb{P})$, employing (2.19) gives

$$\mathbf{V}^T \cdot \mathrm{D}\, H\left(M(\mathbf{u})\right) = 0,$$

which in index notation can be written as

$$\sum_{k=1}^{N} \frac{\partial H}{\partial U_k}\left(M\left(\mathbf{u}\right)\right) \cdot V_k = 0$$

where V_k is the k^{th} component of the vector $\mathbf{V}$.

The dissipative dynamics occur in $\mathbb{R}^N$ and the equilibrium dynamics can be modeled by quantities in $\mathbb{R}^n$. This allows to derive relations between the projection matrix, the Maxwellian and the basis in $\mathbb{R}^n$ as presented in the next lemma.

Lemma 2.3.4. *For the Maxwellian as defined in (2.3), the following holds*

$$\sum_{j=1}^{N} \mathbb{P}_{ij} \frac{\partial M_j(\mathbf{u})}{\partial u_k} = \delta_{ik},\ \ i,k = 1,2,\ldots,n, \tag{2.22}$$

$$\sum_{j=1}^{N} \mathbb{P}_{ij} \frac{\partial^2 M_j(\mathbf{u})}{\partial u_k \partial u_l} = 0,\ \ i,k,l = 1,2,\ldots,n, \tag{2.23}$$

Proof. Recalling (2.5)

$$\mathbb{P}M(\mathbf{u}) = \mathbf{u}.$$

Differentiating with respect to $\mathbf{u}$, we have

$$\mathbb{P}\,\mathrm{D}_u\, M(\mathbf{u}) = \mathrm{I},$$

The above equation can be written in index notation to give (2.22). The above equation can also be expressed as

$$\mathbb{P}\frac{\partial M\left(\mathbf{u}\right)}{\partial u_k} = e_k = (0\ldots1\ldots0)^T,\ \ k = 1,\ldots,n$$

where u_k is the k^{th} component. We can observe the the vectors e_k form a basis in $\mathbb{R}^n$.

Next, differentiating (2.22) again with respect to u_l we get (2.23).

□

Lemma 2.3.5. *For the Maxwellian as defined in (2.3), the following holds*

$$Im\ \left(\mathrm{D}_u\, M(\mathbf{u})\mathbb{P} - \mathrm{I}\right) \subset Ker\ (\mathbb{P})\ \ \forall\ \mathbf{u} \in \mathbb{R}^n. \tag{2.24}$$

Proof. Multiplying (2.22) from the right by the projection matrix and employing the fact that multiplication with the identity matrix is commutative, we get

$$\mathbb{P}\,\mathrm{D}_u\, M(\mathbf{u})\mathbb{P} = \mathrm{I}\,\mathbb{P} = \mathbb{P}\,\mathrm{I}$$

Hence

$$\mathbb{P}\,\mathrm{D}_u\, M(\mathbf{u})\mathbb{P} - \mathbb{P}\,\mathrm{I} = 0$$

$$\Rightarrow \mathbb{P}\,(\mathrm{D}_u\, M(\mathbf{u})\mathbb{P} - \mathrm{I}) = 0$$

We can deduce that the columns of $\mathrm{D}_u\, M(\mathbf{u})\mathbb{P} - \mathrm{I}$ are in Ker $(\mathbb{P})$, i.e.

$$\text{Im }\ (\mathrm{D}_u\, M(\mathbf{u})\mathbb{P} - \mathrm{I}) \subset \text{Ker }(\mathbb{P}).$$

□

Remark 2.3.6. Note that (2.24) can be written in index notation as

$$\sum_{j=1}^{N} \mathbb{P}_{ij} \sum_{k=1}^{n} \left(\frac{\partial M_j\,(\mathbf{u})}{\partial u_k} \cdot \mathbb{P}_{km} - \delta_{jm}\delta_{ik} \right) = 0,\ \forall\ m = 1, 2, \ldots, N \text{ and } i = 1, 2, \ldots, n.$$

Next, we derive an orthogonality property which will be useful when deriving the error estimates.

Lemma 2.3.7. *Let* $\mathbf{u} \in \mathbb{R}^n$*, then any vector* $\mathbf{V}$ *of the form*

$$\mathbf{V} = \mathbf{A} - \mathrm{D}\,M\,(\mathbf{u}) \cdot \mathbb{P}\mathbf{A}, \quad \mathbf{A} \in \mathbb{R}^N \tag{2.25}$$

satisfies

$$\mathbf{V}^T \cdot \mathrm{D}^2\,H\,(M\,(\mathbf{u})) \cdot \mathrm{D}\,M\,(\mathbf{u}) = 0 \tag{2.26}$$

Proof. To show that (2.26) holds, we first derive some other geometric properties.

Employing (2.22), we have

$$\mathbb{P}\,\mathrm{D}\,M\,(\mathbf{u}) \cdot \mathbb{P}\mathbf{A} = \mathbb{P}\mathbf{A}. \tag{2.27}$$

Hence

$$\mathbb{P}\mathbf{V} = 0 \tag{2.28}$$

and

$$\mathbf{V} \in \text{ Ker }(\mathbb{P}).$$

Next, differentiating the vector $\mathbf{V}$ with respect to $\mathbf{u}$ and projecting it, we get

$$\mathbb{P}\,\mathrm{D}_u\,\mathbf{V} = -\,\mathrm{D}_u\,(\mathbb{P}\,\mathrm{D}\,M(\mathbf{u}) \cdot \mathbb{P}\mathbf{A}) = 0, \tag{2.29}$$

where we have employed (2.27). Hence

$$\mathrm{D}_u\,\mathbf{V} \in \text{ Kern }(\mathbb{P}). \tag{2.30}$$

From (2.19) and (2.28) we have

$$\mathbf{V}^T \cdot \mathrm{D}\, H\left(M\left(\mathbf{u}\right)\right) = 0.$$

Differentiating with respect to $\mathbf{u}$, we get

$$\mathrm{D}_u \left(\mathbf{V}^T \cdot \mathrm{D}\, H\left(M\left(\mathbf{u}\right)\right)\right) = \left(\mathrm{D}\, H\left(M\left(\mathbf{u}\right)\right)\right)^T \cdot \mathrm{D}_u\, \mathbf{V} + \mathbf{V}^T \cdot \mathrm{D}^2\, H\left(M\left(\mathbf{u}\right)\right) \cdot \mathrm{D}\, M\left(\mathbf{u}\right) = 0$$

The first term to the right vanishes due to (2.19) and (2.30), which gives

$$\mathbf{V}^T \cdot \mathrm{D}^2\, H\left(M\left(\mathbf{u}\right)\right) \cdot \mathrm{D}\, M\left(\mathbf{u}\right) = 0. \tag{2.31}$$

□

Remark 2.3.8. Note that Tzavaras derived the properties (2.15)-(2.22) in [94]. We have further introduced (2.24)-(2.26).

2.3.2 Entropy consistency

Next, we employ the geometric properties presented in Section 2.3 to show entropy consistency for the induced entropy-entropy flux pair (η, q) of the simple system, i.e. we show that η is strictly convex and that the pair (η, q) satisfies compatibility condition (2.11).

Lemma 2.3.9. *The induced entropy η as defined in (2.10) is strictly convex.*

Proof. From orthogonality conditions (2.19) and (2.23) we have

$$\sum_{k=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_k} \cdot \frac{\partial^2 M_k(\mathbf{u})}{\partial u_i \partial u_j} = 0, \forall\ i, j = 1, 2, \ldots, n. \tag{2.32}$$

H is strictly convex, hence $\mathrm{D}^2\, H\left(\mathbf{U}\right)$ is a symmetric positive definite matrix and as a result

$$\sum_{i=1}^{N} X_i \cdot \sum_{j=1}^{N} \frac{\partial^2 H\left(M\left(\mathbf{u}\right)\right)}{\partial U_i \partial U_j} \cdot X_j > 0 \tag{2.33}$$

holds.

The components of $\mathrm{D}^2\, \eta(\mathbf{u})$ in index notation can be written as

$$\begin{aligned}
\frac{\partial^2 \eta(\mathbf{u})}{\partial u_i \partial u_j} = \frac{\partial^2 H\left(M\left(\mathbf{u}\right)\right)}{\partial u_i \partial u_j} &= \sum_{k=1}^{N} \frac{\partial M_k(\mathbf{u})}{\partial u_j} \cdot \sum_{m=1}^{N} \frac{\partial^2 H\left(M\left(\mathbf{u}\right)\right)}{\partial U_k \partial U_m} \cdot \frac{\partial M_m(\mathbf{u})}{\partial u_i} \\
&\quad + \sum_{k=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_k} \cdot \frac{\partial^2 M_k(\mathbf{u})}{\partial u_i \partial u_j} \\
&= \sum_{k=1}^{N} \frac{\partial M_k(\mathbf{u})}{\partial u_j} \cdot \sum_{m=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_k \partial U_m} \frac{\partial M_m(\mathbf{u})}{\partial u_i},
\end{aligned}$$

where the second term vanishes due to (2.32).

Hence for $\mathbf{y} \in \mathbb{R}^n$

$$\begin{aligned}\mathbf{y}^T \mathrm{D}^2 \eta(\mathbf{u})\mathbf{y} &= \sum_{m=1}^{N} \left(\sum_{i=1}^{n} y_i \frac{\partial M_m(\mathbf{u})}{\partial u_i} \right) \cdot \sum_{k=1}^{N} \frac{\partial^2 H\left(M\left(\mathbf{u}\right)\right)}{\partial U_k \partial U_m} \cdot \left(\sum_{j=1}^{n} \frac{\partial M_k(\mathbf{u})}{\partial u_j} y_j \right) \\ &= \left(\mathbf{y}\,\mathrm{D}\,M\left(\mathbf{u}\right)\right)^T \cdot \mathrm{D}^2 H\left(M\left(\mathbf{u}\right)\right) \cdot \left(\mathbf{y}\,\mathrm{D}\,M\left(\mathbf{u}\right)\right) > 0,\end{aligned}$$

where we have used the fact that H is convex (2.33) and that $\mathrm{D}\,M\left(\mathbf{u}\right)$ is injective. □

Lemma 2.3.10. *The induced entropy and entropy flux pair (η, q) of the model hierarchy 2.2 as defined in (2.10) satisfies the compatibility condition*

$$D_u \eta(\mathbf{u}) \cdot D_u \mathbf{g}(\mathbf{u}) = D_u q(\mathbf{u}). \tag{2.34}$$

Proof. Employing (2.19) and (2.24), we have

$$\sum_{j=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_j} \cdot \sum_{k=1}^{n} \left(\frac{\partial M_j\left(\mathbf{u}\right)}{\partial u_k} \cdot \mathbb{P}_{km} - \delta_{jm}\delta_{ik} \right) = 0, \text{ for } m = 1, 2, \ldots, N, \text{ and } i = 1, 2, \ldots, n. \tag{2.35}$$

Let $\mathbf{V} \in \mathbb{R}^N$, then from (2.35) we have

$$\sum_{k=1}^{n} \sum_{m,j=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_j} \cdot \left(\frac{\partial M_j\left(\mathbf{u}\right)}{\partial u_k} \cdot \mathbb{P}_{km} - \delta_{jm}\delta_{ik} \right) \cdot V_m = 0 \text{ for } i = 1, 2, \ldots, n. \tag{2.36}$$

We wish to show that (2.34) holds. This can be written as follows

$$\left(\mathrm{D}_u\, \eta(\mathbf{u}) \cdot \mathrm{D}_u \left(\mathbb{P}\mathbf{F}\left(M\left(\mathbf{u}\right)\right)\right)\right)_j = \left(\mathrm{D}\, q(\mathbf{u})\right)_j = \frac{\partial Q(M\left(\mathbf{u}\right))}{\partial u_j} \text{ for } j = 1, \ldots, n.$$

Expanding the quantity to the left

$$\left(\mathrm{D}_u\, \eta(\mathbf{u}) \cdot \mathrm{D}_u \left(\mathbb{P}\mathbf{F}\left(M\left(\mathbf{u}\right)\right)\right)\right)_j = \sum_{k=1}^{n} \left(\sum_{m=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m} \cdot \frac{\partial M_m(\mathbf{u})}{\partial u_k} \right) \cdot \left(\sum_{s=1}^{N} \mathbb{P}_{ks} \frac{\partial F_s\left(M\left(\mathbf{u}\right)\right)}{\partial u_j} \right)$$

$$\begin{aligned}
&= \sum_{k=1}^{n} \sum_{s,m=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m} \cdot \left(\frac{\partial M_m(\mathbf{u})}{\partial u_k} \mathbb{P}_{ks} \right) \cdot \frac{\partial F_s\left(M\left(\mathbf{u}\right)\right)}{\partial u_j} \\
&= \sum_{k=1}^{n} \sum_{s,m=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m} \cdot \left(\frac{\partial M_m(\mathbf{u})}{\partial u_k} \mathbb{P}_{ks} + \delta_{ms}\delta_{jk} - \delta_{ms}\delta_{jk} \right) \cdot \frac{\partial F_s\left(M\left(\mathbf{u}\right)\right)}{\partial u_j} \\
&= \sum_{k=1}^{n} \sum_{s,m=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m} \cdot \left(\frac{\partial M_m(\mathbf{u})}{\partial u_k} \mathbb{P}_{ks} - \delta_{ms}\delta_{jk} \right) \cdot \frac{\partial F_s\left(M\left(\mathbf{u}\right)\right)}{\partial u_j} \\
&+ \sum_{k=1}^{n} \sum_{s,m=1}^{N} \frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m} \cdot \delta_{ms}\delta_{jk} \cdot \frac{\partial F_s\left(M\left(\mathbf{u}\right)\right)}{\partial u_j}
\end{aligned}$$

The first term vanishes due to (2.36). Hence, we have

$$\begin{aligned}\left(\mathrm{D}_u\,\eta(\mathbf{u})\cdot\mathrm{D}_u\left(\mathbb{P}\mathbf{F}\left(M\left(\mathbf{u}\right)\right)\right)\right)_j &= \sum_{k=1}^{n}\sum_{s,m=1}^{N}\frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m}\cdot\delta_{ms}\delta_{jk}\cdot\frac{\partial F_s\left(M\left(\mathbf{u}\right)\right)}{\partial u_j}\\ &= \sum_{k=1}^{n}\delta_{jk}\sum_{m=1}^{N}\frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m}\cdot\frac{\partial F_m\left(M\left(\mathbf{u}\right)\right)}{\partial u_j}\\ &= \sum_{m=1}^{N}\frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m}\cdot\frac{\partial F_m\left(M\left(\mathbf{u}\right)\right)}{\partial u_j}\end{aligned}$$

Further expanding using the chain rule, we get

$$\begin{aligned}\sum_{m=1}^{N}\frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m}\cdot\frac{\partial F_m\left(M\left(\mathbf{u}\right)\right)}{\partial u_j} &= \sum_{m=1}^{N}\frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m}\cdot\sum_{i=1}^{N}\frac{\partial F_m\left(M\left(\mathbf{u}\right)\right)}{\partial U_i}\cdot\frac{\partial M_i\left(\mathbf{u}\right)}{\partial u_j}\\ &= \sum_{i=1}^{N}\left(\sum_{m=1}^{N}\frac{\partial H\left(M\left(\mathbf{u}\right)\right)}{\partial U_m}\cdot\frac{\partial F_m\left(M\left(\mathbf{u}\right)\right)}{\partial U_i}\right)\cdot\frac{\partial M_i\left(\mathbf{u}\right)}{\partial u_j}\\ &= \sum_{i=1}^{N}\frac{\partial Q\left(M\left(\mathbf{u}\right)\right)}{\partial U_i}\cdot\frac{\partial M_i\left(\mathbf{u}\right)}{\partial u_j}\\ &= \frac{\partial Q(M\left(\mathbf{u}\right))}{\partial u_j}\end{aligned}$$

Hence the pair (η, q) satisfies the compatibility condition (2.11). □

2.4 Relative entropy stability framework

The relative entropy stability framework is a well established stability theory for hyperbolic balance laws. It has been used to show weak-strong uniqueness under suitable regularity conditions. It has also been employed to derive a posteriori error estimates for discontinuous Galerkin schemes and model adaptation. Some relevant works are briefly described in this section. The list of works is not meant to be exhaustive but to give an impression of the previous work done. The interested reader is referred to the cited research and the references within.

Dafermos [26] and DiPerna[29] established the weak-strong stability framework for multi-dimensional systems of conservation laws equipped with convex entropy-entropy flux. It was shown that the weak solution satisfying the entropy inequality is unique as along as a Lipschitz continuous solution exists. However, if a Lipschitz solution does not exist, then the entropy inequality is no longer sufficient to guarantee uniqueness. For instance, De Lellis and Székelyhidi showed [28] that one entropy inequality is not enough to single out a unique entropy admissible weak solution for the Euler equations in more than one space dimension. In fact, for a large class of initial data it is possible to construct infinitely many weak solutions.

The structure and the model hierarchy we look at was introduced by Tzavaras in [94], where the relative entropy stability framework was employed to show the convergence of smooth solutions to the complex system with vanishing ε to that of the simple system, as long as the simple system has a strong solution. This was further extended in the work of [69] where an additional weakly dissipative source term was introduced.

The relative entropy framework has also been employed in various other contexts. We cite a few examples to given an idea of the range of the research that has been conducted. Berthelin and Vasseur [6] proved that the kinetic equations (Fokker-Planck equation for isothermal case and BGK- like equations) asymptotically converge to the multi-dimensional isentropic gas dynamics equations. Lattanzio and Tzavars showed in [61] the convergence of weak entropy admissible solutions of compressible gas dynamics in large friction limit to the porus media equations. Feireisl et. al. [33] showed for the compressible Navier-Stokes equations that any finite energy weak solution, i.e a weak solution satisfying an energy inequality, is a suitable weak solution [34], i.e a weak solution satisfying a relative entropy inequality with a couple of functions satisfying some constraints.

Furthermore, the stability framework was exploited in [40], [42],[43] for constructing a posteriori error estimates for Runge Kutta Discontinuous Galerkin methods. It was also employed in [44] to construct a posteriori error estimates to carry out mesh and model adaptation for convection dominated problems. The model hierarchy consisted of the Navier-Stokes-Fourier system and the compressible Euler equations.

The analysis in our research is in the spirit of the ideas that came before. The novelty of this research is exploiting the entropy and the geometric structure to derive computable a posteriori error estimates for the model hierarchy under consideration, which allow error balancing between the two sources of error; discretization and modeling errors, allowing heterogeneous mesh and model adaptation.

Next, the notions of relative entropy, relative entropy flux and relative entropy dissipation, which form the basis of the relative entropy stability framework are introduced.

2.4.1 Relative entropy

The relative entropy stability framework uses the notion of relative entropy and relative entropy flux to compare two solutions.

The relative entropy and entropy flux to compare states $\mathbf{U}, \mathbf{V} \in \mathbb{R}^N$ is defined as

$$H\left(\mathbf{U}|\mathbf{V}\right) := H(\mathbf{U}) - H(\mathbf{V}) - \mathrm{D}\, H(\mathbf{V}) \cdot \left(\mathbf{U} - \mathbf{V}\right), \tag{2.37}$$

$$Q\left(\mathbf{U}|\mathbf{V}\right) := Q(\mathbf{U}) - Q(\mathbf{V}) - \mathrm{D}\, H(\mathbf{V}) \cdot \left(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}\left(\mathbf{V}\right)\right). \tag{2.38}$$

2.4.2 Relative entropy dissipation

The difference in modeling of the reaction is characterised by so called relative entropy dissipation. This approach was introduced by Tzavaras in [94] and also employed by Miroshnikov et.al. in [69].

The relative entropy dissipation is defined as

$$\mathfrak{D}(\mathbf{U}|\mathbf{V}) := -\left(\mathrm{D}\,H\left(\mathbf{U}\right) - \mathrm{D}\,H(\mathbf{V})\right)\cdot\left(\mathbf{R}(\mathbf{U}) - \mathbf{R}(\mathbf{V})\right), \tag{2.39}$$

for states $\mathbf{U}, \mathbf{V} \in \mathbb{R}^N$.

The relative dissipation between the state $\mathbf{U}$ and its counterpart on the equilibrium manifold $M(\mathbb{P}\mathbf{U})$ is of particular interest, i.e.

$$\mathfrak{D}(\mathbf{U}|M(\mathbb{P}\mathbf{U})) = -\left(\mathrm{D}\,H\left(\mathbf{U}\right) - \mathrm{D}\,H(M(\mathbb{P}\mathbf{U}))\right)\cdot\left(\mathbf{R}(\mathbf{U}) - \mathbf{R}(M(\mathbb{P}\mathbf{U}))\right). \tag{2.40}$$

The relative entropy dissipation in this case simplifies to

$$-\frac{1}{\varepsilon}\mathfrak{D}(\mathbf{U}|M(\mathbb{P}\mathbf{U})) = \frac{1}{\varepsilon}\,\mathrm{D}\,H\left(\mathbf{U}\right)\cdot\mathbf{R}(\mathbf{U}). \tag{2.41}$$

We assume that for every $\mathcal{B} \subset \mathbb{R}^N$, such that $\mathbf{U}, M(\mathbb{P}\mathbf{U}) \in \mathcal{B}$, there exists $\nu\left(\mathcal{B}\right) > 0$ such that

$$\mathfrak{D}(\mathbf{U}|M(\mathbb{P}\mathbf{U})) \geq \nu|\mathbf{U} - M(\mathbb{P}\mathbf{U})|^2. \tag{2.42}$$

2.5 A posteriori error analysis

Equipped with the abstract framework of the model hierarchy, we turn to the a posteriori error analysis.

2.5.1 Domain decomposition

As discussed in Section 2.1, dynamic heterogeneous model adaptation involves decomposing the computational domain after each time step and employing the simple system in a sub-domain and the complex system in the rest of the computational domain. But, for the sake of simplicity and clarity of the error analysis, in this chapter we fix the sub-domains in which the simple and the complex system are employed. A heterogeneous mesh and model adaptation strategy, where the domain is decomposed after each time step, is proposed in Chapter 3.

Let the sub-domain where we solve the complex system and the sub-domain where we solve the simple system be denoted by Ω_c and Ω_s respectively. The sub-domains are such that they do not intersect, i.e.

$$\overline{\Omega}_c \cup \overline{\Omega}_s = \mathbb{T}, \quad \Omega_c \cap \Omega_s = \emptyset. \tag{2.43}$$

The set of interfaces is denoted by

$$\overline{\Omega}_c \cap \overline{\Omega}_s =: \Gamma. \tag{2.44}$$

2.5.2 Reconstruction

We employ the relative entropy stability framework to derive the error estimates. The error analysis requires the quantity being compared to the exact solution to be Lipschitz continuous. As the numerical solution itself generally will not have the necessary regularity, a reconstruction technique is employed. The reconstruction should be such that it is explicitly computable and that it satisfies the necessary regularity conditions so that relative entropy framework can be applied. The error analysis itself is independent of the discretization method and the reconstruction method. Hence at this point, we assume the existence of a numerical solution to the model adaptive system and a Lipschitz continuous reconstruction of the numerical solution.

Hypothesis 2.5.1. Let $\mathbf{V}_h \in \mathbb{R}^N$ be a numerical solution for time $0 \leq t \leq T$ on the computational domain $\mathbb{T}$, with initial condition $\mathbf{V}_0 : \mathbb{T} \to \mathbb{R}^N$ on a decomposed domain as defined in Section 2.5.1, such that the restrictions

$$\mathbf{U}_h := \mathbf{V}_h\Big|_{\Omega_c}, \quad \mathbf{u}_h := \mathbb{P}\mathbf{V}_h\Big|_{\Omega_s} \tag{2.45}$$

are numerical solutions to (2.1) and (2.5) on Ω_c and Ω_s respectively, where appropriate coupling conditions are employed at the interfaces.

Based on the numerical solution as defined in the Hypothesis 2.5.1, we assume the existence of a reconstruction as follows.

Hypothesis 2.5.2. We assume the existence of a space-time Lipschitz continuous reconstruction $\hat{\mathbf{U}}_h : \Omega_c \times [0,T] \to \mathbb{R}^N$, $\hat{\mathbf{u}}_h : \Omega_s \times [0,T] \to \mathbb{R}^N$ of the numerical solution defined in Hypothesis 2.5.1 on Ω_c and Ω_s respectively, such that

$$\hat{\mathbf{U}}_h(x,t) = M(\hat{\mathbf{u}}_h(x,t)), \quad \forall\, x \in \Gamma, t \in [0,T]. \tag{2.46}$$

The reason for reconstructing the numerical solution such that (2.46) is satisfied at the interface will become apparent in Section 2.5.6, where we derive the error estimates.

Remark 2.5.3. The reconstructions defined in Hypothesis 2.5.2 satisfy the following perturbed systems of equations

$$\partial_t \hat{\mathbf{U}}_h + \partial_x \mathbf{F}(\hat{\mathbf{U}}_h) - \frac{1}{\varepsilon}\mathbf{R}(\hat{\mathbf{U}}_h) =: \mathcal{R}_c \tag{2.47}$$

$$\partial_t \hat{\mathbf{u}}_h + \partial_x \mathbb{P}\mathbf{F}(M(\hat{\mathbf{u}}_h)) =: r_s \tag{2.48}$$

$$\partial_t M\left(\hat{\mathbf{u}}_h\right) + \partial_x \mathbf{F}(M(\hat{\mathbf{u}}_h)) =: \mathcal{R}_s \tag{2.49}$$

with explicitly computable residuals $\mathcal{R}_c \in L_2(\Omega_c \times [0,T], \mathbb{R}^N)$, $\mathcal{R}_s \in L_2(\Omega_s \times [0,T], \mathbb{R}^N)$ and $r_s \in L_2(\Omega_s \times [0,T], \mathbb{R}^n)$, where $\mathbb{P}\mathcal{R}_s = r_s$.

The reconstructions also satisfy the following entropy balance laws

$$\partial_t H\left(M\left(\hat{\mathbf{u}}_h\right)\right) + \partial_x Q(M\left(\hat{\mathbf{u}}_h\right)) = \mathrm{D}\, H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \mathcal{R}_s \tag{2.50}$$

and

$$\partial_t H(\hat{\mathbf{U}}_h) + \partial_x Q(\hat{\mathbf{U}}_h) = \frac{1}{\varepsilon} \mathrm{D}\, H(\hat{\mathbf{U}}_h) \cdot \mathbf{R}(\hat{\mathbf{U}}_h) + \mathrm{D}\, H(\hat{\mathbf{U}}_h) \cdot \mathcal{R}_c \tag{2.51}$$

on Ω_s and Ω_c respectively.

Remark 2.5.4. We employ a multiwave resolution based Runge-Kutta Discontinuous Galerkin method (RKDG) for discretization [37], [39] and the reconstruction method introduced in [40], [42] in the context of a posteriori error analysis for hyperbolic conservation laws. The details of the discretization method and the reconstruction will be provided in Chapter 3.

2.5.3 Structure of the error estimate

The objective of the error analysis is to bound from above the distance between the exact solution to the complex system and the numerical solution to the model adaptive system. The two sources of error present are the discretization and the modeling error. The discretization error is incurred for both the simple system and the complex system and the modeling error only for the simple system. We have the choice to either model or mesh coarsen when employing the complex system to reduce the required computational resources. The nature and the consequence of mesh and model coarsening are very different. Hence, the decision to mesh or model coarsen should be made independently. To enable this, the error estimates should consist of distinguishable contributions of the discretization and modeling error.

We wish to bound the distance between the numerical and the exact solution. To this end, we assume the existence of an exact solution $\mathbf{U}$ as follows.

Hypothesis 2.5.5. Let there exist a smooth solution $\mathbf{U} : \mathbb{T} \times [0, T] \to \mathbb{R}^N$ where $T > 0$ to (2.1) with initial condition $\mathbf{V}_0 : \mathbb{T} \to \mathbb{R}^N$. Furthermore, we assume that the solution $\mathbf{U}$ takes values in a a priori known compact and convex set $\mathcal{D}$, i.e.

$$\mathbf{U}(x, t) \in \mathcal{D}, \quad \forall (x, t) \in \mathbb{T} \times [0, T]. \tag{2.52}$$

Remark 2.5.6. Let $\mathbf{U}$ take values in $\mathcal{D}$ and let $\mathbf{u} := \mathbb{P}\mathbf{U}$. Furthermore, let $h_i : \mathbb{R}^N \to \mathbb{R}$ and $m_k : \mathbb{R}^n \to \mathbb{R}^n$ be as follows

$$h_i(\mathbf{U}) := \frac{\partial H(\mathbf{U})}{\partial U_i}, \quad i = 0, \ldots, N, \tag{2.53}$$

$$m_k(\mathbf{u}) := \mathrm{D_u}\, M_k(\mathbf{u}), \quad k = 0, \ldots, N. \tag{2.54}$$

Then, there are constants $0 < C_{\bar{F}} < \infty$ and $0 < C_{\underline{H}}, C_{\bar{H}}, C_{\bar{M}} < \infty$ such that

$$\left(\sum_{i=0}^{N} \left(\mathbf{V}^T \mathrm{D}^2\, \mathbf{F}_i(\mathbf{U}) \mathbf{V} \right)^2 \right)^{\frac{1}{2}} \leq C_{\bar{F}} |\mathbf{V}|^2 \tag{2.55}$$

$$C_{\underline{H}} |\mathbf{V}|^2 \leq \mathbf{V}^T \mathrm{D}^2\, H(\mathbf{U}) \mathbf{V} \leq C_{\bar{H}} |\mathbf{V}|^2 \tag{2.56}$$

$$\left(\sum_{i=0}^{N}\left(\mathbf{V}^T\,\mathrm{D}^2\,h_i(\mathbf{U})\mathbf{V}\right)^2\right)^{\frac{1}{2}} \leq C_{\bar{H}}|\mathbf{V}|^2 \tag{2.57}$$

$$\left(\sum_{k=0}^{n}\left(\left(\mathbb{P}\mathbf{V}\right)^T\,\mathrm{D_u}\,m_k(\mathbf{u})\,\left(\mathbb{P}\mathbf{V}\right)\right)^2\right)^{\frac{1}{2}} \leq C_{\bar{M}}|\mathbf{V}|^2 \tag{2.58}$$

for all $\mathbf{V} \in \mathbb{R}^N$, where $|\cdot|$ is the Euclidean norm for vectors and Frobenius norm for matrices. The constants can be explicitly calculated from $\mathcal{D}, H, M$ and $\mathbf{F}$.

Remark 2.5.7. Furthermore, for $\mathbf{U}, \mathbf{V} \in \mathcal{D}$, the strict convexity of the entropy implies

$$C_{\underline{H}}|\mathbf{U}-\mathbf{V}|^2 \leq H(\mathbf{U}|\mathbf{V}). \tag{2.59}$$

We can observe that the relative entropy forms a measure of distance between the states $\mathbf{U}$ and $\mathbf{V}$.

In the error analysis, we wish to bound the distance between $\mathbf{U}$ and $\mathbf{V}_h$. To this end, we employ the following error splitting

$$\begin{aligned}|\mathbf{U}-\mathbf{U}_h| &\leq |\mathbf{U}-\hat{\mathbf{U}}_h| + |\hat{\mathbf{U}}_h - \mathbf{U}_h|,\\ |\mathbf{U}-M\left(\mathbf{u}_h\right)| &\leq |\mathbf{U}-M(\hat{\mathbf{u}}_h)| + |M(\hat{\mathbf{u}}_h)-M(\mathbf{u}_h)|.\end{aligned}$$

The first terms on the right in the equations above will be bounded by the error estimates, derived employing the relative entropy framework, and the second terms on the right are explicitly computable.

2.5.4 Residual Splitting

The residual of the simple system contains information about the discretization and modeling errors. In this section, we propose a splitting of the residual of the simple system, with the objective of distinguishing the modeling and the discretization error. Furthermore, the residual splitting is done such that the modelling assumption inherent to the simple system is suitably accounted for.

The simplifying assumption we make in order to derive the simple system is that the dynamics is the equilibrium dynamics. Specifically, the assumption is that the speed of the reaction is infinitely fast. As a result, the evolution of the quantities of interest takes place on equilibrium manifold $\mathcal{M}$. Furthermore, since the governing equations of the complex system are simplified by projecting the complex system using the projection matrix $\mathbb{P}$, the simple system accounts for the part of the convection that takes place in Ker $(\mathbb{P})^{\perp}$, but not the part that takes place in Ker $(\mathbb{P})$. If the unaccounted convection becomes significantly large, it could push the system out of equilibrium. Even if the reaction speed is fast, it will be finite. Hence, assuming equilibrium may no longer be appropriate. We should switch to the complex system, if the ratio of convection which might push the states out of equilibrium and the speed of the reaction becomes large. The modelling error indicator, that we have to define, should appropriately reflect this. To this end, a residual splitting is proposed.

Multiplying (2.48) by the Jacobian of the Maxwellian, we get

$$\mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot \partial_t \hat{\mathbf{u}}_h + \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot \mathbb{P} \partial_x \mathbf{F}(M(\hat{\mathbf{u}}_h)) = \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot r_s. \tag{2.60}$$

Furthermore, (2.49) can be expanded as

$$\mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot \partial_t \hat{\mathbf{u}}_h + \partial_x \mathbf{F}(M(\hat{\mathbf{u}}_h)) = \mathcal{R}_s. \tag{2.61}$$

Subtracting the above two equations we have

$$(\mathrm{I} - \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot \mathbb{P}) \cdot \partial_x \mathbf{F}(M(\hat{\mathbf{u}}_h)) = \mathcal{R}_s - \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot r_s. \tag{2.62}$$

From Lemma 2.3.5 we can infer that $(\mathrm{I} - \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot \mathbb{P})$ maps the derivative of the flux to the Kernel of $\mathbb{P}$. Hence, the quantity to the right in (2.62) measures the part of the convection dynamics which is not accounted for when employing the simple system.

This observation inspires the following residual decomposition of the residual $\mathcal{R}_s$:

$$\mathcal{R}_s = \mathcal{R}_\delta + \mathcal{R}_\epsilon, \tag{2.63}$$

where

$$\mathcal{R}_\epsilon := \mathcal{R}_s - \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot r_s, \quad \mathcal{R}_\delta := \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot r_s. \tag{2.64}$$

$\mathcal{R}_\epsilon$ is referred to as the modeling residual and $\mathcal{R}_\delta$ as the weighted discretization residual of the simple system.

Remark 2.5.8. Note that

$$\mathbb{P}\, \mathrm{D}\, M\left(\hat{\mathbf{u}}_h\right) \cdot r_s = r_s \Rightarrow \mathcal{R}_\epsilon \in \mathrm{Ker}\ (\mathbb{P}), \tag{2.65}$$

where we have employed (2.22).

2.5.5 Relative entropy identities

Relative entropy allows to bound from above the distance between two states. With the objective of deriving the error estimates, identities for time evolution of relative entropy are derived.

Simple System

For the simple system we wish to compare $\mathbf{U}$ as defined in Hypothesis 2.5.5 and $\hat{\mathbf{u}}_h$ as defined in Hypothesis 2.5.2. To this end, we derive a relative entropy identity for $H(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right))$.

To account for the finite speed of the reaction, we are able to derive the error estimates such that the modelling error indicator that we define scales with $\varepsilon^{0.5}$. A method to derive the error estimates in order to achieve a better scaling seems to be not possible.

The scaling by $\varepsilon^{0.5}$ is achieved by introducing the term $\frac{1}{\varepsilon}\mathfrak{D}\Big(\mathbf{U}|M\left(\mathbf{u}\right)\Big)$ in the relative entropy identity we derive below.

From (2.8),(2.39), (2.50), (2.51) and the definitions (2.37) and (2.38) we have

$$\begin{aligned}
&\partial_t H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)+\partial_x Q\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)+\frac{1}{\varepsilon}\mathfrak{D}\Big(\mathbf{U}|M\left(\mathbf{u}\right)\Big)\\
&\quad=\Big(\partial_t H\left(\mathbf{U}\right)+\partial_x Q(\mathbf{U})\Big)-\Big(\partial_t H\left(M\left(\hat{\mathbf{u}}_h\right)\right)+\partial_x Q\left(M\left(\hat{\mathbf{u}}_h\right)\right)\Big)+\frac{1}{\varepsilon}\mathfrak{D}\Big(\mathbf{U}|M\left(\mathbf{u}\right)\Big)\\
&\qquad-\partial_t\Big(\mathrm{D}\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\cdot\left(\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)\right)\Big)-\partial_x\Big(\mathrm{D}\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\cdot\left(\mathbf{F}\left(\mathbf{U}\right)-\mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right)\right)\Big)\\
&\quad=-\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\partial_t M\left(\hat{\mathbf{u}}_h\right)\Big)\cdot\Big(\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)\Big)\\
&\qquad-\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\partial_x M\left(\hat{\mathbf{u}}_h\right)\Big)\cdot\Big(\mathbf{F}\left(\mathbf{U}\right)-\mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right)\Big)-\frac{1}{\varepsilon}\,\mathrm{D}\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\cdot\mathbf{R}\left(\mathbf{U}\right)
\end{aligned}$$

The last term on the right vanishes due to the fact that $\mathbf{R}(\mathbf{U})\in$ Ker $(\mathbb{P})$, for which property (2.19) holds.

Furthermore employing (2.50) and the commutativity property (2.7), we get

$$\begin{aligned}
&\partial_t H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)+\partial_x Q\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)+\frac{1}{\varepsilon}\mathfrak{D}\Big(\mathbf{U}|M\left(\mathbf{u}\right)\Big)\\
&\quad=-\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\partial_x M\left(\hat{\mathbf{u}}_h\right)\Big)\cdot\Big(\mathbf{F}\left(\mathbf{U}\right)-\mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right)-\mathrm{D}\,\mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right)\cdot\left(\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)\right)\Big)\\
&\qquad-\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\mathcal{R}_\delta\Big)\cdot\Big(\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)\Big)-\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\cdot\mathcal{R}_\epsilon\Big)\cdot\Big(\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)\Big).
\end{aligned}\tag{2.66}$$

From (2.64) we can observe that $\mathcal{R}_\epsilon$ is of the form (2.25). Employing Lemma (2.26), we have

$$\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\mathcal{R}_\epsilon\Big)\cdot\Big(\mathrm{D}\,M\left(\hat{\mathbf{u}}_h\right)\Big)=0\tag{2.67}$$

Equation (2.67) can be employed to express the second term on the right in (2.66) as

$$\begin{aligned}
&\Big(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\mathcal{R}_\epsilon\Big)\cdot\Big(\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)\Big)\\
&\quad=\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\mathcal{R}_\epsilon\Big)\cdot\Big(\mathbf{U}-M(\mathbf{u})+M(\mathbf{u})-M\left(\hat{\mathbf{u}}_h\right)-\mathrm{D}\,M\left(\hat{\mathbf{u}}_h\right)\cdot\left(\mathbb{P}\left(M(\mathbf{u})-M\left(\hat{\mathbf{u}}_h\right)\right)\right)\Big)
\end{aligned}\tag{2.68}$$

$$\begin{aligned}
&\quad=\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\mathcal{R}_\epsilon\Big)\cdot\Big(\mathbf{U}-M(\mathbf{u})\Big)\\
&\qquad+\Big(\mathrm{D}^2\,H\left(M\left(\hat{\mathbf{u}}_h\right)\right)\mathcal{R}_\epsilon\Big)\cdot\Big(M(\mathbf{u})-M\left(\hat{\mathbf{u}}_h\right)-\mathrm{D}\,M\left(\hat{\mathbf{u}}_h\right)\cdot\left(\mathbb{P}\left(M(\mathbf{u})-M\left(\hat{\mathbf{u}}_h\right)\right)\right)\Big).
\end{aligned}\tag{2.69}$$

Altogether, we have

$$\partial_t H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)+\partial_x Q\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)+\frac{1}{\varepsilon}\mathfrak{D}\Big(\mathbf{U}|M\left(\mathbf{u}\right)\Big)=J_1^s+J_2^s+J_3^s.\tag{2.70}$$

where

$$J_1^s := -\Big(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \partial_x M\left(\hat{\mathbf{u}}_h\right) \Big) \cdot \Big(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right) - \mathrm{D}\,\mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \left(\mathbf{U} - M\left(\hat{\mathbf{u}}_h\right)\right) \Big), \tag{2.71}$$

$$J_2^s := -\Big(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \mathcal{R}_\delta \Big) \cdot \Big(\mathbf{U} - M\left(\hat{\mathbf{u}}_h\right) \Big), \tag{2.72}$$

$$\begin{aligned} J_3^s := &-\Big(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \mathcal{R}_\epsilon \Big) \cdot \Big(\mathbf{U} - M(\mathbf{u}) \Big) \\ &- \Big(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \mathcal{R}_\epsilon \Big) \cdot \Big(M(\mathbf{u}) - M\left(\hat{\mathbf{u}}_h\right) - \mathrm{D}\,M\left(\hat{\mathbf{u}}_h\right) \cdot \left(\mathbb{P}\left(M(\mathbf{u}) - M\left(\hat{\mathbf{u}}_h\right)\right)\right) \Big). \end{aligned} \tag{2.73}$$

Complex System

Next, we proceed in a similar fashion to derive a relative entropy identity for the complex system. When we employ the complex system, we wish to compare $\mathbf{U}$ as defined in Hypothesis 2.5.5 to (2.1) and $\hat{\mathbf{U}}_h$ satisfying (2.47) as defined in Hypothesis 2.5.2. To this end, we derive a relative entropy identity for $H(\mathbf{U}|\hat{\mathbf{U}}_h)$.

We have

$$\begin{aligned} &\partial_t H(\mathbf{U}|\hat{\mathbf{U}}_h) + \partial_x Q(\mathbf{U}|\hat{\mathbf{U}}_h) \\ &= \Big(\partial_t H\left(\mathbf{U}\right) + \partial_x Q(\mathbf{U})\Big) - \Big(\partial_t H(\hat{\mathbf{U}}_h) + \partial_x Q(\hat{\mathbf{U}}_h)\Big) \\ &\qquad - \partial_t\Big(\mathrm{D}\,H(\hat{\mathbf{U}}_h) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h\Big) \Big) - \partial_x\Big(\mathrm{D}\,H(\hat{\mathbf{U}}_h) \cdot \Big(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}(\hat{\mathbf{U}}_h)\Big) \Big) \\ &= \frac{1}{\varepsilon} \mathrm{D}\,H\left(\mathbf{U}\right) \cdot \mathbf{R}(\mathbf{U}) - \frac{1}{\varepsilon} \mathrm{D}\,H(\hat{\mathbf{U}}_h) \cdot \mathbf{R}(\mathbf{U}) \\ &\qquad - \partial_t\Big(\mathrm{D}\,H(\hat{\mathbf{U}}_h)\Big) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h\Big) - \partial_x\Big(\mathrm{D}\,H(\hat{\mathbf{U}}_h)\Big) \cdot \Big(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}(\hat{\mathbf{U}}_h)\Big) \\ &= -\Big(\mathrm{D}^2 H(\hat{\mathbf{U}}_h)\partial_t \hat{\mathbf{U}}_h\Big) \cdot (\mathbf{U} - \hat{\mathbf{U}}_h) - \Big(\mathrm{D}^2 H(\hat{\mathbf{U}}_h)\partial_x \hat{\mathbf{U}}_h\Big) \cdot \Big(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}(\hat{\mathbf{U}}_h)\Big) \\ &\qquad + \frac{1}{\varepsilon}\Big(\mathrm{D}\,H\left(\mathbf{U}\right) - \mathrm{D}\,H(\hat{\mathbf{U}}_h)\Big) \cdot \mathbf{R}(\mathbf{U}) \\ &= -\Big(\mathrm{D}^2 H(\hat{\mathbf{U}}_h)\partial_x \hat{\mathbf{U}}_h\Big) \cdot \Big(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}(\hat{\mathbf{U}}_h) - \mathrm{D}\,\mathbf{F}(\hat{\mathbf{U}}_h) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h\Big) \Big) \\ &\qquad + \Big(\mathrm{D}\,H\left(\mathbf{U}\right) - \mathrm{D}\,H(\hat{\mathbf{U}}_h)\Big) \cdot \frac{1}{\varepsilon}\mathbf{R}(\mathbf{U}) - \left(\mathrm{D}^2 H(\hat{\mathbf{U}}_h) \left(\frac{1}{\varepsilon}\mathbf{R}(\hat{\mathbf{U}}_h) + \mathcal{R}_c\right) \right) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h\Big) \end{aligned}$$

where we have employed (2.51) along with the commutativity property (2.7).

Rearranging the terms, we have

$$\begin{aligned}
&\partial_t H(\mathbf{U}|\hat{\mathbf{U}}_h) + \partial_x Q(\mathbf{U}|\hat{\mathbf{U}}_h) \\
&\quad = -\Big(\mathrm{D}^2 H(\hat{\mathbf{U}}_h) \cdot \partial_x \hat{\mathbf{U}}_h \Big) \cdot \Big(\mathbf{F}(\mathbf{U}) - \mathbf{F}(\hat{\mathbf{U}}_h) - \mathrm{D}\,\mathbf{F}(\hat{\mathbf{U}}_h) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h \Big) \Big) \\
&\quad - \Big(\mathrm{D}^2 H(\hat{\mathbf{U}}_h) \mathcal{R}_c \Big) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h \Big) \\
&\quad + \Big(\mathrm{D}\,H(\mathbf{U}) - \mathrm{D}\,H(\hat{\mathbf{U}}_h) - \mathrm{D}^2 H(\hat{\mathbf{U}}_h) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h \Big) \Big) \cdot \frac{1}{\varepsilon} \mathbf{R}(\hat{\mathbf{U}}_h) \\
&\quad + \frac{1}{\varepsilon} \Big(\mathrm{D}\,H(\mathbf{U}) - \mathrm{D}\,H(\hat{\mathbf{U}}_h) \Big) \cdot \Big(\mathbf{R}(\mathbf{U}) - \mathbf{R}(\hat{\mathbf{U}}_h) \Big)
\end{aligned}$$

Employing (2.39) to express the last term to the right as $\frac{1}{\varepsilon}\mathfrak{D}\Big(\mathbf{U}|\hat{\mathbf{U}}_h\Big)$, we have

$$\partial_t H(\mathbf{U}|\hat{\mathbf{U}}_h) + \partial_x Q(\mathbf{U}|\hat{\mathbf{U}}_h) + \frac{1}{\varepsilon}\mathfrak{D}\left(\mathbf{U}|\hat{\mathbf{U}}_h\right) = J_1^c + J_2^c + J_3^c \tag{2.74}$$

where

$$J_1^c := -\Big(\mathrm{D}^2 H(\hat{\mathbf{U}}_h) \cdot \partial_x \hat{\mathbf{U}}_h \Big) \cdot \Big(\mathbf{F}(\mathbf{U}) - \mathbf{F}(\hat{\mathbf{U}}_h) - \mathrm{D}\,\mathbf{F}(\hat{\mathbf{U}}_h) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h \Big) \Big), \tag{2.75}$$

$$J_2^c := -\Big(\mathrm{D}^2 H(\hat{\mathbf{U}}_h) \cdot \mathcal{R}_c \Big) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h \Big), \tag{2.76}$$

$$J_3^c := \Big(\mathrm{D}\,H(\mathbf{U}) - \mathrm{D}\,H(\hat{\mathbf{U}}_h) - \mathrm{D}^2 H(\hat{\mathbf{U}}_h) \cdot \Big(\mathbf{U} - \hat{\mathbf{U}}_h \Big) \Big) \cdot \frac{1}{\varepsilon} \mathbf{R}(\hat{\mathbf{U}}_h). \tag{2.77}$$

2.5.6 Error estimates

Grönwall's lemma and Young's inequality for products (for exponent 2) is frequently employed in the error analysis. We state these next for the sake of completeness.

Lemma 2.5.9. *Let $T > 0$ and $\phi(t) \in C^0([0,T])$ and $a(t), b(t) \in L^1([0,T])$ be non-negative functions with b being a non-decreasing function and satisfying*

$$\phi(t) \leq \int_0^t a(s)\phi(s)\mathrm{d}s + b(t), \quad \forall\ t \in [0,T] \tag{2.78}$$

then

$$\phi(t) \leq b(t) \cdot \exp\left(\int_0^t a(s)\mathrm{d}s \right), \quad \forall\ t \in [0,T]. \tag{2.79}$$

Lemma 2.5.10. *Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}^N$ and $\alpha > 0$, then for the scalar product between the vectors*

$$\mathbf{a} * \mathbf{b} \leq \frac{|a|^2}{2\alpha} + \frac{\alpha |b|^2}{2} \tag{2.80}$$

holds.

Remark 2.5.11. Let $f : \Omega \times [0,T] \to \mathbb{R}$, then the L^2 and the L^∞ norms are denoted by

$$\|f\|_{L^2} := \left(\int_0^T \int_\Omega f^2(z,t)\,\mathrm{d}z\,\mathrm{d}t \right)^{\frac{1}{2}},$$

$$\|f\|_{L^\infty} := \inf \{\, C \geq 0 : |f(z,t)| \leq C \quad \text{for} \quad z \in \Omega, t \in [0,T] \quad a.e \,\}.$$

respectively.

Next, we derive the error estimates for the model hierarchy under consideration.

Theorem 2.5.12. *Let* $\mathbf{U}$ *be a smooth solution as defined in Hypothesis 2.5.5 and* $\hat{\mathbf{U}}_h$ *and* $\hat{\mathbf{u}}_h$ *be Lipschitz continuous reconstructions as defined in Hypothesis 2.5.2 on* Ω_c *and* Ω_s *respectively. Furthermore, Let* $\mathbf{U}$ *and* $\hat{\mathbf{U}}_h, \hat{\mathbf{u}}_h$ *take values only in* $\mathcal{D}$*. Then for* $0 \leq t \leq T$*, the following error estimate holds*

$$\begin{aligned} &\int_{\Omega_c} \left| \mathbf{U}(\cdot,t) - \hat{\mathbf{U}}_h(\cdot,t) \right|^2 \mathrm{d}x + \int_{\Omega_s} \left| \mathbf{U}(\cdot,t) - M(\hat{\mathbf{u}}_h(\cdot,t)) \right|^2 \mathrm{d}x \\ &\qquad \leq \frac{1}{C_{\underline{H}}} \left(I + D_c + D_s + \mathcal{M}_s \right) \exp\left(\frac{1}{C_{\underline{H}}} \max\left(G_c, G_s\right) t \right), \end{aligned} \tag{2.81}$$

where

$$I = \int_{\Omega_s} H(\mathbf{U}|M(\hat{\mathbf{u}}_h))\Big|_{t=0} \mathrm{d}x + \int_{\Omega_c} H(\mathbf{U}|\hat{\mathbf{U}}_h)\Big|_{t=0} \mathrm{d}x,$$

$$D_c = \frac{1}{2} \int_0^t \int_{\Omega_c} |\,\mathrm{D}^2 H(\hat{\mathbf{U}}_h) \cdot \mathcal{R}_c|^2 \,\mathrm{d}x\,\mathrm{d}\tau,$$

$$D_s = \frac{1}{2} \int_0^t \int_{\Omega_s} |\,\mathrm{D}^2 \eta(\hat{\mathbf{u}}_h) \cdot \mathcal{R}_\delta|^2 \,\mathrm{d}x\,\mathrm{d}\tau,$$

$$\mathcal{M}_s = \frac{\varepsilon}{\nu} \int_0^t \int_{\Omega_s} |\,\mathrm{D}^2 \eta(\hat{\mathbf{u}}_h) \cdot \mathcal{R}_\epsilon|^2 \,\mathrm{d}x\,\mathrm{d}\tau,$$

$$G_c = \frac{1}{2} + C_{\bar{F}} C_{\bar{H}} \|\partial_x \hat{\mathbf{U}}_h\|_{L^\infty} + C_{\bar{H}} \left(\left\| \frac{1}{\varepsilon} \mathbf{R}(\hat{\mathbf{U}}_h) \right\|_{L^\infty} + \left\| \frac{1}{\varepsilon} \,\mathrm{D}\,\mathbf{R}(\hat{\mathbf{U}}_h) \right\|_{L^\infty} \right),$$

$$G_s = \frac{1}{2} + C_{\bar{F}} C_{\bar{H}} \|\partial_x M\left(\hat{\mathbf{u}}_h\right)\|_{L^\infty} + C_{\bar{H}} C_{\bar{M}} |\mathbb{P}|^2 \|\mathcal{R}_\epsilon\|_{L^\infty}.$$

Proof. Integrating (2.74) and (2.70) on Ω_c and Ω_s, we have

$$\begin{aligned} &\int_0^t \int_{\Omega_c} \partial_t H\left(\mathbf{U}|\hat{\mathbf{U}}_h\right) + \partial_x Q(\mathbf{U}|\hat{\mathbf{U}}_h) + \frac{1}{\varepsilon} \mathfrak{D}\left(\mathbf{U}|\hat{\mathbf{U}}_h\right) \mathrm{d}x\,\mathrm{d}\tau \\ &\quad + \int_0^t \int_{\Omega_s} \partial_t H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right) + \partial_x Q(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)) + \frac{1}{\varepsilon} \mathfrak{D}\Big(\mathbf{U}|M\left(\mathbf{u}\right)\Big) \,\mathrm{d}x\,\mathrm{d}\tau \\ &\qquad = \int_0^t \int_{\Omega_s} J_1^s + J_2^s + J_3^s \,\mathrm{d}x\,\mathrm{d}\tau + \int_0^t \int_{\Omega_c} J_1^s + J_2^c + J_3^c \,\mathrm{d}x\,\mathrm{d}\tau, \end{aligned}$$

which simplifies to

$$\begin{aligned}\int_{\Omega_s} H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)\mathrm{dx} + \int_{\Omega_c} H\left(\mathbf{U}|\hat{\mathbf{U}}_h\right)\mathrm{dx} + \int_0^t\int_{\Omega_s}\frac{1}{\varepsilon}\mathfrak{D}\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)\mathrm{dx}\,\mathrm{d}\tau + \int_0^t\int_{\Omega_c}\frac{1}{\varepsilon}\mathfrak{D}\left(\mathbf{U}|\hat{\mathbf{U}}_h\right)\mathrm{dx}\,\mathrm{d}\tau \\ = \int_{\Omega_s} H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)\Big|_{t=0}\mathrm{dx} + \int_{\Omega_c} H\left(\mathbf{U}|\hat{\mathbf{U}}_h\right)\Big|_{t=0}\mathrm{dx} \\ + \sum_{x_\gamma\in\Gamma}\int_0^t Q\left(\mathbf{U}\left(x_\gamma,\cdot\right)|M\left(\hat{\mathbf{u}}_h\left(x_\gamma,\cdot\right)\right)\right) - Q\left(\mathbf{U}\left(x_\gamma,\cdot\right)|\hat{\mathbf{U}}_h\left(x_\gamma,\cdot\right)\right)\mathrm{d}\tau \\ + \int_0^t\int_{\Omega_s} J_1^s + J_2^s + J_3^s\,\mathrm{dx}\,\mathrm{d}\tau + \int_0^t\int_{\Omega_c} J_1^s + J_2^c + J_3^c\,\mathrm{dx}\,\mathrm{d}\tau\,.\end{aligned}$$

From the relative entropy dissipation condition (2.42), we get

$$\begin{aligned}\int_{\Omega_s} H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)\mathrm{dx} + \int_{\Omega_c} H\left(\mathbf{U}|\hat{\mathbf{U}}_h\right)\mathrm{dx} + \frac{\nu}{\varepsilon}\int_0^t\int_{\Omega_s}|\mathbf{U}-M(\mathbf{u})|^2\,\mathrm{dx}\,\mathrm{d}\tau \\ \leq \int_{\Omega_s} H\left(\mathbf{U}|M\left(\hat{\mathbf{u}}_h\right)\right)\Big|_{t=0}dx + \int_{\Omega_c} H\left(\mathbf{U}|\hat{\mathbf{U}}_h\right)\Big|_{t=0}\mathrm{dx} \\ + \left|\int_0^t\int_{\Omega_c}\frac{1}{\varepsilon}\mathfrak{D}\left(\mathbf{U}|\hat{\mathbf{U}}_h\right)\mathrm{dx}\,\mathrm{d}\tau\right| \\ + \int_0^t\int_{\Omega_s} J_1^s + J_2^s + J_3^s\,\mathrm{dx}\,\mathrm{d}\tau \\ + \int_0^t\int_{\Omega_c} J_1^s + J_2^c + J_3^c\,\mathrm{dx}\,\mathrm{d}\tau \\ + \sum_{x_\gamma\in\Gamma}\int_0^t Q\left(\mathbf{U}\left(x_\gamma,\cdot\right)|M\left(\hat{\mathbf{u}}_h\left(x_\gamma,\cdot\right)\right)\right) - Q\left(\mathbf{U}\left(x_\gamma,\cdot\right)|\hat{\mathbf{U}}_h\left(x_\gamma,\cdot\right)\right)\mathrm{d}\tau,\end{aligned}$$

Note that for $x_\gamma \in \Gamma$

$$\begin{aligned}&\int_0^t Q\left(\mathbf{U}\left(x_\gamma,\cdot\right)|M\left(\hat{\mathbf{u}}_h\left(x_\gamma,\cdot\right)\right)\right) - Q\left(\mathbf{U}\left(x_\gamma,\cdot\right)|\hat{\mathbf{U}}_h\left(x_\gamma,\cdot\right)\right)\mathrm{d}\tau \\ &\quad= \int_0^t Q\left(\hat{\mathbf{U}}_h\left(x_\gamma,\cdot\right)\right) - Q\left(M\left(\hat{\mathbf{u}}_h\left(x_\gamma,\cdot\right)\right)\right)\mathrm{d}\tau \\ &\quad+ \int_0^t\left(\mathrm{D}\,H\left(\hat{\mathbf{U}}_h\left(x_\gamma,\cdot\right)\right) - \mathrm{D}\,H\left(M\left(\hat{\mathbf{u}}_h\left(x_\gamma,\cdot\right)\right)\right)\right)\cdot\mathbf{F}\left(\mathbf{U}\left(x_\gamma,\cdot\right)\right)\mathrm{d}\tau \\ &\quad- \int_0^t \mathrm{D}\,H\left(\hat{\mathbf{U}}_h\left(x_\gamma,\cdot\right)\right)\cdot\mathbf{F}\left(\hat{\mathbf{U}}_h\left(x_\gamma,\cdot\right)\right) - \mathrm{D}\,H\left(M\left(\hat{\mathbf{u}}_h\left(x_\gamma,\cdot\right)\right)\right)\cdot\mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\left(x_\gamma,\cdot\right)\right)\right)\mathrm{d}\tau\end{aligned} \tag{2.82}$$

Since the reconstruction satisfies the condition (2.46) the term above vanishes. We can observe that (2.82) would not be computable if the reconstruction would not satisfy condition (2.46) since it would require knowledge about the value of the flux for the exact solution at the interface.

Next, we derive estimates for the terms J_1^s, J_2^s, J_3^s and J_1^c, J_2^c, J_3^c.

For J_1^s, employing (2.55), (2.56) we have

$$\begin{aligned}
&\left| \int_0^t \int_{\Omega_s} J_1^s \,\mathrm{d}x\,\mathrm{d}\tau \right| \\
&= \left| \int_0^t \int_{\Omega_s} \left(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \partial_x M\left(\hat{\mathbf{u}}_h\right) \right) \cdot \left(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right) - \mathrm{D}\,\mathbf{F}\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \left(\mathbf{U} - M\left(\hat{\mathbf{u}}_h\right)\right) \right) \mathrm{d}x\,\mathrm{d}\tau \right| \\
&\leq C_{\bar{F}} C_{\bar{H}} \|\partial_x M\left(\hat{\mathbf{u}}_h\right)\|_{L^\infty} \int_0^t \int_{\Omega_s} |\mathbf{U} - M\left(\hat{\mathbf{u}}_h\right)|^2 \,\mathrm{d}x\,\mathrm{d}\tau\,.
\end{aligned}$$

Next, employing Young's inequality of products and (2.56) for J_2^s we have

$$\begin{aligned}
\left| \int_0^t \int_{\Omega_s} J_2^s \,\mathrm{d}x\,\mathrm{d}\tau \right| &= \left| \int_0^t \int_{\Omega_s} \left(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \mathcal{R}_\delta \right) \cdot \left(\mathbf{U} - M\left(\hat{\mathbf{u}}_h\right)\right) \mathrm{d}x\,\mathrm{d}\tau \right| \\
&\leq \frac{1}{2} \int_0^t \int_{\Omega_s} |\,\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \mathcal{R}_\delta|^2 \,\mathrm{d}x\,\mathrm{d}\tau + \frac{1}{2} \int_0^t \int_{\Omega_s} |\mathbf{U} - M\left(\hat{\mathbf{u}}_h\right)|^2 \,\mathrm{d}x\,\mathrm{d}\tau\,.
\end{aligned}$$

Next, we derive an error estimate for J_3^s

$$\begin{aligned}
\left| \int_0^t \int_{\Omega_s} J_3^s \,\mathrm{d}x\,\mathrm{d}\tau \right| &\leq \left| \int_0^t \int_{\Omega_s} \left(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \mathcal{R}_\epsilon \right) \cdot \left(\mathbf{U} - M(\mathbf{u})\right) \mathrm{d}x\,\mathrm{d}\tau \right| \\
&+ \left| \int_0^t \int_{\Omega_s} \left(\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \mathcal{R}_\epsilon \right) \cdot \left(M(\mathbf{u}) - M\left(\hat{\mathbf{u}}_h\right) - \mathrm{D}\,M\left(\hat{\mathbf{u}}_h\right) \cdot \mathbb{P}\left(M(\mathbf{u}) - M\left(\hat{\mathbf{u}}_h\right)\right)\right) \mathrm{d}x\,\mathrm{d}\tau \right|.
\end{aligned}$$

To derive an upper bound for J_3^s we can employ Young's inequality of products for the first term on the right. For the second term on the right employing (2.56),(2.58) and the fact that $\mathbb{P}M(\mathbf{u}) = \mathbb{P}\mathbf{U}$, we have

$$\begin{aligned}
\left| \int_0^t \int_{\Omega_s} J_3^s \,\mathrm{d}x\,\mathrm{d}\tau \right| &\leq \frac{\varepsilon}{\nu} \int_0^t \int_{\Omega_s} |\,\mathrm{D}^2 H\left(M\left(\hat{\mathbf{u}}_h\right)\right) \cdot \mathcal{R}_\epsilon|^2 \,\mathrm{d}x\,\mathrm{d}\tau \\
&+ \frac{\nu}{\varepsilon} \int_0^t \int_{\Omega_s} |\mathbf{U} - M(\mathbf{u})|^2 \,\mathrm{d}x\,\mathrm{d}\tau \\
&+ C_{\bar{H}} C_{\bar{M}} |\mathbb{P}|^2 \|\mathcal{R}_\epsilon\|_{L^\infty} \int_0^t \int_{\Omega_s} |\mathbf{U} - M\left(\hat{\mathbf{u}}_h\right)|^2 \,\mathrm{d}x\,\mathrm{d}\tau\,.
\end{aligned}$$

We can observe that the introduction of the dissipation term $\frac{1}{\varepsilon}\mathfrak{D}\Big(\mathbf{U}|M\left(\mathbf{u}\right)\Big)$ and introducing zeros in (2.66)-(2.68) allows to introduce the scaling by ε in the error estimates.

We proceed similarly to bound the terms arising in Ω_c where the complex system is employed.

For J_1^c, we have

$$\begin{aligned}
\left| \int_0^t \int_{\Omega_c} J_1^c \,\mathrm{d}x\,\mathrm{d}\tau \right| &= \left| \int_0^t \int_{\Omega_c} \left(\mathrm{D}^2 H(\hat{\mathbf{U}}_h) \cdot \partial_x \hat{\mathbf{U}}_h \right) \cdot \left(\mathbf{F}\left(\mathbf{U}\right) - \mathbf{F}(\hat{\mathbf{U}}_h) - \mathrm{D}\,\mathbf{F}(\hat{\mathbf{U}}_h) \cdot \left(\mathbf{U} - \hat{\mathbf{U}}_h\right) \right) \mathrm{d}x\,\mathrm{d}\tau \right| \\
&\leq C_{\bar{F}} C_{\bar{H}} \|\partial_x \hat{\mathbf{U}}_h\|_{L^\infty} \int_0^t \int_{\Omega_c} |\mathbf{U} - \hat{\mathbf{U}}_h|^2 \,\mathrm{d}x\,\mathrm{d}\tau\,.
\end{aligned}$$

Next employing Young's inequality of products for J_2^c, we have

$$\begin{aligned}\left|\int_0^t\int_{\Omega_c} J_2^c \,\mathrm{dx}\,\mathrm{d}\tau\right| &= \left|\int_0^t\int_{\Omega_c}\left(\mathrm{D}^2 H(\hat{\mathbf{U}}_h)\cdot\mathcal{R}_c\right)\left(\mathbf{U}-\hat{\mathbf{U}}_h\right)\mathrm{dx}\,\mathrm{d}\tau\right| \\ &\leq \frac{1}{2}\int_0^t\int_{\Omega_c}|\,\mathrm{D}^2 H(\hat{\mathbf{U}}_h)\cdot\mathcal{R}_c|^2\,\mathrm{dx}\,\mathrm{d}\tau+\frac{1}{2}\int_0^t\int_{\Omega_c}|\mathbf{U}-\hat{\mathbf{U}}_h|^2\,\mathrm{dx}\,\mathrm{d}\tau\,.\end{aligned}$$

For J_3^c, employing (2.57) we have

$$\begin{aligned}\left|\int_0^t\int_{\Omega_c} J_3^c \,\mathrm{dx}\,\mathrm{d}\tau\right| &= \left|\int_0^t\int_{\Omega_c}\left(\mathrm{D}\,H\,(\mathbf{U})-\mathrm{D}\,H(\hat{\mathbf{U}}_h)-\mathrm{D}^2 H(\hat{\mathbf{U}}_h)\cdot\left(\mathbf{U}-\hat{\mathbf{U}}_h\right)\right)\cdot\frac{1}{\varepsilon}\mathbf{R}(\hat{\mathbf{U}}_h)\,\mathrm{dx}\,\mathrm{d}\tau\right| \\ &\leq C_{\bar{H}}\left\|\frac{1}{\varepsilon}\mathbf{R}(\hat{\mathbf{U}}_h)\right\|_{L^\infty}\int_0^t\int_{\Omega_c}|\mathbf{U}-\hat{\mathbf{U}}_h|^2\,\mathrm{dx}\,\mathrm{d}\tau\,.\end{aligned}$$

Finally

$$\left|\int_0^t\int_{\Omega_c}\frac{1}{\varepsilon}\mathfrak{D}\left(\mathbf{U}|\hat{\mathbf{U}}_h\right)\mathrm{dx}\,\mathrm{d}\tau\right| \leq C_{\bar{H}}\left\|\frac{1}{\varepsilon}\,\mathrm{D}\,\mathbf{R}(\hat{\mathbf{U}}_h)\right\|_{L^\infty}\int_0^t\int_{\Omega_c}|\mathbf{U}-\hat{\mathbf{U}}_h|^2\,\mathrm{dx}\,\mathrm{d}\tau\,. \tag{2.83}$$

Using (2.56) we have

$$\begin{aligned}&\int_{\Omega_c}|\mathbf{U}-\hat{\mathbf{U}}_h|^2\,\mathrm{dx}+\int_{\Omega_s}|\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)|^2\,\mathrm{dx} \\ &\quad\leq\frac{1}{C_{\underline{H}}}\left(I+M_s+D_c+D_s\right) \\ &\quad+\frac{1}{C_{\underline{H}}}\max\left(Gc,Gs\right)\left(\int_0^t\int_{\Omega_c}|\mathbf{U}-\hat{\mathbf{U}}_h|^2\,\mathrm{dx}\,\mathrm{d}\tau+\int_0^t\int_{\Omega_s}|\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)|^2\,\mathrm{dx}\,\mathrm{d}\tau\right).\end{aligned}$$

Using Grönwall's lemma we obtain

$$\int_{\Omega_c}|\mathbf{U}-\hat{\mathbf{U}}_h|^2\,\mathrm{dx}+\int_{\Omega_s}|\mathbf{U}-M\left(\hat{\mathbf{u}}_h\right)|^2\,\mathrm{dx}\leq\frac{1}{C_{\underline{H}}}\left(I+\mathcal{M}_s+D_c+D_s\right)\cdot\exp\left(\frac{1}{C_{\underline{H}}}\max\left(G_c,G_s\right)t\right).$$

□

Chapter 3

Numerical Methods

In this chapter, the numerical methods employed to carry out the mesh and model adaptation simulations are discussed.

Outline

The outline of this chapter is as follows

Section 3.1: In this section, the spatial semi-discretization employing Discontinuous Galerkin method is discussed. Since numerical solutions to hyperbolic conservation laws may develop numerical oscillations near discontinuities, the numerical solution needs to be limited. The limiting method we employ is introduced.

Section 3.2: We need to compute a Lipschitz reconstruction from the numerical solution to be able to employ the relative entropy stability framework. To this end, a method to reconstruct Runge Kutta Discontinuous Galerkin solutions is discussed.

Section 3.3: In this section, the notations employed for mesh and model adaptive numerical solutions are discussed. Model adaptation requires that we convert the solution from one system to the other. Algorithms to do so are described next. Furthermore, limiting and reconstruction of the mesh and model adaptive solution is discussed.

Section 3.4: In this section, a mesh and model adaptation strategy based on the error analysis done in the previous chapter is proposed.

3.1 Semi-discrete scheme

We employ the Runge Kutta Discontinuous Galerkin Method (RKDG), i.e. the spatial discretization is done using the Discontinuous Galerkin method (DG) and the time stepping is done by solving the resulting system of ordinary differential equations by employing a strong stability preserving Runge-Kutta (SSP-RK) scheme. In the following section, semi-discretization employing modal Discontinuous Galerkin (DG) method on a uniform mesh is discussed. For more details on DG methods the reader is referred to text books such as [56] and [79] and for details on SSP-RK the reader is referred to [20],[22].

3.1.1 Discontinuous Galerkin Method

We discuss the semi-discretization for the complex system. The simple system can be semi-discretized in a similar fashion.

We seek to approximate solutions to

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = \frac{1}{\varepsilon}\mathbf{R}(\mathbf{U}), \quad \mathbf{U} : \Omega \times [0,T] \to \mathbb{R}^N \tag{3.1}$$

for some $T > 0$, subject to initial condition $\mathbf{U}_0 \in L^\infty(\Omega, \mathbb{R}^N)$ on the computational domain $\Omega := [a,b]$ with periodic boundary conditions.

The solution is approximated at times $0 = t^0 < t^1 < \ldots t^{\mathfrak{N}} = T$ employing an equidistant mesh with $\mathcal{N}_E$ number of cells. The cells in the equidistant mesh are given by

$$\mathcal{V}_k := (x_k, x_{k+1}), \quad k \in \mathcal{I} := \{1, \ldots, \mathcal{N}_E\}, \tag{3.2}$$

where

$$x_k := a + kh, \; h := \frac{b-a}{\mathcal{N}_E}. \tag{3.3}$$

The solution to (3.1) is approximated by piece-wise continuous polynomials of degree p. To this end, we introduce the space

$$S_x^{p,N}(\Omega) := \{\mathbf{v} : \Omega \to \mathbb{R}^N : \mathbf{v}|_{\mathcal{V}_k} \in \Pi_p(\mathcal{V}_k, \mathbb{R}^N) \;\; \forall \; k \;\; \in \mathcal{I}\} \tag{3.4}$$

where $\Pi_p(\mathcal{V}_k, \mathbb{R}^N)$ is the space of polynomials of degree less than or equal to p.

In order to approximate the solution in space $S_x^{p,N}(\Omega)$ we make use of orthonormal set of basis functions as defined below.

Definition 3.1.1. *Let*

$$\Phi := \{\phi_i^k, \; k \in \mathcal{I}, \; i = 0, \ldots, p\} \tag{3.5}$$

be a set of basis functions spanning $S_x^{p,N}(\Omega)$ such that

$$\langle \phi_i^k, \phi_{i'}^{k'} \rangle = \delta_{ii'}\delta_{kk'} \;\; \textit{and} \;\; \textit{supp}(\phi_i^k) = \mathcal{V}_k. \tag{3.6}$$

Let $\mathbf{U}_h \in S_x^{p,N}(\Omega)$ be a numerical solution such that

$$\mathbf{U}_h(x,t) := \sum_{k=1}^{\mathcal{N}_E} \sum_{i=0}^{p} \mathbf{U}_i^k(t)\phi_i^k(x), \quad \forall \; x \in \Omega, \; t \in \{t^{\mathfrak{n}}\}_{n=0}^{\mathfrak{N}}, \tag{3.7}$$

where the coefficients are obtained by

$$\mathbf{U}_i^k(t) = \int_{\mathcal{V}_i} \mathbf{U}_h(x,t)\phi_i^k(x)\,\mathrm{dx}, \quad \forall \; k \in \mathcal{I}, \forall \; i = 0, \ldots, p. \tag{3.8}$$

The semi-discrete numerical scheme, i.e. the time evolution of the coefficients is derived by multiplying (3.1) by ϕ_i^k and integrating by parts over Ω. Then, the orthogonality properties (3.6) can be employed to give

$$\begin{aligned}\frac{\mathrm{d}\,\mathbf{U}_i^k(t)}{\mathrm{d}\tau} &= -\left(\mathbf{F}_h^{k+1}(t)\phi_i^k(x_{k+1}^-) - \mathbf{F}_h^k(t)\phi_i^k(x_k^+)\right) \\ &+ \int_{\mathcal{V}_k} \mathbf{F}\left(\mathbf{U}_h\left(x,t\right)\right)\frac{d\phi_i^k(x)}{dx} + \frac{1}{\varepsilon}\mathbf{R}\left(\mathbf{U}_h\left(x,t\right)\right)\phi_i^k(x)\,\mathrm{dx}, \quad \forall\ k\ \in \mathcal{I}, i = 0,\ldots,p,\end{aligned} \tag{3.9}$$

where

$$x_k^{\pm} = \lim_{s\to 0^+} x \pm s, \tag{3.10}$$

and $\mathbf{F}_h^k$ is a consistent and Lipschitz continuous numerical flux of the form

$$\mathbf{F}_h^k(t) = \mathbf{F}_h\left(\mathbf{U}_h^{k-1}(x_k^-,t), \mathbf{U}_h^k(x_k^+,t)\right). \tag{3.11}$$

In particular, we employ the Local Lax-Friedrichs numerical flux given by

$$\mathbf{F}_{LLF}^k(t) := \frac{1}{2}\left(\mathbf{F}\left(\mathbf{U}_h^{k-1}\left(x_k^-,t\right) + \mathbf{F}\left(\mathbf{U}_h^k(x_k^+,t\right)\right)\right) - \frac{1}{2}\lambda_{max}\left(\mathbf{U}_h^k\left(x_k^+,t\right) - \mathbf{U}_h^{k-1}\left(x_k^-,t\right)\right), \tag{3.12}$$

where

$$\lambda_{max} = \max\left(\max_{i\in 1,\ldots,N}\left(\left|\lambda_i\left(\mathbf{U}_h^k(x_k^+,t)\right)\right|\right), \max_{i\in 1,\ldots,N}\left(\left|\lambda_i\left(\mathbf{U}_h^{k-1}(x_{k-1}^-,t)\right)\right|\right)\right) \tag{3.13}$$

and $\lambda_i\left(\mathbf{U}\right), i = 1,\ldots,N$ are the eigenvalues of $\mathrm{D}\,\mathbf{F}\left(\mathbf{U}\right)$.

Other types of numerical fluxes such as approximate Riemann solvers, like Roe's Riemann solver are available but are more computationally expensive. Hence, we restrict ourselves to Local Lax Friedrichs flux. See [53], [84] and [91] for more details.

We employ an explicit strong stability preserving Runge-Kutta method of order $q = p+1$ to step in time. For more information about SSP-RK schemes in the context of hyperbolic conservation laws the reader is referred to [21], [50] and [85].

Remark 3.1.2. When employing explicit time discretization, there is a limitation on the size of the time step Δt employed in the form of Courant-Friedrichs-Levy (CFL) condition [19],[50]. The CFL condition can be expressed as

$$\Delta t \leq c \cdot \min\left(\frac{\Delta x_{\min}}{\lambda_f}, \frac{1}{\lambda_s}\right), \tag{3.14}$$

where $\Delta x_{\min}$ is the global minimum mesh width and λ_f and λ_s is the spectral radius of the Jacobian of the flux function and the Jacobian of the source term respectively. The factor c depends on the SSP-RK scheme employed [50].

The following restriction is is sufficient in most cases [14],[21]

$$\Delta t \leq \frac{1}{2p+1} \cdot \max\left(\frac{\Delta x_{\min}}{\lambda_f}, \frac{1}{\lambda_s}\right). \tag{3.15}$$

Remark 3.1.3. For the sake of convenience, for $\mathfrak{n} = 0, \ldots, \mathfrak{N}$, the numerical solution at time $t^{\mathfrak{n}}$ is denoted by $\mathbf{U}_h^{\mathfrak{n}}$ where $\mathbf{U}_h^{\mathfrak{n}} = \mathbf{U}_h(\cdot, t^{\mathfrak{n}})$ and the numerical solution in cell $\mathcal{V}_i, i \in \mathcal{I}$ at time $t^{\mathfrak{n}}$ by $\mathbf{U}_h^{\mathfrak{n},i} := \mathbf{U}_h^{\mathfrak{n}}|_{\mathcal{V}_i}$.

3.1.2 Limiting

Limiting is required when employing RKDG method for hyperbolic balance or conservation laws to suppress spurious numerical oscillations near discontinuities. Limiting generally consists of identifying so called troubled cells which are cells with significant oscillations. The numerical solution in these cells is locally corrected by limiting the slope of the numerical solution so as to remove the oscillations. Since the limiter purely modifies the higher order coefficients of the DG solution, it preserves the mean values in the cells, i.e. the zeroth order coefficients, hence maintaining conservation. The limiter we employ [36] is based on the Cockburn-Shu limiter developed for multidimensional systems of conservation laws [19],[22]. The Cockburn-Shu limiter consists of using a minmod function to identify troubled cells. Additionally, a constant $\mathfrak{m} > 0$ is employed in order to avoid limiting at critical points in regions where the solution is locally smooth. A review on limiters in the context of hyperbolic conservation laws can be found in [18], [81].

Before we discuss limiting, we recall some basic properties of hyperbolic systems of conservation laws, which are to be employed in the limiting procedure. More details on the properties of hyperbolic conservation laws can be found in text books such as [26], [47] and [62].

Definition 3.1.4. *The system of equations (3.1) is hyperbolic if* $\forall\ \mathbf{U} \in \mathbb{R}^N$ *the Jacobian of the flux function function has real eigenvalues and it possesses a complete set of left and right eigenvectors. Consequently, the Jacobian of the flux is diagonizable, allowing it to be expressed as*

$$\mathrm{D}\,\mathbf{F}(\mathbf{U}) = \mathfrak{R}(\mathbf{U})\Lambda(\mathbf{U})\mathfrak{L}(\mathbf{U}), \tag{3.16}$$

where

$$\mathfrak{L}^T(\mathbf{u}) := (\mathbf{l}_1(\mathbf{U}), \mathbf{l}_2(\mathbf{U}), \ldots, \mathbf{l}_N(\mathbf{U}))\,, \quad \mathfrak{R}(\mathbf{u}) := (\mathbf{r}_1(\mathbf{U}), \mathbf{r}_2(\mathbf{U}), \ldots, \mathbf{r}_N(\mathbf{U})) \tag{3.17}$$

are the matrices containing the left and the right eigenvectors, satisfying

$$\mathfrak{L}\mathfrak{R} = \mathrm{I} \tag{3.18}$$

and

$$\Lambda(\mathbf{U}) := diag\,(\lambda_1, \lambda_2, \ldots, \lambda_N) \tag{3.19}$$

is a diagonal matrix containing the eigenvalues.

Remark 3.1.5. We assume that the left and right eigenvectors are normalized such that

$$|\mathbf{l}_i| = |\mathbf{r}_i| = 1, \quad 1 \le i \le N. \tag{3.20}$$

Lemma 3.1.6. *Let the system of equations (3.1) be hyperbolic, then any vector* $\mathbf{U} \in \mathbb{R}^N$ *can be spanned by the eigenvectors such that*

$$\mathbf{U} = \sum_{i=1}^{N} \langle \mathbf{l}_i(\mathbf{U}), \mathbf{U} \rangle \mathbf{r}_i = \mathfrak{R}\mathbf{w} \tag{3.21}$$

where $\mathbf{w} := \mathfrak{L}\mathbf{U}$ *is the vector of local characteristic variables.*

With that in mind, the limiting of the numerical solution is done as follows.

Algorithm 1 Limiting

1: **procedure** $\mathtt{L}(\mathbf{U}_h^{\mathfrak{n},i}, \mathfrak{m})$
2: Locally transform the numerical solution in cell $\mathcal{V}_i$ to characteristic variables $\mathbf{w}_h^i := \mathfrak{L}(\mathbf{U}_0^i(t^{\mathfrak{n}}))\mathbf{U}_h^{\mathfrak{n},i}$.
3: Project the characteristic variables to $\Pi_1(\mathcal{V}_i, \mathbb{R}^N)$, i.e.

$$P_{\Pi_1}\mathbf{w}_h^i(x, t^{\mathfrak{n}}) = \mathbf{w}_0^i(t^{\mathfrak{n}}) + \mathbf{w}_1^i(t^{\mathfrak{n}}) \cdot \frac{(2x - (x_i + x_{i+1}))}{x_{i+1} - x_i}, \quad x \in \mathcal{V}_i \tag{3.22}$$

4: Let

$$\tilde{w}_1^{i,k} = m(w_1^{i,k}, w_0^{i+1,k} - w_0^{i,k}, w_0^{i,k} - w_0^{i-1,k}), \tag{3.23}$$

for $k = 1, \ldots, N$. Note that the superscript k denotes the k^{th} entry of the vector and m is the function

$$m(a, b, c) := \begin{cases} a, & \text{if } |a| \leq \mathfrak{m}\,(x_{k+1} - x_k)^2 \\ \tilde{m}(a, b, c), & \text{otherwise} \end{cases}, \tag{3.24}$$

where $\tilde{m}$ is the minmod function

$$\tilde{m}(a, b, c) := \begin{cases} \operatorname{sign}(a)\min(|a|, |b|, |c|), & \text{if } \operatorname{sign}(a) = \operatorname{sign}(b) = \operatorname{sign}(c) \\ 0, & \text{otherwise} \end{cases}. \tag{3.25}$$

5: If $\tilde{\mathbf{w}}_1^i \neq \mathbf{w}_1^i$ then set $\mathbf{w}_1^i = \tilde{\mathbf{w}}_1^i$.
6: Transform the characteristic variables $\mathbf{w}_h^i$ back to give the limited numerical solution $\bar{\mathbf{U}}_h^{\mathfrak{n},i} := \mathfrak{R}(\mathbf{U}_0^i(t^{\mathfrak{n}}))\mathbf{w}_h^i$.
7: **return** $\bar{\mathbf{U}}_h^{\mathfrak{n},i}$
8: **end procedure**

3.2 Reconstruction

We employ the relative entropy framework to derive the error estimates, which requires the quantity being compared to the exact solution to be Lipschitz continuous. The numerical solution in general will not have the necessary regularity. In this section, we discuss a method to compute such a space-time reconstruction from SSP-RK DG numerical solutions to hyperbolic balance laws. The reconstruction method was introduced by Giesselmann et al. in [40] and [42] in the context of a posteriori error analysis for hyperbolic conservation laws. We provide technical details of the reconstruction method and highlight some important properties, but do not provide an in depth analysis and discussion of the reconstruction method. The details can be found in the works cited above.

Before defining the reconstructions first we introduce a space of piecewise polynomial functions

$$S_t^q(t^{\mathfrak{n}}, t^{\mathfrak{n}+1}; \mathbb{V}) := \{\mathbf{v} : [t^{\mathfrak{n}}, t^{\mathfrak{n}+1}] \to \mathbb{V} : \mathbf{v} \in \Pi_q((t^{\mathfrak{n}}, t^{\mathfrak{n}+1}), \mathbb{V})\} \tag{3.26}$$

where $\mathbb{V}$ is some vector space.

The reconstruction of the numerical solution is done in two steps; first a reconstruction in time $\hat{\mathbf{U}}^t \in S_t^q(0, T; S_x^{p,N}(\Omega))$ is computed followed by a space-time reconstruction $\hat{\mathbf{U}}^{st}(\cdot, t) \in S_x^{p+1,N}(\Omega)$ for $t \in [t^{\mathfrak{n}}, t^{\mathfrak{n}+1}]$.

3.2.1 Reconstruction in time

The reconstruction in time is computed as a piecewise polynomial in time. The polynomial degree is chosen to be equal to the convergence order of the time stepping method. The reconstruction is computed employing a Hermite interpolation. To this end, the numerical solutions $\{\mathbf{U}_h^{\mathfrak{n}}\}_{\mathfrak{n}=0}^{\mathfrak{N}}$ at times $\{t^{\mathfrak{n}}\}_{\mathfrak{n}=0}^{\mathfrak{N}}$ are employed. The reconstruction does not depend on the type of time stepping method employed, which provides flexibility in the choice of time stepping method.

The reconstruction in time $\hat{\mathbf{U}}^t$ is defined by prescribing the values and the time derivatives at certain points in time. To this end, let $(m, d, r) \in \mathbb{N}_0^3$ where $\mathbb{N}_0$ is the set of all natural numbers and zero. A reconstruction in time such that the values and the first $d+1$ derivatives at $t^{\mathfrak{n}-m}, \ldots, t^{\mathfrak{n}}$ and the values and first $r+1$ derivatives at t^{n+1} are prescribed is denoted as a $\mathcal{H}(m, d, r)$ reconstruction.

Let $\hat{\mathbf{U}}^{\mathfrak{n}}$ be a polynomial corresponding to $\hat{\mathbf{U}}^t|_{[t^{\mathfrak{n}}, t^{\mathfrak{n}+1}]}$. The difference between the two is that $\hat{\mathbf{U}}^{\mathfrak{n}}$ is defined on all $\mathbb{R}$. Then $\hat{\mathbf{U}}^t|_{[t^{\mathfrak{n}}, t^{\mathfrak{n}+1}]}$ can be computed by prescribing, as per $\mathcal{H}(m, d, r)$ Hermite interpolation, the values and derivatives in time for $\hat{\mathbf{U}}^{\mathfrak{n}}$.

Definition 3.2.1. *Let $\hat{\mathbf{U}}^t \in S_t^q(0, T; S_x^{p,N}(\Omega))$ be the $\mathcal{H}(m, d, r)$ reconstruction in time with $q = (d+2)(p+1) + r + 1$ defined by requiring*

$$\mathrm{d}_t^k \hat{\mathbf{U}}^{\mathfrak{n}}(t^j) = \mathrm{d}_t^k \mathbf{U}^j(t^j), \quad \text{for } k = 0, \ldots, d+1 \text{ and } j = \mathfrak{n} - m, \ldots, \mathfrak{n} \tag{3.27}$$

$$\mathrm{d}_t^k \hat{\mathbf{U}}^{\mathfrak{n}}(t^{\mathfrak{n}+1}) = \mathrm{d}_t^k \mathbf{U}^{\mathfrak{n}+1} \quad \text{for } k = 0, \ldots, r+1 \tag{3.28}$$

$$\hat{\mathbf{U}}^t|_{[t^{\mathfrak{n}}, t^{\mathfrak{n}+1}]} = \hat{\mathbf{U}}^{\mathfrak{n}}|_{[t^{\mathfrak{n}}, t^{\mathfrak{n}+1}]}, \quad \text{for } \mathfrak{n} = 0, \ldots, \mathfrak{N} - 1 \tag{3.29}$$

Remark 3.2.2. The first and zeroth order derivative in time is readily available from (3.9) and computed by default for time stepping. Higher order time derivatives of the numerical solution can be replaced by approximations without affecting the optimality of the reconstruction. For more details on the computation of higher derivatives refer to Section 2.3 in [40].

Since Hermite interpolation is employed, from the standard results for Hermite interpolation [58],[89], we have the following result for the reconstruction in time.

Lemma 3.2.3. *Let $\hat{\mathbf{U}}^t$ be a $H(m, d, r)$ reconstruction as defined in 3.2.1, then $\hat{\mathbf{U}}^t$ is well defined, explicitly computable, Lipschitz continuous and $\min(d+1, r+1)$ times differentiable.*

3.2.2 Reconstruction in space

The method outlined in the previous section can be employed to obtain a Lipschitz continuous reconstruction in time $\hat{\mathbf{U}}^t \in S_t^q(0,T;S_x^{p,N}(\Omega))$ from the numerical solution. In the next step, $\hat{\mathbf{U}}^t$ is employed to compute a space time reconstruction.

To this end, we employ a locally Lipschitz function $\mathbf{W}$ as stated below.

Hypothesis 3.2.4. Let there be a locally Lipschitz function $\mathbf{W} : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ such that for any compact $\mathcal{K} \subset \mathbb{R}^N$ there exists a constant $C(\mathcal{K}) > 0$ such that

$$|\mathbf{W}(\mathbf{U},\mathbf{V}) - \mathbf{U}| \leq C(\mathcal{K})|\mathbf{U}-\mathbf{V}|,\ |\mathbf{W}(\mathbf{U},\mathbf{V}) - \mathbf{V}| \leq C(\mathcal{K})|\mathbf{U}-\mathbf{V}|,\quad \forall\ \mathbf{U},\mathbf{V} \in \mathcal{K}. \tag{3.30}$$

Hypothesis 3.2.5. Employing the locally Lipschitz function $\mathbf{W}$, we assume the numerical flux $\mathbf{F}_h : \mathbb{R}^n \times \mathbb{R}^N \to \mathbb{R}^N$ falls into one of the following two categories:

i

$$\mathbf{F}_h(\mathbf{U},\mathbf{V}) = \mathbf{F}(\mathbf{W}(\mathbf{U},\mathbf{V})),\quad \forall\ \mathbf{U},\mathbf{V} \in \mathbb{R}^N \tag{3.31}$$

ii

$$\mathbf{F}_h(\mathbf{U},\mathbf{V}) = \mathbf{F}(\mathbf{W}(\mathbf{U},\mathbf{V})) - \mu(\mathbf{U},\mathbf{V};h)h^\alpha(\mathbf{U}-\mathbf{V}),\quad \forall\ \mathbf{U},\mathbf{V} \in \mathbb{R}^N \text{ and some } \alpha \in \mathbb{N}_0 \tag{3.32}$$

where the function μ is such that for any compact $\mathcal{K} \in \mathbb{R}^N$ and h small enough there exists $\mu_{\mathcal{K}} > 0$ such that

$$|\mu(\mathbf{U},\mathbf{V};h)| \leq \mu_{\mathcal{K}}\left(1 + \frac{|\mathbf{U}-\mathbf{V}|}{h}\right). \tag{3.33}$$

Remark 3.2.6. We employ second order polynomials for DG semi-discretization and a third order strong stability preserving Runge Kutta scheme when conducting numerical experiments. In particular, we employ the Local Lax Friedrichs flux. To reconstruct the numerical solution obtained from the mentioned SSP-RK scheme, we employ $H(0,0,0)$ for reconstruction in time and $\mathbf{W}(\mathbf{U},\mathbf{V}) := \frac{1}{2}(\mathbf{U}+\mathbf{V})$ for reconstruction in space. Note that the Lax-Friedrichs flux is of the form (3.32) with

$$\alpha = 0,\ \mu(\mathbf{U},\mathbf{V}) = \lambda\,\mathrm{I} - \frac{\mathbf{F}(\mathbf{U}) - 2\mathbf{F}(\mathbf{W}(\mathbf{U},\mathbf{V})) + \mathbf{F}(\mathbf{V})}{2|\mathbf{V}-\mathbf{U}|^2} \bigotimes (\mathbf{V}-\mathbf{U}). \tag{3.34}$$

With the help of the function $\mathbf{W}$, we can define a spatial reconstruction method, which when applied to the reconstruction in time $\hat{\mathbf{U}}(\cdot,t)$ for some $t \in [0,T]$ provides a space-time Lipschitz continuous reconstruction $\hat{\mathbf{U}} \in S_x^{p+1,N}(\Omega)$. The definition of the reconstruction in space is as follows.

Definition 3.2.7. *Let $\hat{\mathbf{U}}^t$ be the reconstruction in time in accordance with Definition 3.2.1. Then, the space-time reconstruction $\hat{\mathbf{U}}^{st}(\cdot,t) \in S_x^{p+1,N}(\Omega)$ for $t \in [0,T]$ is defined by*

$$\int_\Omega \hat{\mathbf{U}}^{st}(\cdot,t)\phi_i^k(x)\,\mathrm{dx} = \int_\Omega \hat{\mathbf{U}}^t(\cdot,t)\phi_i^k\,\mathrm{dx},\quad \textit{for}\ \forall\ k\ \textit{and}\ i = 0,\ldots,p-1 \tag{3.35}$$

$$\hat{\mathbf{U}}^{st}(x_k^\pm,t) = \mathbf{W}(\hat{\mathbf{U}}^t(x_k^+,t),\hat{\mathbf{U}}^t(x_k^-,t)),\quad \forall\ k. \tag{3.36}$$

Employing Lemma 3.2.3 and Definition 3.2.7 for reconstruction in space, we have the following.

Lemma 3.2.8. *Let $\hat{\mathbf{U}}^{st}$ be the space-time reconstruction as defined in 3.2.7, then for $t \in [0, T]$ the function $\hat{\mathbf{U}}^{st}(\cdot, t)$ is well-defined, locally computable and Lipschitz continuous in space and time.*

The reader is referred to [40] for the proof.

Remark 3.2.9. Note that the reconstruction in time defined above employing $\mathcal{H}(m, d, r)$ Hermite interpolation requires that the numerical flux and the source term needs a certain amount of regularity, particularly $\mathbf{F}_h, \mathbf{R} \in C^{\max(d,r)}\left(\mathbb{R}^N, \mathbb{R}^N\right)$.

Remark 3.2.10. The reconstruction as defined in 3.2.7, allows us to compute a residual as follows

$$\mathcal{R}_c := \partial_x \hat{\mathbf{U}}^{st} + \partial_x \mathbf{F}\left(\hat{\mathbf{U}}^{st}\right) - \frac{1}{\varepsilon}\mathbf{R}(\hat{\mathbf{U}}^{st}). \tag{3.37}$$

We can refer to the reconstruction as optimal if the convergence order of the residual $\mathcal{R}_c$ matches the convergence order of the numerical method. We assume that the time step is such that it satisfies a CFL type condition, i.e. $\Delta t = \mathcal{O}(\Delta x_{\min})$, and that the order of the time stepping method is $q = p + 1$. Chi-Wang Shu et al. [99],[100] derived error estimates for second and third order RK-DG schemes. The error estimates they derived for a q^{th} order Runge-Kutta DG method employing polynomial of degree p for systems of hyperbolic conservation laws is $\mathcal{O}(h^{p+\gamma})$ where $\gamma = 1$ for upwind type fluxes and $\frac{1}{2}$ for general monotone fluxes.

The optimality of the space time reconstruction in particular depends on the choice of the numerical flux and the function $\mathbf{W}$. Assuming that the error of the SSP-RK DG scheme is $\mathcal{O}(h^{p+\gamma})$, Giesselmann et al [42] showed that the reconstruction is conditionally optimal for numerical fluxes of type (3.31), where the residual is of the order $\mathcal{O}(h^{p+\gamma})$ and in general sub-optimal for fluxes with artificial viscosity of type (3.32), where the residuals are of the order $\mathcal{O}(h^{p+\gamma+\alpha-1})$. For more details, the reader is referred to the cited works. Note that in the numerical experiments we conducted, where we employed the Local Lax Friedrichs flux, the error was of order $\mathcal{O}(h^{p+1})$. Hence, in the mesh and model adaptation strategy we propose, the order of the residual is assumed to be $\mathcal{O}(h^p)$ when employing the Local Lax Friedrichs flux.

3.3 Mesh and model adaptive simulation

Dynamic mesh and model adaptation consists of locally making a decision about the mesh and model refinement level to be employed with the objective of reducing the computational resources needed without significantly affecting the accuracy of the numerical simulation. This leads to a sub-domain where the complex system is employed and a sub-domain where the simple system is employed. In Sections 3.3.1 and 3.3.2, the notations employed for mesh and model adaptive meshes and numerical solutions are introduced. Since our error analysis requires that we compute a Lipschitz reconstruction from the numerical solution, we define a method of reconstruction for mesh and model adaptive

solutions in Section 3.3.3. Furthermore, since the sub-domains can dynamically change after each time step, when switching from the complex to the simple system in some cells, the numerical solutions in those cells need to be converted from the complex to the simple system and vice versa. The algorithms to do such conversions are defined in Section 3.3.4. Finally, coupling between the two models needs to be specified at the spatial interfaces, which is defined in Section 3.3.6.

3.3.1 Notation for mesh

Let the mesh and the index set of cells at time $t^{\mathfrak{n}}$ be denoted by $\mathcal{G}^{\mathfrak{n}}$ and $\mathcal{I}^{\mathfrak{n}} := \{1, \ldots, \mathcal{N}^{\mathfrak{n}}\}$ respectively, where $\mathcal{N}^{\mathfrak{n}}$ is the total number of cells in the mesh $\mathcal{G}^{\mathfrak{n}}$.

Furthermore, let $\Delta x^{\mathfrak{n}}$ be an array containing the mesh-widths of the cells in the mesh. Consequently, the cells in the mesh $\mathcal{G}^{\mathfrak{n}}$ are given by

$$\mathcal{V}_k := (x_k, x_{k+1}), \quad \forall\ k\ \in\ \mathcal{I}^{\mathfrak{n}}, \tag{3.38}$$

where

$$x_k = a + \sum_{i=1}^{k-1} \Delta x_i^{\mathfrak{n}}. \tag{3.39}$$

Model indicator

Let $\Theta^{\mathfrak{n}}$ be an array indicating the model being employed in the cells that constitute mesh $\mathcal{G}^{\mathfrak{n}}$.

The entries of the array $\Theta^{\mathfrak{n}}$ are such that

$$\Theta_k^{\mathfrak{n}} := \begin{cases} 1, & \text{if complex system is employed in } \mathcal{V}_k \in \mathcal{G}^{\mathfrak{n}} \\ 0, & \text{if simple system is employed in } \mathcal{V}_k \in \mathcal{G}^{\mathfrak{n}} \end{cases}.$$

Furthermore, let

$$\Omega_c^{\mathfrak{n}} := \bigcup_{k \in \mathcal{I}^{\mathfrak{n}}, \Theta_k = 1} \mathcal{V}_k, \quad \Omega_s^{\mathfrak{n}} := \bigcup_{k \in \mathcal{I}^{\mathfrak{n}}, \Theta_k = 0} \mathcal{V}_k, \tag{3.40}$$

be the sub-domains where the complex and the simple system are employed respectively.

The set of interfaces, i.e. the cell boundaries where different models are employed in the cells sharing the cell boundary is denoted by $\Gamma^{\mathfrak{n}}$, where

$$\Gamma^{\mathfrak{n}} := \bar{\Omega}_c^{\mathfrak{n}} \cap \bar{\Omega}_s^{\mathfrak{n}}. \tag{3.41}$$

Remark 3.3.1. Since the model and mesh refinement level to be employed is determined after each time step, the mesh and the models employed may differ from one time step to the next.

3.3.2 Numerical Solution

The numerical solution at time $t^{\mathfrak{n}}$, where the mesh employed is $\mathcal{G}^{\mathfrak{n}}$ and models employed in the cells are according to the model indicator $\Theta^{\mathfrak{n}}$, is denoted by $\mathbf{U}_h^{\mathfrak{n},i}$ and $\mathbf{u}_h^{\mathfrak{n},j}$ for some cell $\mathcal{V}_i \in \Omega_c^{\mathfrak{n}}$ and for some cell $\mathcal{V}_j \in \Omega_s^{\mathfrak{n}}$ respectively.

Since the simple system is employed in $\Omega_s^{\mathfrak{n}}$ and the complex system in $\Omega_c^{\mathfrak{n}}$, for $\mathbf{U}_h^{\mathfrak{n},i}$ and $\mathbf{u}_h^{\mathfrak{n},j}$

$$\mathbf{U}_h^{\mathfrak{n},i} \in S_x^{p,N}(\mathcal{V}_i),\ \mathbf{u}_h^{\mathfrak{n},j} \in S_x^{p,n}(\mathcal{V}_j) \tag{3.42}$$

holds.

3.3.3 Reconstruction

We need to compute a Lipschitz reconstruction from the numerical solution of a mesh and model adaptive numerical simulation in order to employ the error estimates derived in the previous chapter. Furthermore, the space-time reconstruction should satisfy the condition (2.46) at the interface. Next, we discuss a method to obtain a Lipschitz continuous space-time reconstruction from a mesh and model adaptive solution.

For cells away from the interface, the reconstruction is carried out by employing the method outlined in Section 3.2. For cells at the interface, i.e. a cell boundary where the complex system is employed on one side of the interface and the simple system on the other side, the reconstruction in time is carried out as discussed in Section 3.2.1, while the method to reconstruct in space needs to be modified.

We discuss the reconstruction, without loss of generality, assuming $q = 3$ and $p = 2$ and employing $H(0,0,0)$ Hermite interpolation for reconstruction in time. Let $x_k \in \Gamma^{\mathfrak{n}}$ and let the complex system be employed in cell $\mathcal{V}_k$ and the simple system in cell $\mathcal{V}_{k+1}$.

The reconstruction in space for cells on the interface is done as follows.

Definition 3.3.2. *Let the reconstruction in time on the time interval $[t^{\mathfrak{n}-1}, t^{\mathfrak{n}}]$ for cells $\mathcal{V}_k$ and $\mathcal{V}_{k+1}$ be $\hat{\mathbf{U}}_k^{t,\mathfrak{n}} \in S_t^q(t^{\mathfrak{n}-1}, t^{\mathfrak{n}}; S_x^{p,N}(\mathcal{V}_k))$ and $\hat{\mathbf{u}}_{k+1}^{t,\mathfrak{n}} \in S_t^q(t^{\mathfrak{n}-1}, t^{\mathfrak{n}}; S_x^{p,n}(\mathcal{V}_{k+1}))$ respectively. Then the space-time reconstructions $\hat{\mathbf{U}}_k^{st}(\cdot,t) \in S_x^{p+1,N}(\mathcal{V}_k)$ and $\hat{\mathbf{u}}^{st}(\cdot,t) \in S_x^{p+1,n}(\mathcal{V}_{k+1})$ for $t \in [t^{\mathfrak{n}-1}, t^{\mathfrak{n}}]$ on cells $\mathcal{V}_k$ and $\mathcal{V}_{k+1}$ are defined by requiring*

$$\int_{\mathcal{V}_k} \hat{\mathbf{U}}_k^{st,n}(x)\phi_i^k(x)\,\mathrm{dx} = \int_{\mathcal{V}_k} \hat{\mathbf{U}}^{t,\mathfrak{n}}(x,t)\phi_i^k(x)\,\mathrm{dx}, \quad \textit{for } i = 0,\ldots,p-1 \tag{3.43}$$

$$\int_{\mathcal{V}_{k+1}} \hat{\mathbf{u}}_{k+1}^{st,n}(x)\phi_i^{k+1}(x)\,\mathrm{dx} = \int_{\mathcal{V}_{k+1}} \hat{\mathbf{u}}_{k+1}^{t,\mathfrak{n}}(x,t)\phi_i^{k+1}(x)\,\mathrm{dx}, \quad \textit{for } i = 0,\ldots,p-1 \tag{3.44}$$

$$\hat{\mathbf{U}}_k^{st}(x_k,t) = M\left(\frac{1}{2}\left(\hat{\mathbf{u}}^{t,\mathfrak{n}}(x_k,t) + \mathbb{P}\hat{\mathbf{U}}^{t,\mathfrak{n}}(x_k,t)\right)\right) \tag{3.45}$$

$$\hat{\mathbf{u}}^{st}(x_k,t) = \mathbb{P}M\left(\frac{1}{2}\left(\hat{\mathbf{u}}^{t,\mathfrak{n}}(x_k,t) + \mathbb{P}\hat{\mathbf{U}}^{t,\mathfrak{n}}(x_k,t)\right)\right) \tag{3.46}$$

3.3.4 Model conversion

Dynamic heterogeneous model adaptation involves switching between the models. To this end, we need to define algorithms to convert the numerical solution from one model to the other. In this section, the algorithms employed for the model conversion are described.

Simple system to the complex system

Let cell $\mathcal{V}_i$ be such that the numerical solution in the cell

$$\mathbf{u}_h^{\mathfrak{n},i}(x) := \sum_{k=0}^{p} \mathbf{u}_k^i(t^{\mathfrak{n}})\phi_k^i(x), \quad x \in \mathcal{V}_i \tag{3.47}$$

needs to be converted from the simple system to the complex system. We wish to convert $\mathbf{u}_h^{\mathfrak{n},i} \in S_x^{p,n}(\mathcal{V}_i)$ to $\tilde{\mathbf{U}}_h^{\mathfrak{n},i} \in S_x^{p,N}(\mathcal{V}_i)$.

The first step is to change the representation of the numerical solution $\mathbf{u}_h^{\mathfrak{n},i}$ in the cell $\mathcal{V}_i$ from a modal to a nodal form. We employ Gauss-Legendre nodes to do so. Next, the values of the Maxwellian at the nodes are calculated. Finally, an interpolating polynomial can be calculated using polynomial interpolation and represented in the desired form.

Algorithm 2 Converting the numerical solution from the simple to the complex system

1: **procedure** $\mathtt{C}_{cs}(\mathbf{u}_h^{\mathfrak{n},i})$
2: Calculate the values of the numerical solution

$$\mathfrak{u}_i^{\mathfrak{n},k} := \mathbf{u}_h^{\mathfrak{n},i}(x_i^k), \quad k = 0, \ldots, p \tag{3.48}$$

at the Gauss-Legendre nodes $x_i^0 < \ldots < x_i^p$ on the cell $\mathcal{V}_i$.
3: Calculate the Maxwellian of the nodal values, given by

$$\mathfrak{U}_i^{\mathfrak{n},k} := M(\mathfrak{u}_i^{\mathfrak{n},k}), \quad k = 0, \ldots, p. \tag{3.49}$$

4: Calculate an interpolating polynomial employing the calculated values of the Maxwellian and express it in the desired form

$$\tilde{\mathbf{U}}_h^{\mathfrak{n},i}(x) := \sum_{k=0}^{p} \tilde{\mathbf{U}}_k^{\mathfrak{n},i}(t^{\mathfrak{n}})\phi_k^i(x), \quad x \in \mathcal{V}_i. \tag{3.50}$$

5: **return** $\tilde{\mathbf{U}}_h^{\mathfrak{n},i}$
6: **end procedure**

Complex system to the simple system

Next, we describe the algorithm to convert the numerical solution from the complex to the simple system. Let cell $\mathcal{V}_i$ be such that the numerical solution in the cell

$$\mathbf{U}_h^{\mathtt{n},i}(x) := \sum_{k=0}^{p} \mathbf{U}_k^i(t^{\mathtt{n}})\phi_k^i(x), \ x \in \mathcal{V}_i. \tag{3.51}$$

needs to be converted from the complex to the simple system. We wish to convert $\mathbf{U}_h^i \in S_x^{p,N}(\mathcal{V}_i)$ to $\tilde{\mathbf{u}}_h^i \in S_x^{p,n}(\mathcal{V}_i)$. The numerical solution is converted into the desired form by simply projecting it with the projection matrix $\mathbb{P}$.

Algorithm 3 Converting the numerical solution from the complex to the simple system

1: **procedure** $\mathtt{C}_{sc}(\mathbf{U}_h^{\mathtt{n},i})$
2: The numerical solution converted to the simple system will be given by

$$\tilde{\mathbf{u}}_h^{\mathtt{n},i}(x) := \sum_{k=0}^{p} \mathbb{P}\mathbf{U}_k^i(t^{\mathtt{n}})\phi_k^i(x), \quad x \in \mathcal{V}_i \tag{3.52}$$

3: **return** $\tilde{\mathbf{u}}_h^{\mathtt{n},i}$
4: **end procedure**

3.3.5 Limiting

As previously discussed in Section 3.1.2, limiting is needed to suppress numerical oscillations near discontinuities for numerical solutions to hyperbolic conservation and balance laws. We employ a limiter, where the limiting is performed on the characteristic variables. We need to know the eigenvectors of the Jacobian of the flux to convert the balanced or the conserved variables to the characteristic variables. The flux of the simple system is a function of the Maxwellian of the conserved variables $\mathbf{u}$. To obtain the Maxwellian values, a system of non-linear equations needs to be solved. Hence, the eigenvectors for the simple system will in general not be explicitly known as a function of the conserved variables of the simple system. On the other hand, the eigenvectors of the Jacobian of the flux for the complex system are explicitly known. Therefore, in the model adaptive simulations, limiting for the cells in which the simple system is employed is carried out by converting the numerical solution to the complex system, followed by limiting the converted numerical solution and finally the limited numerical solution is converted back to the simple system, i.e. for $\mathbf{u}_h^{\mathtt{n},i}$ in cell $\mathcal{V}_i$ such that $\Theta_i^{\mathtt{n}} = 0$ is limited by $\mathtt{C}_{\mathtt{cs}}\left(\mathtt{L}\left(\mathtt{C}_{sc}\left(\mathbf{u}_h^{\mathtt{n},i}\right)\right)\right)$.

3.3.6 Coupling conditions

In the model adaptive numerical simulations, we employ models of different complexity in different sub-domains, which gives rise to interfaces, which are inherently artificial.

Consequently coupling conditions in the form of numerical fluxes at the interface need to be specified.

Coupling of non-linear hyperbolic conservation and balance laws is a well studied area of research. The coupling approach strongly depends on the problem at hand. In the works [1], [2], coupling of two-phase flows between a homogeneous relaxation and a homogeneous equilibrium model is discussed. The flow is in physical equilibrium when the homogeneous equilibrium model is employed and in non-equilibrium when the relaxation model is employed. In [48] and [80], the authors studied coupling of hyperbolic problems with discontinuous fluxes with applications to acoustic flows. Another key area with industrial and scientific applications is coupling of different physical systems, such as coupling of the linear elastic equations with compressible fluid flow equations in fluid-structure interaction problems [55] or compressible gas flows coupled with gas generators in gas networks [88]. Coupling of systems of hyperbolic conservation and balance laws pose non-trivial difficulties since the coupling methods need to be physically consistent and should avoid creation of artefacts at the interface.

For the problem at hand, we propose a coupling and give a heuristic argument as to why we expect the coupling to work reasonably well provided that the interface is placed suitably.

Let cells $\mathcal{V}_i, \mathcal{V}_{i+1}$ at time $t^{\mathfrak{n}}$ be such that the complex system and the simple system is used in cells $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$ respectively. Let the numerical solution in cells $\mathcal{V}_i, \mathcal{V}_{i+1}$ be $\mathbf{U}_h^{\mathfrak{n},i}$ and $\mathbf{u}_h^{\mathfrak{n},i+1}$. The cell boundary at x_{i+1} is an interface since we employ different models across the cell boundary at x_{i+1}. We need to specify numerical fluxes $\mathbf{F}_h^i(t^{\mathfrak{n}})$ and $\mathbf{g}_h^{i+1}(t^{\mathfrak{n}})$ at x_{i+1} for cell $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$.

Let

$$\mathbf{U}_i^-(t^{\mathfrak{n}}) := \lim_{s\to 0^+} \mathbf{U}_h^{\mathfrak{n},i}(x_{i+1}-s),\ \mathbf{u}_i^+(t^{\mathfrak{n}}) := \lim_{s\to 0^+} \mathbf{u}_h^{\mathfrak{n},i+1}(x_{i+1}+s). \tag{3.53}$$

be the trace values at the interface.

If the numerical solution in the cell $\mathcal{V}_i$ is away from the equilibrium manifold $\mathcal{M}$, it implies that $\mathbf{U}_i^-(t^{\mathfrak{n}}) \neq M(\mathbb{P}\mathbf{U}_i^-(t^{\mathfrak{n}}))$. The larger the distance $|\mathbf{U}_i^- - M(\mathbb{P}\mathbf{U}_i^-)|$ is, i.e. the further away the numerical solution in cell $\mathcal{V}_i$ is from the equilibrium manifold, the higher the likelihood of producing artefacts at the interface. This can be reasoned both mathematically and physically. From the viewpoint of physics, the further away the numerical solution in cell $\mathcal{V}_i$ is from the equilibrium manifold, the higher the likelihood for the dynamics in cell $\mathcal{V}_i$ to significantly differ from equilibrium dynamics. The non-equilibrium dynamics will be disregarded if the numerical solution travels from cell $\mathcal{V}_i$ to $\mathcal{V}_{i+1}$. From the viewpoint of mathematics, if the numerical solution is away from the equilibrium manifold, it will result in a jump discontinuity at the interface since we employ the projection matrix and the Maxwellian in cell $\mathcal{V}_{i+1}$. Both perspectives hint that if the numerical solution is away from the equilibrium manifold or if the dynamics in cell $\mathcal{V}_i$ differs to a large extent from the equilibrium dynamics, the coupling would lead to artefacts.

The objective of model adaptation is to employ the simple model instead of the complex one to reduce the computational resources employed, but not at the expense of huge modelling errors. Hence, the domain decomposition should be done in such a way that it does not gives rise to artefacts. Greedily decomposing the domain would be counter-productive to the objective of model adaptation.

Hence, when proposing the numerical fluxes at the interface we proceed under the assumption that the domain decomposition is carried out such that it satisfies the following criteria:

(i) The interfaces are at points in space such that the numerical solution in the cell where the complex system is employed is close to the equilibrium manifold, i.e. $\left(\frac{1}{\Delta x_i}\int\limits_{\mathcal{V}_i} H\left(\mathbf{U}_h^{\mathfrak{n},i}(x)|M\left(\mathbb{P}\mathbf{U}_h^{\mathfrak{n},i}(x)\right)\right)\mathrm{dx}\right)^{0.5} \ll 1$. Note that, relative entropy is employed to quantify the distance to the equilibrium manifold in order to remain consistent with the rest of mesh and model adaptation strategy.

(ii) We assume that the time evolution of the quantities in cell $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$ can be adequately described by evolution on the equilibrium manifold.

The above two assumptions are equivalent to requiring that the modelling assumption employed to derive the simple system is largely satisfied in cell $\mathcal{V}_i$. In the case that the above two criteria are satisfied, we expect the coupling to work reasonably well. But, if not the case, then the expectation is that the coupling will give rise to artefacts.

Another desirable characteristic is for the numerical flux to be conservative. With this in mind, we define a numerical flux below.

Definition 3.3.3. *Let $\mathbf{F}_h(\mathbf{U},\mathbf{V}) : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ be a conservative numerical flux for the complex system. Let $\mathbf{U}_i^-(t^{\mathfrak{n}})$ and $\mathbf{u}_i^+(t^{\mathfrak{n}})$ be the trace values as defined in (3.53) then let*

$$\begin{aligned}\mathbf{F}_h^i(t^{\mathfrak{n}}) &:= \mathbf{F}_h\left(\mathbf{U}_i^-, M(\mathbf{u}_i^+)\right),\\ \mathbf{g}_h^{i+1}(t^{\mathfrak{n}}) &:= \mathbb{P}\mathbf{F}_h\left(\mathbf{U}_i^-, M(\mathbf{u}_i^+)\right).\end{aligned} \tag{3.54}$$

3.3.7 Mesh adaptation

We employ a multiresolution based Runge-Kutta Discontinuous Galerkin method developed by Müller et. al. for mesh and model adaptive simulations. The core idea is to employ a hierarchy of nested meshes. Starting from a uniform mesh, the mesh is recursively subdivided at the midpoints of the cells. This results in a hierarchy of meshes at different resolution levels as shown in Figure 3.1. For the sake of illustration, let the order of polynomials employed be $p = 0$. Then, the cell averages, i.e. the numerical solution also referred to as single scale information, on some fine level l can be represented as cell

averages, i.e. single scale information on the coarser level $l-1$, plus an array of detail coefficients. This process can be recursively repeated. The decomposition of the numerical solution into fine scale information and single scale information can be exploited to carry out mesh adaptation by locally discarding the detail coefficients deemed insignificant and adding more details where needed. This leads to an adaptive mesh as shown in Figure 3.2. The details about the multiresolution based RKDG method in one space dimension can be found in [36], [37] and its extension to multiple space dimensions in [39].

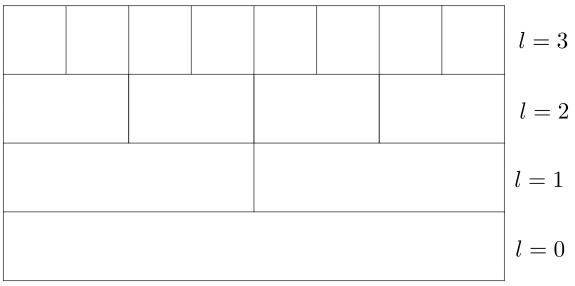

Figure 3.1: Nested mesh hierarchy

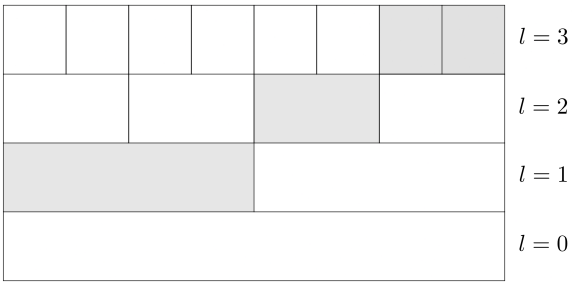

Figure 3.2: Mesh adaptation

Remark 3.3.4. The numerical simulations are carried out using the `Multiwave` library developed for first-order non-linear hyperbolic PDE systems developed and maintained by Siegfried Müller and Aleksey Sikstel [70].

3.3.8 Notation for mesh adaptation

In the mesh and model adaptive numerical simulations, the maximum level of refinement is denoted by L and the number of cells on the coarsest mesh at level 0 by $\mathcal{N}_0$. In addition to the notations introduced in Section 3.3.1, let $\mathcal{L}^{\mathfrak{n}}$ be an array containing the resolution levels being employed in cells constituting the mesh $\mathcal{G}^{\mathfrak{n}}$. Due to the dyadic refinement of the meshes in the mesh hierarchy, the mesh width of cell $\mathcal{V}_i$ is given by

$$\Delta x_i^{\mathfrak{n}} = \frac{b-a}{\mathcal{N}_0} \cdot \frac{1}{2^{\mathcal{L}_i^{\mathfrak{n}}}}, \quad \forall\, i \in \mathcal{I}^{\mathfrak{n}}. \tag{3.55}$$

3.4 Model and mesh adaptation

Let the mesh employed at time $t^{\mathfrak{n}}$ be $\mathcal{G}^{\mathfrak{n}}$ and the models employed be in accordance with the model indicator array $\Theta^{\mathfrak{n}}$. Let the array of mesh widths and resolution levels be $\Delta x^{\mathfrak{n}}$ and $\mathcal{L}^{\mathfrak{n}}$ respectively. In the discussion that follows, the assumption is that Local Lax Friedrichs flux is employed.

The objective of mesh and model adaptation is to reduce the computational resources needed by mesh coarsening and/or model coarsening. The level of mesh and model refinement to be employed in the next time step is locally determined for each space time cell $\mathcal{V}_k \times [t^{\mathfrak{n}-1}, t^{\mathfrak{n}}]$ for $k \in \mathcal{I}^{\mathfrak{n}}$. To this end, we define error indicators in the next section. The

error indicators will be defined based on the a posteriori error analysis conducted in the previous chapter. To complement the information provided by the error indicators, we also define mesh and model coarsening distances and finally propose a mesh and model adaptation strategy.

3.4.1 Mesh and modelling error indicators

With the intent of defining the error indicators, let cells $\mathcal{V}_i$ and $\mathcal{V}_j$ be such that the complex system is employed in cell $\mathcal{V}_i$ and the simple system in cell $\mathcal{V}_j$. Let the numerical solution and the reconstruction in the cells $\mathcal{V}_i$ and $\mathcal{V}_j$ be $\mathbf{U}_i^{\mathtt{n}}, \hat{\mathbf{U}}_i^{st,\mathtt{n}}$ and $\mathbf{u}_j^{\mathtt{n}}, \hat{\mathbf{u}}_j^{st,\mathtt{n}}$ respectively.

With the above quantities in hand, let

$$\mathtt{D}_c^{\mathtt{n},i} := \Delta x_i^{\mathtt{n}} \cdot \left(\frac{1}{\Delta x_i^{\mathtt{n}}} \cdot \frac{1}{t^{\mathtt{n}+1} - t^{\mathtt{n}}} \cdot \frac{1}{2} \int_{t^{\mathtt{n}}}^{t^{\mathtt{n}+1}} \int_{\mathcal{V}_i} |\, \mathrm{D}\, H\left(\hat{\mathbf{U}}_i^{st,\mathtt{n}}\right) \cdot \mathcal{R}_c^i|^2 \,\mathrm{dx}\,\mathrm{d}\tau \right)^{0.5} \tag{3.56}$$

be the so called discretization error indicator for the complex system, where the residual $\mathcal{R}_c^i$ is the residual in cell $\mathcal{V}_i$ as defined in (2.47).

Analogously, let

$$\mathtt{D}_s^{\mathtt{n},j} := \Delta x_j^{\mathtt{n}} \cdot \left(\frac{1}{\Delta x_j^{\mathtt{n}}} \cdot \frac{1}{t^{\mathtt{n}+1} - t^{\mathtt{n}}} \cdot \frac{1}{2} \int_{t^{\mathtt{n}}}^{t^{\mathtt{n}+1}} \int_{\mathcal{V}_j} |\, \mathrm{D}\, \eta\left(\hat{\mathbf{u}}_j^{st,\mathtt{n}}\right) \cdot \mathcal{R}_\delta^j|^2 \,\mathrm{dx}\,\mathrm{d}\tau \right)^{0.5} \tag{3.57}$$

be the discretization error indicator for the simple system where the residual $\mathcal{R}_\delta^j$ is as defined in (2.64).

Furthermore, for the simple system, we can define the modelling error indicator

$$\mathtt{M}_s^{\mathtt{n},i} := \left(\frac{1}{\Delta x_j} \cdot \frac{1}{t^{\mathtt{n}+1} - t^{\mathtt{n}}} \cdot \frac{\varepsilon}{\nu} \int_{t^{\mathtt{n}}}^{t^{\mathtt{n}+1}} \int_{\mathcal{V}_j} |\, \mathrm{D}^2\, \eta(\hat{\mathbf{u}}_j^{st,\mathtt{n}}) \cdot \mathcal{R}_\epsilon^j|^2 \,\mathrm{dx}\,\mathrm{d}\tau \right)^{0.5}, \tag{3.58}$$

where the residual $\mathcal{R}_\epsilon^j$ is as defined in (2.64).

The error indicators defined above are based on the error estimates derived in Theorem 2.5.12. In particular, $\mathtt{D}_s^{\mathtt{n},i}, \mathtt{D}_s^{\mathtt{n},i}$ and $\mathtt{M}_s^{\mathtt{n},i}$ are the terms D_c, D_s and $\mathcal{M}_s$ that appear in the pre-exponential factor scaled by the mesh width and time-step to account for the size of the space-time cell $\mathcal{V}_i \times [t^{\mathtt{n}-1}, t^{\mathtt{n}}]$. The discretization error indicators are additionally scaled by the mesh width.

Some notable characteristics and observations about the above defined error indicators are as follows:

(i) Due to the sub-optimality of the reconstruction in the case of Local Lax Friedrichs flux, as noted in Remarks 3.2.6 and 3.2.10, even if the solution is smooth, the residual of the complex system behaves like $\mathcal{R}_c^i = \mathcal{O}\left((\Delta x_i^{\mathtt{n}})^p\right)$. Similarly, for the simple

system; $r_s^j = \mathcal{O}\left(\left(\Delta x_j^{\mathtt{n}}\right)^p\right)$ and numerical experiments show that $\mathcal{R}_\delta^j = \mathcal{O}\left(\left(\Delta x_j^{\mathtt{n}}\right)^p\right)$. In contrast, in smooth regions, the error scales like $\mathcal{O}\left(\left(\Delta x_{i/j}^{\mathtt{n}}\right)^{p+1}\right)$. To account for this sub-optimality, the discretization error indicators are additionally scaled by the mesh width. Consequently, the discretization error indicators behave like $\mathcal{O}\left(\left(\Delta x_{i/j}^{\mathtt{n}}\right)^{p+1}\right)$.

(ii) Near discontinuities $\|\partial_x \hat{\mathbf{U}}_i^{st,\mathtt{n}}\|_{L^\infty}$ and $\|\partial_x \hat{\mathbf{u}}_j^{st,\mathtt{n}}\|_{L^\infty}$ behave like $\mathcal{O}(\Delta x_{i/j}^{-1})$. As a result, when $\Delta x \to 0$ the upper bound in (2.5.12) will not converge. This is due to the fact in general the entropy solutions need not be unique if the solution is not Lipschitz continuous [26]. In the case of one space dimension and given that the initial data and the characteristic fields satisfy some conditions, it can be shown that the entropy solutions are unique, see [47]. Hence, the error indicators in general will not be informative near discontinuities. On the other hand, the likelihood that the simplifying assumption made to derive the simple system holds at discontinuities will be slim. Consequently, by default, we have to employ the complex system at discontinuities. Hence, although the error indicators are not informative at discontinuities, this does not prove to be an obstacle to the objective of devising a model adaptation strategy.

(iii) Recall the modelling assumption we make when deriving the simple system is that the dynamics is the equilibrium dynamics. Hence, if the modelling assumption holds, the time evolution takes place such that the numerical solution remains on the equilibrium manifold. Hence, starting from the numerical solution on the equilibrium manifold, no modelling error should be incurred, when the reaction speed is infinitely fast, i.e. as $\varepsilon \to 0$ or if the convection is such that it does not push the numerical solution away from the equilibrium manifold. Note that in the regions where the solution is smooth, as $\varepsilon \to 0$, $\mathtt{M}_s^{\mathtt{n},i} \to 0$. Furthermore, it is clear that if convection has no contribution to Ker $(\mathbb{P})$, i.e. if $\mathcal{R}_\epsilon := \left(\mathrm{I} - \mathrm{D}\, M(\hat{\mathbf{u}}_j^{st,\mathtt{n}}) \cdot \mathbb{P}\right) \cdot \partial_x \mathbf{F}(M(\hat{\mathbf{u}}_j^{st,\mathtt{n}})) = 0$ then $\mathtt{M}_s^{\mathtt{n},j} = 0$. We can conclude that the behaviour of the modelling error indicator $\mathtt{M}_s^{\mathtt{n},i}$ is consistent with the physics of the problem.

(iv) $\mathtt{D}_s^{\mathtt{n},j}, \mathtt{D}_c^{\mathtt{n},i}$ and $\mathtt{M}_s^{\mathtt{n},j}$ are indicators for the discretization and modelling errors incurred in the space-time cell $\mathcal{V}_i \times [t^{\mathtt{n}-1}, t^{\mathtt{n}}]$. In the case that these indicators are large, they point to the need for model and mesh refinement. Although these indicators inform us about mesh and model refinement, they in themselves do not provide sufficient information to allow mesh and model coarsening due to the following reasons:

a) When we coarsen the mesh, we loose some information. Hence, the mesh should be coarsened, only if the amount of information lost is below some threshold.

b) With regards to model coarsening, we should model coarsen only if the numerical solution is sufficiently close to the equilibrium manifold and if the evolution of the quantities can be adequately described by evolution on the equilibrium manifold. If either of these two criteria are not satisfied, to the extent we prescribe, then the model should not be coarsened.

To address the points above, we define so called mesh coarsening distance for the simple and the complex system and model coarsening distance and model coarsening error indicator for the complex system.

3.4.2 Mesh and model coarsening distances

Since we employ the relative entropy framework to derive the error estimates, we also employ the relative entropy when defining the coarsening distances.

Model coarsening distance

When the model is coarsened, the numerical solution is converted from the complex to the simple system. Hence, model coarsening distance is defined as

$$\kappa_{\varepsilon}^{\mathfrak{n},i} := \left(\frac{1}{\Delta x_i} \cdot \frac{1}{t^{\mathfrak{n}} - t^{\mathfrak{n}-1}} \int_{\mathcal{V}_i} H\left(\mathbf{U}_h^{\mathfrak{n},i} | M(\mathbb{P}\mathbf{U}_h^{\mathfrak{n},i})\right) \mathrm{dx}\right)^{0.5}. \tag{3.59}$$

Note that each time we coarsen the model, the error incurred due to coarsening the model adds up. To account for this, the integral of the relative entropy between the numerical solution and the model coarsened numerical solution is scaled by the inverse of the time step size. This ensures that the total error incurred in a model adaptive simulation is not sensitive to the total number of time steps taken in a model adaptive simulation.

Mesh coarsening distance

Let cells $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$ be such that the complex system is employed in these cells. Let the numerical solutions be $\mathbf{U}_h^{\mathfrak{n},i} \in S_x^{p,N}(\mathcal{V}_i)$ and $\mathbf{U}_h^{\mathfrak{n},i+1} \in S_x^{p,N}(\mathcal{V}_{i+1})$ in cells $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$ respectively. Assuming that the mesh is to be coarsened, i.e. cells $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$ are to be combined to give a coarser cell $\tilde{\mathcal{V}}_k = (x_i, x_{i+2})$ and the coarsened numerical solution $\tilde{\mathbf{U}}_h^{\mathfrak{n},k} \in S_x^{p,N}(\tilde{\mathcal{V}}_k)$.

Then, the mesh coarsening distance is defined as

$$\kappa_{\delta}^{\mathfrak{n},k} := \left(\frac{1}{\Delta x_i} \cdot \frac{1}{t^{\mathfrak{n}} - t^{\mathfrak{n}-1}} \cdot \int_{\mathcal{V}_i} H\left(\mathbf{U}_h^{\mathfrak{n},i} | \tilde{\mathbf{U}}_h^{\mathfrak{n},k}\right) \mathrm{dx} + \frac{1}{\Delta x_{i+1}} \cdot \frac{1}{t^{\mathfrak{n}} - t^{\mathfrak{n}-1}} \cdot \int_{\mathcal{V}_{i+1}} H\left(\mathbf{U}_h^{\mathfrak{n},i+1} | \tilde{\mathbf{U}}_h^{\mathfrak{n},k}\right) \mathrm{dx}\right)^{0.5}. \tag{3.60}$$

Similar to model coarsening, each time we coarsen the mesh, errors due to coarsening the mesh aggregate. To account for this, the integral of the relative entropy between the numerical solution and the mesh coarsened numerical solution is scaled by the inverse of the size of the time step.

Coarsening distance for the simple system can be analogously defined.

Let cells $\mathcal{V}_j$ and $\mathcal{V}_{j+1}$ be such that the simple system is employed in these cells. Let the numerical solutions be $\mathbf{u}_h^{\mathfrak{n},j} \in S_x^{p,n}(\mathcal{V}_j)$ and $\mathbf{u}_h^{\mathfrak{n},j+1} \in S_x^{p,n}(\mathcal{V}_{j+1})$ in cells $\mathcal{V}_j$ and $\mathcal{V}_{j+1}$

respectively. The numerical solutions can be converted to the complex system employing Algorithm 2 to give $\mathbf{U}_c^{\mathtt{n},j} := \mathtt{C}_{sc}(\mathbf{u}_h^{\mathtt{n},j})$ and $\mathbf{U}_c^{\mathtt{n},j+1} := \mathtt{C}_{sc}(\mathbf{u}_h^{\mathtt{n},j+1})$. Assuming the cells $\mathcal{V}_j$ and $\mathcal{V}_{j+1}$ are combined to give a coarse cell $\tilde{\mathcal{V}}_k = (x_j, x_{j+2})$ and the coarsened numerical solution $\tilde{\mathbf{U}}_c^{\mathtt{n},k} \in S_x^{p,N}(\tilde{\mathcal{V}}_k)$.

Then, the mesh coarsening distance is defined as

$$\kappa_\delta^{\mathtt{n},k} := \left(\frac{1}{\Delta x_j} \cdot \frac{1}{t^{\mathtt{n}} - t^{\mathtt{n}-1}} \cdot \int_{\mathcal{V}_j} H\left(\mathbf{U}_c^{\mathtt{n},j} | \tilde{\mathbf{U}}_h^{\mathtt{n},k}\right) \mathrm{dx} + \frac{1}{\Delta x_{j+1}} \cdot \frac{1}{t^{\mathtt{n}} - t^{\mathtt{n}-1}} \cdot \int_{\mathcal{V}_{j+1}} H\left(\mathbf{U}_c^{\mathtt{n},j+1} | \tilde{\mathbf{U}}_h^{\mathtt{n},k}\right) \mathrm{dx} \right)^{0.5}. \tag{3.61}$$

Remark 3.4.1. The equivalent counterpart of (3.60) for the simple system would be calculating the coarsening distance by evaluating the Maxwellian of the solutions $\mathbf{u}_h^{\mathtt{n},j}, \mathbf{u}_h^{\mathtt{n},j+1}$ and a coarsened numerical solution $\tilde{\mathbf{u}}_c^{\mathtt{n},k}$. The coarsening distance defined in (3.61) provides essentially the same information. Mesh coarsening distance for the simple system is employed in the form as defined in (3.61) for the ease of implementation. The two notions differ since coarsening the solution and computation of the Maxwellian does not commute.

3.4.3 Modeling coarsening error indicator for the complex system

Let the system being employing in cell $\mathcal{V}_i$ be the complex system and the numerical solution be $\mathbf{U}_h^{\mathtt{n},i} \in S_x^{p,N}(\mathcal{V}_i)$. Let $\tilde{\mathbf{u}}_c^{\mathtt{n},i} := \mathtt{C}_{cs}\left(\mathbf{U}_h^{\mathtt{n},i}\right)$ be the numerical solution converted from the complex to the simple system. Let $\hat{\mathbf{u}}_{c,i}^{st,\mathtt{n}} : \mathcal{V}_i \times [t^{\mathtt{n}-1}, t^{\mathtt{n}}] \to \mathbb{R}^n$ be a space-time Lipschitz continuous reconstruction of $\tilde{\mathbf{u}}_c^{\mathtt{n},i}$.

Recall that the modelling error indicator $\mathtt{M}_s^{\mathtt{n},i}$ allows us quantify the extent to which the simplifying assumption made to derive the simple system holds for cells in $\Omega_s^{\mathtt{n}}$. When switching from the complex to the simple system, it is appropriate that we perform the same check. To this end, let

$$\mathcal{R}_{\epsilon,c}^i := \left(\mathrm{I} - \mathrm{D}\, M\left(\hat{\mathbf{u}}_{c,i}^{st,\mathtt{n}}\right) \cdot \mathbb{P}\right) \cdot \partial_x \mathbf{F}\left(M\left(\hat{\mathbf{u}}_{c,i}^{st,\mathtt{n}}\right)\right). \tag{3.62}$$

Employing the residual defined above, let

$$\mathtt{M}_c^{\mathtt{n},i} := \left(\frac{1}{\Delta x_i} \cdot \frac{1}{t^{\mathtt{n}+1} - t^{\mathtt{n}}} \cdot \frac{\varepsilon}{\nu} \int_{t^{\mathtt{n}}}^{t^{\mathtt{n}+1}} \int_{\mathcal{V}_i} |\,\mathrm{D}^2\, \eta(\hat{\mathbf{u}}_{c,i}^{st,\mathtt{n}}) \cdot \mathcal{R}_{\epsilon,c}^i|^2 \,\mathrm{dx}\,\mathrm{d}\tau \right)^{0.5}, \tag{3.63}$$

which we refer to as the model coarsening error indicator for the complex system.

We can observe that $\mathtt{M}_c^{\mathtt{n},i}$ is the the complex system counterpart of $\mathtt{M}_s^{\mathtt{n},i}$. This helps us determine the extent to which the simplifying modeling assumption would hold in the case that we do decide to model coarsen. If the simplifying assumption does not hold to the extent we prescribe, we continue employing the complex system.

3.4.4 Mesh and model adaptation strategy

When devising the mesh and model adaptation strategy, we seek to balance the two sources of errors; discretization and modeling. We take this approach in order to avoid employing a mesh which is counterproductive to model adaptation and vice versa. For example, employing a very fine mesh to compensate for employing the coarse model. To this end, we propose a mesh and model adaptation strategy based on the defined error indicators and coarsening distances.

The information available to us is in the form of error indicators and coarsening distances for mesh and model adaptation. That being the case, a tolerance τ_r is set for the error indicators and tolerance τ_κ for the coarsening distances. Two different tolerances are set for the error indicators and the coarsening distances since they provide two different pieces of information.

Mesh adaptation

The mesh adaptation strategy needs to account for the structure of the error estimates and the nature of the solutions to hyperbolic conservation laws.

The mesh refinement level to be employed can be determined based on the behaviour of the discretization error indicator upon mesh refinement and mesh coarsening. Additionally, if the mesh is to be coarsened, we should do so only if the mesh coarsening distance is below the prescribed tolerance τ_κ. Furthermore, since the behaviour of the discretization error indicators will not behave exactly as discussed in Section 3.4.1, a factor of safety $\mathfrak{f}_\delta > 0$ is employed when coarsening the mesh. Bearing this in mind, in the first step, Algorithm 4 is employed to determine the level of mesh refinement to be employed in each cell at the end of every time step.

In the next step, the mesh refinement levels obtained from the mesh adaptation algorithm are modified to account for the nature of the solutions to hyperbolic conservation laws. Namely, we know that solutions to hyperbolic conservation laws can develop discontinuities in finite time. Hence, solutions to hyperbolic conservation laws may have regions of high gradients, discontinuities and plateaus. In the regions of high gradients and discontinuities, the solution will be smeared over a patch of cells. Moreover, the information, including discontinuities, propagates with finite time and in a certain direction. Consequently, discontinuities like shocks can travel in one time step from one cell to the next.

To account for these characteristics, the mesh refinement levels computed employing Algorithm 4 are modified as follows:

(i) Safety cells are appended to the left and the right of the patches of cells where the highest level L is employed.

(ii) In the case where the mesh refinement level goes from the highest level L to the

Algorithm 4 Spatial mesh adaptation

1: **procedure** $(\mathcal{L}^{\mathtt{n}}, \mathcal{L}^{\mathtt{n}+1}, \mathtt{D}^{\mathtt{n},i}_{c/s}, \kappa^{\mathtt{n},i}_{\delta})$

2:
$$l = \begin{cases} L & \text{if } \mathtt{D}^{\mathtt{n},i}_{c/s} \geq \tau_r \cdot 2^{L(p+1)} \\ \frac{\log\left(\mathtt{D}^{\mathtt{n},i}_{c/s}/\tau_r\right)}{\log(2^{p+1})} & \text{if } \tau_r < \log(\mathtt{D}^{\mathtt{n},i}_{c/s}) < 2^{L(p+1)} \cdot \tau_r \\ 0 & \text{if } \mathtt{D}^{\mathtt{n},i}_{c/s} < \tau_r \end{cases}$$

3: **if** $l < \mathcal{L}^{\mathtt{n}}_i$ and $\kappa^{\mathtt{n},i}_{\delta} < \tau_\kappa$ and $(2^{p+1} + \mathfrak{f}_\delta) \cdot \mathtt{D}^{\mathtt{n},i}_{c/s} < 2^{(\mathcal{L}^{\mathtt{n}}_i - 1)(p+1)} \cdot \tau_r$ **then**

4:
$$\mathcal{L}^{\mathtt{n}+1}_i = \mathcal{L}^{\mathtt{n}}_i - 1$$

5: **else if** $l > \mathcal{L}^{\mathtt{n}}_i$ **then**
$$\mathcal{L}^{\mathtt{n}+1}_i = l$$

6: **else**

7:
$$\mathcal{L}^{\mathtt{n}+1}_i = \mathcal{L}^{\mathtt{n}}_i$$

8: **end if**

9: **return** $\mathcal{L}^{\mathtt{n}+1}$

10: **end procedure**

smallest level $l = 0$, the levels are graded, i.e. the mesh refinement levels are chosen such that the mesh transitions step-wise from the highest to the smallest level.

(iii) When coarsening the mesh, in one time step, the mesh refinement level can only drop by one, see step 4 in Algorithm 4.

(iv) In mesh and model adaptive simulations, when the mesh at coupling interfaces is to be coarsened, i.e. when two fine cells are to be combined to give a coarser cell and the fine cells are such that the simple system is employed in one cell and the complex system in the other, then the mesh is coarsened and the complex system is employed in the coarsened cell.

Remark 3.4.2. Recall that at discontinuities, the error indicators diverge as $\Delta x \to 0$. As a result, the discretization error indicator will be of much larger magnitude than the tolerance τ_r. Hence, automatically, the highest resolution level L is employed at discontinuities. This is also necessary to ensure that the discontinuities are resolved well.

Model adaptation

Model adaptation is done employing the model error indicators and the model coarsening distance. In the same vein as mesh adaptation, the model is coarsened only if the model coarsening distance is less than τ_κ. Furthermore, when model coarsening, we employ $\mathtt{M}^{\mathtt{n},i}_c$

to ascertain if the simplifying assumption, in the case that we do model coarsen, holds to the extent we prescribe. Bearing in mind, that the indicator $\mathtt{M}_c^{\mathfrak{n},i}$ is the counterpart of $\mathtt{M}_s^{\mathfrak{n},i}$ for the complex system, switching from the complex to the simple system and back to the complex system in a matter of a few time steps should be avoided. This can occur when the modelling error indicators are close to the tolerance. To avoid switching frequently between the two systems, a factor of safety $0 < \mathfrak{f}_\varepsilon < 1$ is employed when comparing $\mathtt{M}_c^{\mathfrak{n},i}$ to the tolerance. Consequently, more caution is taken when switching from the complex to the simple system.

The models to be employed then are determined using Algorithm 5.

Algorithm 5 Spatial model adaptation

procedure $(\Theta_i^{\mathfrak{n}}, \Theta_i^{\mathfrak{n}+1}, \mathtt{M}_{c/s}^{\mathfrak{n},i}, \kappa_\varepsilon^{\mathfrak{n},i})$
  if $\Theta_i^{\mathfrak{n}} = 1$ and $\mathtt{M}_c^{\mathfrak{n},i} < \mathfrak{f}_\varepsilon \cdot \tau_r$ and $\kappa_\varepsilon^{\mathfrak{n},i} < \tau_\kappa$ **then**

$$\Theta^{\mathfrak{n}+1} = 0$$

  else if $\Theta_i^{\mathfrak{n}} = 0$ and $\mathtt{M}_s^{\mathfrak{n},i} > \tau_r$ **then**

$$\Theta^{\mathfrak{n}+1} = 1$$

  else

$$\Theta^{\mathfrak{n}+1} = \Theta^{\mathfrak{n}}$$

  end if
  return $\Theta^{\mathfrak{n}+1}$
end procedure

In the next step, the models to be employed are adjusted as follows:

(i) Safety cells are appended to the left and the right of the patch of cells where the complex system is employed. Since shocks can travel from once cell to the next in one time step, the presence of safety cells avoids a setting where shock impinges an interface.

(ii) The patch of cells where simple system is employed cannot be a single cell. Such a patch will not lead to a significant reduction in computational resources. Furthermore, there is a high likelihood that in the cell where the simple system is employed, that the complex system would be needed in the next few time steps since the complex system needs to be employed in the surrounding cells. This could cause frequent switching between the two systems, which justifies the prevention of the use of the simple system in patches consisting of single cells.

Remark 3.4.3. In Section 3.3.6, we proposed a numerical flux to be employed at interfaces. The numerical flux is expected to work well as long as the numerical solution is sufficiently close to the equilibrium manifold and the dynamics is close to equilibrium dynamics in the cells at the interfaces where the complex system is employed. The model adaptation strategy precisely evaluates if these criteria are satisfied. Hence, the strategy is expected to avert placement of interfaces at points in computational domain which would lead to the creation of artefacts.

Chapter 4

Chemically reacting flows

A practically relevant example which fits into the framework discussed in Chapter 2 is that of chemically reacting flows. In this chapter, the modeling of chemically reacting flows is discussed.

Outline

The outline of the chapter is as follows

Section 4.1: Multi-component Euler equations can be employed to describe chemically reacting flows. In this section, the governing equations and the equation of state are specified for the case of ideal mixtures.

Section 4.2: Along with the equation of state, the reaction rates need to be specified to complete the description of chemically reacting fluid mixtures. In this section, the reaction rates employing mass action kinetics are defined.

Section 4.3: Since the modeling of chemically reacting flows employing continuum mechanics is grounded in thermodynamics, the system is equipped with an entropic structure. The entropic structure is discussed in this section.

Section 4.4: We apply the mesh and model adaptation strategy to chemically reacting flows. A model hierarchy is defined and the compliance of the model hierarchy with the abstract framework discussed in Chapter 2 is assessed.

4.1 Constitutive relations

Consider a multi-component fluid mixture with miscible (chemical) constituents $A_1, A_2, \ldots, A_{\mathcal{N}_c}$. The constituents undergo $\mathcal{N}_r$ chemical reactions leading to consumption of the reactants and creation of products which can be formally expressed as

$$\sum_{i=1}^{\mathcal{N}_c} \alpha_i^j A_i \rightleftharpoons \sum_{i=1}^{\mathcal{N}_c} \beta_i^j A_i \ , \ j = 1, \ldots, \mathcal{N}_r$$

where $\alpha_i^j, \beta_i^j \geq 0$ are the stoichiometric coefficients of the reactions.

The time evolution of chemically reacting flows can be modeled by continuum physics with different levels of complexity [10], namely Class-I, Class-II and Class-III models. The primitive variables of the Class-I models are partial mass densities of the constituents $\rho_i, i = 1, \ldots, \mathcal{N}_c$, the barycentric velocity v and the temperature T of the mixture, while that of the Class-II and Class-III models are partial densities, partial velocities, temperature and partial densities, partial velocities and partial temperatures respectively. The type of model to be employed depends on the application, for example modeling of plasma requires the use of Class-III model. We restrict ourselves to chemically reacting flows that can be described by Class-I model.

Assuming that the fluid mixture occupies a space $\Omega \subset \mathbb{R}$, balance laws can be derived from continuum physics to describe the time evolution of the partial mass densities ρ_i, the total momentum ρv and total energy ρE of the fluid mixture, where the total mass density is $\rho = \sum_{i=1}^{\mathcal{N}_c} \rho_i$ and v is the barycentric velocity. The derived governing equations based on the universal principles of conservation of total mass, total momentum and total energy give rise to an unclosed system of equations. The governing equations need to be supplemented by closure relations by providing information about the equation of state and the chemical reaction rates. The constitutive relations are derived based on the principles of continuum thermodynamics to ensure balance of entropy ensuring compliance with the second law of thermodynamics.

In this section, the governing equations and the constitutive relations for inviscid chemically reacting flows employing a Class-I model of non-equilibrium thermodynamics is discussed. We only provide a general outline and description of the governing equations and the constitutive modeling. For details of the systematic derivation of the closure relations based on thermodynamics principles for Class-I and Class-II models for viscous and inviscid fluid flows, we refer to the work of Bothe and Dreyer [10] and the text book [45]. For analysis on the closed system of equations refer to [52] and [74] and the text books [27] and [72] for the principles of thermodynamics.

The equations describing the local balance of partial mass densities ρ_i for $i = 1, \ldots, \mathcal{N}_c$, total momentum ρv and total energy ρE in multi-component reactive flows are as follows

$$\partial_t \rho_i + \partial_x (\rho_i v) = M_i \sum_{j=1}^{\mathcal{N}_r} \nu_i^j \mathfrak{R}_j, \tag{4.1}$$

$$\partial_t (\rho v) + \partial_x \left(\rho v^2 + p \right) = 0, \tag{4.2}$$

$$\partial_t (\rho E) + \partial_x \left((p + \rho E)\, v \right) = 0, \tag{4.3}$$

where $\nu_i^j = \beta_i^j - \alpha_i^j$, M_i is the molecular mass of constituent A_i and p is the total pressure of the fluid mixture. Since the chemical reactions are reversible, the source term which determines the consumption and production of the constituents is determined by $\mathfrak{R}_j := \mathfrak{R}_j^f - \mathfrak{R}_j^b$, where $\mathfrak{R}_j^f$ is the forward reaction rate and $\mathfrak{R}_j^b$ the backward reaction rate.

In order to ensure conservation of total mass, the stoichiometric coefficients are such that

$$\sum_{i=1}^{\mathcal{N}_c} M_i \nu_i^j = 0 \text{ for } j = 1, \ldots, \mathcal{N}_r. \tag{4.4}$$

leading to conservation of total mass

$$\partial_t \rho + \partial_x(\rho v) = 0. \tag{4.5}$$

The total energy of the fluid mixture ρE can be written as the sum of total internal energy and kinetic energy

$$\rho E = \rho e + \rho k, \;\; k := \frac{1}{2} v^2, \tag{4.6}$$

where k is the kinetic energy density. The pressure p and the total internal energy ρe can be written as the sum of partial pressures p_i and partial internal energies ρe_i of the constituents, i.e.

$$p = \sum_{i=1}^{\mathcal{N}_c} p_i, \quad \rho e = \sum_{i=1}^{\mathcal{N}_c} \rho e_i. \tag{4.7}$$

The closure employed needs to be consistent with two standard principles: the principle of material frame indifference, i.e. the constitutive functions remain invariant under change of reference and the entropy principle, i.e. compliance with the second law of thermodynamics. Under these constitutive principles an entropy is defined, where the dependence is restricted according to

$$\rho s = \rho \tilde{s} \left(\rho e, \rho_1, \ldots, \rho_{\mathcal{N}_c} \right), \tag{4.8}$$

where $\rho \tilde{s}$ is a strictly concave function. The concavity postulate characterises the condition that the free energy of the system has to always decrease and reach its minimum at equilibrium [10].

From the above defined entropy, the temperature T and the chemical potential μ_i for constituent A_i is defined as

$$\frac{1}{T} := \frac{\partial \rho \tilde{s}}{\partial \rho e}, \quad -\frac{\mu_i}{T} := \frac{\partial \rho \tilde{s}}{\partial \rho_i}. \tag{4.9}$$

In general, the principle of material frame indifference and the entropy principle are not sufficient to close the system of equations. Additionally, constitutive functions for partial internal energies and partial pressures are needed.

In the special case of simple mixtures, the dependence of the partial internal energy and pressure is restricted, allowing it to be calculated from the Helmholtz free energy [10],[74].

Definition 4.1.1. *A mixture of fluids with $\mathcal{N}_c$ constituents is called a simple mixture if the dependence of the partial internal energy densities and partial pressures is of the form*

$$e_i = \tilde{e}_i \left(T, \rho_i \right), \quad p_i = \tilde{p}_i \left(T, \rho_i \right). \tag{4.10}$$

For Class-I models, the Helmoholtz free energy $\psi = \tilde{\psi}\left(T, \rho_1, \ldots, \rho_{\mathcal{N}_c}\right)$ needs to be known. The partial quantities can be subsequently calculated from the $\tilde{\psi}$ with the help of the relations

$$e_i = -T^2 \frac{\partial}{\partial T}\left(\frac{\tilde{\psi}}{T}\right), \quad s_i = -\frac{1}{\rho_i}\frac{\partial \rho_i \tilde{\psi}_i}{\partial T}, \tag{4.11}$$

$$\mu_i = \frac{\partial \rho_i \tilde{\psi}_i}{\partial \rho_i}, \quad p_i = -\rho_i \tilde{\psi}_i + \rho_i \mu_i. \tag{4.12}$$

In the case of simple mixture of ideal gases, the Helmholtz free energy is given by

$$\tilde{\psi}_i := e_{0,i} + z_i \frac{R}{M_i}\left(T - T^R\right) - T \cdot \left(s_i^R + z_i \frac{R}{M_i} \ln\left(\frac{T}{T^R}\right) - \frac{R}{M_i}\ln\left(\frac{\rho_i}{\rho_i^R}\right)\right), \tag{4.13}$$

where $z_i = \frac{3}{2}, \frac{5}{2}$ or 3 for mono-atomic, diatomic and poly-atomic constituents respectively, $s_i^R, e_{0,i}^R$ are the reference specific internal energies and entropies at reference states T^R, ρ_i^R for the constituents and R is the universal gas constant, cf. [10] [27]. The reference values can be found in handbooks of chemistry and physics [16] and [63].

Let

$$x_i = \frac{c_i}{c} \quad \text{where } c_i = \frac{\rho_i}{M_i}, c = \sum_{k=1}^{\mathcal{N}_c} c_k \tag{4.14}$$

be the molar concentration of species A_i, for $i = 1, \ldots, \mathcal{N}_c$.

Then, from (4.13) we have

$$e_i = e_{0,i} + c_{v,i}\left(T - T^R\right), \tag{4.15}$$

where

$$c_{v,i} := \frac{\partial e_i}{\partial T} = z_i \frac{R}{M_i}, \; c_{p,i} := c_{v,i} + R \tag{4.16}$$

is the specific heat at constant volume and constant pressure respectively.

The partial entropy density is given by

$$s_i = s_i^R + c_{v,i} \ln\left(\frac{T}{T^R}\right) - \frac{R}{M_i}\ln\left(\frac{\rho_i}{\rho_i^R}\right) \tag{4.17}$$

and the chemical potential of each constituent is given by

$$\mu_i = g_i(T,p) + \tfrac{RT}{M_i}\ln(x_i)\,, \qquad \text{where} \tag{4.18}$$

$$g_i(T,p) = e_{0,i} - T s_i^R + c_{v,i}\left(T - T^R\right) - c_{p,i}\ln\left(\tfrac{M_i}{RT^R\rho_i^R}p\right) \tag{4.19}$$

is the Gibbs free energy.

Finally, the partial pressure is given by

$$p_i = \frac{R}{M_i}\rho_i T. \tag{4.20}$$

and the entropy of the fluid mixture can be calculated from the partial specific entropy densities by $\rho s = \sum_{i=1}^{\mathcal{N}_c} \rho_i s_i$.

Remark 4.1.2. The mixtures which can be described by the closure discussed above are referred to as ideal mixtures. We can observe that with the closure specified above, the total pressure of the system is given by

$$p = RT \sum_{i=1}^{\mathcal{N}_c} \frac{\rho_i}{M_i} \tag{4.21}$$

which is the well-known ideal gas law [27].

4.2 Chemical reaction kinetics

Closure for the mass production, i.e. the forward and backward reaction rates $\mathfrak{R}_j^f$ and $\mathfrak{R}_j^b$ needs to be provided. This is done by specifying a relation between the forward and the backward reaction rates and modelling the reaction rates by employing so called mass action kinetics.

To this end, let

$$\mathcal{A}_j := \sum_{i=1}^{\mathcal{N}_c} \nu_i^j M_i \mu_i, \tag{4.22}$$

be the so called chemical affinity of the fluid mixture for reaction j. Based on kinetic theory, [45], the forward and backwards reaction rate is closed by employing the following non-linear relation

$$ln\left(\frac{\mathfrak{R}_j^f}{\mathfrak{R}_j^b}\right) = -\gamma_j \frac{\mathcal{A}_j}{RT}. \tag{4.23}$$

The factor γ_j is often chosen to be one, but is non-negative and may depend on the variables of the system of equations [30],[32]. The relation (4.23) can be viewed as a compatibility condition between chemistry and thermodynamics. Namely, in Section 4.3.1 we show that the closure leads to non-negative entropy production.

The reaction rates, employing mass action kinetics, are modeled by

$$R_j^f = k_j^f \prod_{i=1}^{\mathcal{N}_c} x_i^{\alpha_i^j}, \; R^b = k_j^b \prod_{i=1}^{\mathcal{N}_c} x_i^{\beta_i^j}, \tag{4.24}$$

where $k_j^f > 0$ and $k_j^b > 0$ are the forward and backward reaction rate constants. The reaction rate constants depend on temperature and pressure. The reaction rates modeled by mass action kinetics are applicable for elementary reactions, where the constituents react in one single step to give products and the characteristic times of the chemical reactions are larger than the mean free time of the molecules [45].

We can note that either the backward or forwards reaction rate constant needs to be specified. The other reaction rate constant can then be determined from (4.23) and (4.18).

For the chemical reactions, equilibrium rate constant is defined as

$$k_j^{\text{eq}} = \frac{k_j^f}{k_j^b} = \exp\left(-\frac{1}{RT}\sum_{i=1}^{\mathcal{N}_c} \nu_i^j M_i g_i\right) \tag{4.25}$$

The modeling of the forward reaction rate is application dependent, but in most cases strongly depends on temperature. Some examples of the modeling of the reactions rates applied to electro-chemistry, hypersonic air flows and combustion can be found in [32], [46], [78] and [96].

4.3 Entropy Structure

4.3.1 Entropy balance law

Next, we derive the entropy balance law and show that the entropy production due to the chemical reactions is always non-negative. To this end, first we derive some identities for balance of internal energy and conservation of total mass and momentum.

To derive the identities, we make use of the material derivative defined below.

Definition 4.3.1. *For a macroscopic thermodynamic quantity $y : \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}$ which depends on time and space the material derivative is defined by*

$$\mathrm{D_t}\, y := \partial_t y + v \cdot \partial_x y \tag{4.26}$$

where v is the velocity.

From the balance of the partial specific densities (4.1), conservation of total momentum (4.2) and total energy (4.5) we have

$$\mathrm{D_t}\, \rho_i = -\rho_i \partial_x v + M_i \sum_{j=1}^{\mathcal{N}_r} \nu_i^j \mathfrak{R}_j, \tag{4.27}$$

$$\mathrm{D_t}\, \rho = -\rho \partial_x v, \tag{4.28}$$

and

$$\mathrm{D_t}\, \rho v = -\rho v \partial_x v - \partial_x p. \tag{4.29}$$

Expanding the total energy as sum of internal and kinetic energy in (4.3) we get

$$\mathrm{D_t}\, \rho e = -\partial_t\left(\frac{1}{2}\rho v^2\right) - \partial_x\left(\frac{1}{2}\rho v^2 \cdot v\right) - (\rho e + p)\,\partial_x v - v\partial_x p. \tag{4.30}$$

Note that for fluid mixtures that can be described by a Class-I model, [10], [27], the Gibbs-Duhem equation relates the chemical potential to the entropy and the system variables as follows

$$p + \rho e = \rho s \cdot T + \sum_{i=1}^{\mathcal{N}_c} \mu_i \rho_i. \tag{4.31}$$

Employing (4.31), (4.27) and (4.28) in (4.30) gives

$$\mathrm{D_t}\,\rho e = -\rho s T \cdot \partial_x v - \sum_{i=1}^{\mathcal{N}_c} \mu_i \rho_i \partial_x v. \tag{4.32}$$

Next, we derive a balance law for the entropy.

Differentiating (4.8) with respect to time gives

$$\frac{\partial \rho s}{\partial t} = \frac{\partial \rho \tilde{s}}{\partial \rho e} \cdot \frac{\partial \rho e}{\partial t} + \sum_{i=1}^{\mathcal{N}_c} \frac{\partial \rho \tilde{s}}{\partial \rho_i} \cdot \frac{\partial \rho_i}{\partial t}, \tag{4.33}$$

which, employing (4.9), simplifies to

$$\frac{\partial \rho s}{\partial t} = \frac{1}{T}\frac{\partial \rho e}{\partial t} - \sum_{i=1}^{\mathcal{N}_c} \frac{\mu_i}{T} \cdot \frac{\partial \rho_i}{\partial t}. \tag{4.34}$$

Analogously, differentiating entropy with respect to the space variable, we have

$$\frac{\partial \rho s}{\partial x} = \frac{1}{T}\frac{\partial \rho e}{\partial x} - \sum_{i=1}^{\mathcal{N}_c} \frac{\mu_i}{T} \cdot \frac{\partial \rho_i}{\partial x}. \tag{4.35}$$

Employing the transport equations for internal energy (4.34) and partial specific densities (4.35) gives

$$\begin{aligned} \partial_t \rho s + \partial_x(\rho s \cdot v) &= \partial_t \rho s + v \cdot \partial_x(\rho s) + \rho s \cdot \partial_x v \\ &= \frac{1}{T} \cdot \mathrm{D_t}\,\rho e - \sum_{i=1}^{\mathcal{N}_c} \left(\frac{\mu_i}{T} \cdot \mathrm{D_t}\,\rho_i \right) + \rho s \cdot \partial_x v. \end{aligned} \tag{4.36}$$

which simplifies to

$$\partial_t \rho s + \partial_x(\rho s \cdot v) = -\frac{1}{T} \sum_{i=1}^{\mathcal{N}_r} \left(\mathfrak{R}_j^f - \mathfrak{R}_j^b \right) \cdot \mathcal{A}_j, \tag{4.37}$$

where we have employed (4.32) and (4.27).

Remark 4.3.2. The entropy production due to chemical reactions is

$$\zeta = -\frac{1}{T} \sum_{i=1}^{\mathcal{N}_r} \left(\mathfrak{R}_j^f - \mathfrak{R}_j^b \right) \cdot \mathcal{A}_j \tag{4.38}$$

which from (4.22) and (4.23) can be expressed as

$$\zeta = R \sum_{i=1}^{\mathcal{N}_r} \frac{1}{\gamma_j} \left(\mathfrak{R}_j^f - \mathfrak{R}_j^b \right) \cdot \left(\ln\left(\mathfrak{R}_j^f \right) - \ln\left(\mathfrak{R}_j^b \right) \right). \tag{4.39}$$

Since logarithm is a monotonously increasing function, we can infer that $\zeta \geq 0$. Consequently, the closures we employ are compliant with the second law of thermodynamics.

The fluid mixture is said to be in chemical equilibrium, if the reaction terms in the balance equations for densities in (4.1) vanish and the entropy production ζ vanishes. We can observe that the entropy production is zero if and only if all of the chemical reactions are individually in equilibrium, where the forward reaction rate equals the backward reaction rate. This is an instance of the principle of detailed balance [10], also referred to as the Wegscheider's condition in chemical kinetics.

4.3.2 Convexity of entropy

Let

$$\mathbf{U} := \begin{bmatrix} \rho_1 & \dots & \rho_{\mathcal{N}_c} & \rho v & \rho E \end{bmatrix}^T, \mathbf{F} := \begin{bmatrix} \rho_1 v & \dots & \rho_{\mathcal{N}_c} v & \rho v^2 + p & \left(\rho e + \frac{1}{2}\rho v^2 + p\right) v \end{bmatrix}^T. \tag{4.40}$$

In order to derive the constitutive relations, strict concavity of the entropy was assumed in (4.8) with respect to the internal energy and partial densities. Let $H := -\rho s$, then, we show that ρs is strictly concave with respect to $\mathbf{U}$ by showing that the Hessian of $H(\mathbf{U})$ is a symmetric positive definite matrix. This approach was employed by Tadmor et. al. to show strict convexity for multi-component Euler equations in [51].

Remark 4.3.3. Note that H and Q are referred to as the mathematical entropy and mathematical entropy flux and ρs and $\rho s v$ as the physical entropy and physical entropy flux.

The Hessian of H can be expressed as

$$\mathcal{H} := \frac{\partial^2 H}{\partial \mathbf{U}^2} = \frac{\partial \mathbf{v}}{\partial \mathbf{U}} = \frac{\partial \mathbf{v}}{\partial \mathbf{Z}} \left(\frac{\partial \mathbf{U}}{\partial \mathbf{Z}}\right)^{-1}. \tag{4.41}$$

where

$$\mathbf{v} := (\mathrm{D}\, H)^T, \ \mathbf{Z} = \begin{bmatrix} \rho_1 & \dots & \rho_{\mathcal{N}_c} & v & T \end{bmatrix}^T \tag{4.42}$$

are the so called entropy and primitive variables [45].

The Gibbs-Duhem equation can be expressed in the form [51]

$$T\, \mathrm{d}\, H = -\,\mathrm{d}\, \rho e + \sum_{i=1}^{\mathcal{N}_c} \mu_i\, \mathrm{d}\, \rho_i \tag{4.43}$$

and from the definition of internal energy (4.7) we have

$$\mathrm{d}\, \rho e = \sum_{i=1}^{\mathcal{N}_c} e_i\, \mathrm{d}\, \rho_i + \rho c_v\, \mathrm{d}\, T. \tag{4.44}$$

Employing (4.43) and (4.44) leads to

$$\mathrm{d}\, H = \frac{1}{T}\left(\sum_{i=1}^{\mathcal{N}_c} (\mu_i - e_i)\, \mathrm{d}\, \rho_i - \rho c_v\, \mathrm{d}\, T\right). \tag{4.45}$$

Hence, the Jacobian of H with respect to the primitive variables is given by

$$\mathrm{D}_Z\, H = \frac{1}{T}\begin{bmatrix} \mu_1 - e_1 & \dots & \mu_{\mathcal{N}_c} - e_{\mathcal{N}_c} & 0 & -\rho c_v \end{bmatrix} \tag{4.46}$$

and the Jacobian of the mapping from $\mathbf{U}$ to the primitive variables and its inverse, [15],

is given by

$$\frac{\partial \mathbf{U}}{\partial \mathbf{Z}} = \begin{bmatrix} 1 & & 0 & 0 & 0 \\ & \ddots & & \vdots & \vdots \\ 0 & & 1 & 0 & 0 \\ v & \dots & v & \rho & 0 \\ e_1 + k & \dots & e_{\mathcal{N}_c} + k & \rho v & \rho c_v \end{bmatrix}, \tag{4.47}$$

$$\left(\frac{\partial \mathbf{U}}{\partial \mathbf{Z}}\right)^{-1} = \begin{bmatrix} 1 & & 0 & 0 & 0 \\ & \ddots & & \vdots & \vdots \\ 0 & & 1 & 0 & 0 \\ -\frac{v}{\rho} & \dots & -\frac{v}{\rho} & \frac{1}{\rho} & 0 \\ \frac{k-e_1}{\rho c_v} & \dots & \frac{k-e_{\mathcal{N}_c}}{\rho c_v} & -\frac{v}{\rho c_v} & \frac{1}{\rho c_v} \end{bmatrix}, \tag{4.48}$$

where k is the kinetic energy as defined in (4.6).

From (4.47) and (4.46) we have

$$\mathbf{v} = \frac{1}{T}\begin{bmatrix} \mu_1 - k & \dots & \mu_{\mathcal{N}_c} - k & v & -1 \end{bmatrix}^T. \tag{4.49}$$

Furthermore, the Jacobian of the entropy variables with respect to the primitive variables is given by

$$\frac{\partial \mathbf{v}}{\partial \mathbf{Z}} = \begin{bmatrix} \frac{R}{\rho_i M_i} & & 0 & -\frac{v}{T} & \frac{k-e_1}{T^2} \\ & \ddots & & \vdots & \vdots \\ 0 & & \frac{R}{\rho_{\mathcal{N}_c} M_{\mathcal{N}_c}} & -\frac{v}{T} & \frac{k-e_{\mathcal{N}_c}}{T^2} \\ 0 & \dots & 0 & \frac{1}{T} & -\frac{v}{T^2} \\ 0 & \dots & 0 & 0 & \frac{1}{T^2} \end{bmatrix}. \tag{4.50}$$

Employing (4.50) and (4.47) in (4.41) leads to

$$\mathcal{H} = \frac{1}{\rho c_v T^2}\begin{bmatrix} & & & -v(k-(e_1-c_vT)) & -(e_1-k) \\ & \xi_{ij} & & \vdots & \vdots \\ & & & -v(k-(e_{\mathcal{N}_c}-c_vT)) & -(e_{\mathcal{N}_c}-k) \\ -v(k-(e_1-c_vT)) & \dots & -v(k-(e_{\mathcal{N}_c}-c_vT)) & v^2+c_vT & -v \\ -(e_i-k) & \dots & -(e_{\mathcal{N}_c}-k) & -v & 1 \end{bmatrix} \tag{4.51}$$

where for $1 \leq i, j \leq \mathcal{N}_c$

$$\xi_{ij} = \rho c_v T^2 \left(\frac{\delta_{ij}}{\rho_i} + v^2 c_v T\right) + (e_i - k)(e_j - k). \tag{4.52}$$

The positive definiteness of the Hessian can be shown by multiplying the matrix from the left and right by the Jacobian of the basic variables with respect to the primitive variables

leading to

$$\mathcal{G} = \left(\frac{\partial \mathbf{U}}{\partial \mathbf{Z}}\right)^T \mathcal{H} \left(\frac{\partial \mathbf{U}}{\partial \mathbf{Z}}\right) = \left(\frac{\partial \mathbf{U}}{\partial \mathbf{Z}}\right)^T \frac{\partial \mathbf{v}}{\partial \mathbf{Z}} \tag{4.53}$$

$$= \begin{bmatrix} \frac{R}{\rho_1 M_1} & & 0 & 0 & 0 \\ & \ddots & & \vdots & \vdots \\ 0 & & \frac{R}{\rho_{\mathcal{N}_c} M_{\mathcal{N}_c}} & 0 & 0 \\ 0 & \dots & 0 & \frac{\rho}{T} & 0 \\ 0 & \dots & 0 & 0 & \frac{\rho c_v}{T^2}. \end{bmatrix} \tag{4.54}$$

Note that, since for $i = 1, \dots \mathcal{N}_c$ the partial densities satisfy $\rho_i > 0$ and $T, \rho c_v > 0$, the matrix $\mathcal{G}$ is positive definite and consequently due to the congruence relation the Hessian $\mathcal{H}$ is also positive definite.

4.4 Abstract framework

In this section, we cast the governing equations for chemically reacting flows in the abstract form discussed in Section 2.2 in Chapter 2.

4.4.1 Complex system

The system of equations formed by employing equations (4.1)-(4.3) and employing the closure discussed in Section 4.2 is assigned to be the complex system. This results in a system of size $N = \mathcal{N}_c + 2$ and the vector of variables and the fluxes are as defined in (4.40). Furthermore, the source term is given by

$$\frac{1}{\varepsilon}\mathbf{R}(\mathbf{U}) := \begin{bmatrix} M_1 \sum_{j=1}^{\mathcal{N}_r} \nu_1^j \mathfrak{R}_j & \dots & M_{\mathcal{N}_c} \sum_{j=1}^{\mathcal{N}_r} \nu_{\mathcal{N}_c}^j \mathfrak{R}_j & 0 & 0 \end{bmatrix}^T \tag{4.55}$$

Qualitatively, when employing the complex system, we model the ideal mixture by tracking the time evolution of the constituents $\mathcal{N}_c$, total momentum and total energy of the chemically reacting fluid mixture.

Remark 4.4.1. Note that for $j = 1, \dots, \mathcal{N}_r$, the reaction term $\mathfrak{R}_j$ can be expressed as

$$\mathfrak{R}_j = k_j^f \prod_{i=1}^{\mathcal{N}_c} x_i^{\alpha_i^j} - k_j^b \prod_{i=1}^{\mathcal{N}_c} x_i^{\beta_i^j} \tag{4.56}$$

$$= k_j^f \left(\prod_{i=1}^{\mathcal{N}_c} x_i^{\alpha_i^j} - \frac{1}{k_j^{eq}} \prod_{i=1}^{\mathcal{N}_c} x_i^{\beta_i^j} \right). \tag{4.57}$$

The forward reaction rate is generally assumed to be of the form $k_j^f := C_j T^{-\zeta_j} \exp\left(-\frac{E_j}{T}\right)$, where $\zeta_j, C_j, E_j \geq 0$ are constants and the k_j^{eq} can be calculated from (4.25).

The forward reaction rate constant can then be expressed as

$$k_j^f := \left(C_j T_r^{-\zeta_j} \exp\left(\frac{-E_j}{T_r}\right)\right) \cdot \left(\left(\frac{T}{T_r}\right)^{-\zeta_j} \cdot \exp\left(\frac{-T_r}{T}\right)\right), \tag{4.58}$$

where T_r is the reference temperature.

Let

$$\tilde{k}_j^f := C_j T_r^{-\zeta_j} \exp\left(\frac{-E_j}{T_r}\right) \tag{4.59}$$

be the the scaled forward reaction rate constant for reactions $\mathfrak{R}_j$. The scaled forward reaction rate constants characterise the speed of the individual reactions. The reference temperature T_r is introduced to account for the strong dependence of the speed of the reactions on the temperature of the fluid mixture.

For chemically reacting flows with multiple reactions, let

$$\varepsilon := \max_{j=1,\dots,\mathcal{N}_r} \left(\frac{1}{\tilde{k}_j^f}\right). \tag{4.60}$$

The maximum value is chosen since it corresponds to the slowest reaction speed, for which, we need to be most cautious when switching to the simple system.

4.4.2 Simple system

The governing equations can be simplified by assuming that the fluid mixture is in chemical equilibrium. We need to define the vector of conserved variables $\mathbf{u} \in \mathbb{R}^n$ and the Maxwellian for which the relation

$$\mathbb{P}M(\mathbf{u}) = \mathbf{u} \tag{4.61}$$

should hold. To this end, we exploit some physical principles and their implications. First, we discuss ways to enforce chemical equilibrium and conservation of mass, followed by providing definitions of the simple system. The definition of the simple system depends on the number of reactions, the number of constituents and the number of non-catalyst constituents that make up the fluid mixture. In the rest of the section, we assume that there is not reaction which is independent. By independent reaction, we mean a reaction such that the reactants and the products involved in that particular reaction do not appear in any of the other reactions. Furthermore, note that in the cases we discuss, the way to define the simple system may not be unique, but we present one way to do so.

Remark 4.4.2. Chemical compounds that precipitate chemical reaction but do not in themselves undergo chemical transformation through those reactions are called catalysts. Formally, if constituent A_i is such that

$$\nu_i^j = 0, \text{ for } j = 0, \dots, \mathcal{N}_r \tag{4.62}$$

holds, then it is referred to as a catalyst. We denote the number of non-catalyst constituents in the fluid mixture by $\mathcal{N}_{cr}$. Note that (4.62) implies $M_i \sum_{j=1}^{\mathcal{N}_r} \nu_i^j \mathfrak{R}_j = 0$, i.e. catalysts are always conserved, irrespective of the fluid being in chemical equilibrium.

Chemical equilibrium

There are two approaches by which chemical equilibrium can be imposed. For all of the cases for which we explain how to define the simple system, one of the two approaches needs to be employed in order to ensure that the number of equations match the number of variables when computing the Maxwellian.

The fluid mixture is in chemical equilibrium when the reaction terms of the complex system vanish. Then, assuming $\mathcal{N}_{cr} = \mathcal{N}_c$, chemical equilibrium is characterised by

$$M_i \sum_{j=1}^{\mathcal{N}_r} \nu_i^j \mathfrak{R}_j = 0, \text{ for } i = 1, \ldots, \mathcal{N}_c - 1, \tag{4.63}$$

which forms a system of non-linear equations of size $\mathcal{N}_c - 1$.

Remark 4.4.3. Since (4.4) holds for stoichiometric coefficients in the reaction terms, (4.63) implies $M_{\mathcal{N}_c} \sum_{j=1}^{\mathcal{N}_r} \nu_{\mathcal{N}_c}^j \mathfrak{R}_j = 0$. Hence, the choice of constituents for which (4.63) is enforced is arbitrary.

A more stringent notion of equilibrium is the principle of detailed balance, which states that for the fluid mixture to be in chemical equilibrium, each of the reactions, individually, needs to be in chemical equilibrium, i.e.

$$\mathfrak{R}_j = 0 \text{ for } j = 1, \ldots, \mathcal{N}_r. \tag{4.64}$$

The system of non-linear equations formed by (4.64) is of size $\mathcal{N}_r$.

Conservation of constituent elements

The principle of conservation of mass implies that the total mass of individual elements that comprise the constituents and the total mass of the fluid mixture has to be conserved.

Assuming that the chemical constituents are made up of chemical elements $a^1, \ldots, a^{\mathcal{N}_e}$, for $i = 1, \ldots, \mathcal{N}_c$, the constituent A_i, can be represented as $a^1_{\xi_i^1} a^2_{\xi_i^2} \ldots a^{\mathcal{N}_e}_{\xi_i^{\mathcal{N}_e}}$, employing the standard nomenclature of chemistry [67], [68]. For example let constituent A_1 be water, then $A_1 := \mathrm{H_2O}$ and $a^1 := \mathrm{H}$ and $a^2 := \mathrm{O}$ and $\xi_1^1 = 2, \xi_1^2 = 1$. In this case, conservation of mass implies that the mass of atomic oxygen (O) and atomic hydrogen (H) will be conserved, irrespective of the fluid mixture being in chemical equilibrium or not.

This notion can be formalized by requiring

$$\sum_{i=1}^{\mathcal{N}_c} \alpha_i^j \xi_i^k = \sum_{i=1}^{\mathcal{N}_c} \beta_i^j \xi_i^k, \text{ for } j = 1, \ldots, \mathcal{N}_r \text{ and } k = 1, \ldots, \mathcal{N}_e. \tag{4.65}$$

Definition of the simple system

The definition of the simple system can be split into two cases, where conservation of mass and chemical equilibrium is enforced in different forms.

Case I: $\mathcal{N}_r \geq \mathcal{N}_c$

If the number of reactions is greater than or equal to the number of constituents, then the definition of the simple system is split into two cases:

- Case Ia: $\mathcal{N}_{cr} = \mathcal{N}_c$
- Case Ib: $\mathcal{N}_{cr} < \mathcal{N}_c$

For both of these cases, chemical equilibrium is imposed in the form (4.63).

Case Ia: $\mathcal{N}_{cr} = \mathcal{N}_c$

The projection matrix $\mathbb{P} \in \mathbb{R}^{n \times N}$, where $n = 3, N = \mathcal{N}_c + 2$, is defined as

$$\mathbb{P} := \begin{bmatrix} 1 & \dots & 1 & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & 1 \end{bmatrix}. \tag{4.66}$$

In this case, the governing equations of the simple system consists of conservation laws for the total mass, momentum and total energy of the fluid mixture. Since the total mass (ρ), the momentum (ρv) and the total energy (ρE) of the fluid mixture is conserved, evidently $\mathbb{P}\mathbf{R}(\mathbf{U})$.

The conversion of a vector of variables $\mathbf{u} \in \mathbb{R}^n$ of the simple system to a vector of variables of the complex system $\mathbf{U} \in \mathbb{R}^N$ can be done by employing (4.61) and (4.63), giving rise to a system of equations of size $\mathcal{N}_c + 2$.

Case Ib: $\mathcal{N}_{cr} < \mathcal{N}_c$

If the number of catalysts in the fluid mixture is non-zero, then the definition of the simple system is slightly modified, where along with the conservation of total mass, momentum and energy of the fluid mixture, we need to solve for the conservation of the catalysts. As an example let constituents A_1 and $A_{\mathcal{N}_c}$ be catalysts, then the projection matrix $\mathbb{P} \in \mathbb{R}^{5 \times \mathcal{N}_c + 2}$ is defined as

$$\mathbb{P} := \begin{bmatrix} 1 & & \dots & & 1 & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & 0 \\ 0 & & \dots & & 0 & 0 & 1 \end{bmatrix}. \tag{4.67}$$

In this case, we can also define the projection matrix as

$$\mathbb{P} := \begin{bmatrix} 0 & 1 & \dots & 1 & 0 & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 & 0 \\ 0 & & \dots & & 0 & 0 & 1 \end{bmatrix}, \tag{4.68}$$

If the projection matrix as defined in (4.68) is employed, then the governing equations of the simple system consist of conservation of total mass of non-catalysts, total mass of catalysts, momentum and total energy of the fluid mixture.

Case *II*: $\mathcal{N}_r < \mathcal{N}_c$

Recall that $\mathcal{N}_e$ denotes the total number of elements that make up the constituents in the fluid mixture. Then, if the number of reactions is less than the number of constituents, the definition of the simple system can be further split into three cases:

- Case *IIa*: $\mathcal{N}_e = \mathcal{N}_c - \mathcal{N}_r$
- Case *IIb*: $\mathcal{N}_e > \mathcal{N}_c - \mathcal{N}_r$
- Case *IIc*: $\mathcal{N}_e < \mathcal{N}_c - \mathcal{N}_r$

For the cases listed above, chemical equilibrium is imposed in the form (4.64).

Case IIa: $\mathcal{N}_e = \mathcal{N}_c - \mathcal{N}_r$

Let $\mathcal{N}_e = \mathcal{N}_c - \mathcal{N}_r$, then the projection matrix is defined as

$$\mathbb{P} := \begin{bmatrix} \frac{\xi_1^1 m_1}{M_1} & \dots & \frac{\xi_{\mathcal{N}_c}^1 m_1}{M_{\mathcal{N}_c}} & 0 & 0 \\ \vdots & & \vdots & \vdots & \vdots \\ \frac{\xi_1^{\mathcal{N}_e} m_{\mathcal{N}_e}}{M_1} & \dots & \frac{\xi_{\mathcal{N}_c}^{\mathcal{N}_e} m_{\mathcal{N}_e}}{M_{\mathcal{N}_c}} & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & 1 \end{bmatrix}, \tag{4.69}$$

where $m_i, \dots, m_{\mathcal{N}_e}$ are the atomic masses of the elements $a_1, \dots, a_{\mathcal{N}_e}$. In this case, when employing the simple system, we model the ideal mixture by assuming the fluid to be in chemical equilibrium and tracking the time evolution of the densities of the chemical elements that make up the chemical constituents, the total momentum and the total energy of the fluid mixture.

Note that, the relation (4.61) along with (4.64) leads to a system of non-linear equations of size $\mathcal{N}_c + 2$, providing a complete system of equations to convert the variables of the simple system to the complex system.

Next, we show that $\mathbb{P}\mathbf{R}(\mathbf{U}) = 0$.

From (4.69) and (4.55) we have

$$\mathbb{P}\left(\frac{1}{\varepsilon}\mathbf{R}(\mathbf{U})\right) = \left[\sum_{i=1}^{\mathcal{N}_c} m_1\xi_i^1 \sum_{j=1}^{\mathcal{N}_r}\left(\beta_i^j-\alpha_i^j\right)\mathfrak{R}_j \quad \dots \quad \sum_{i=1}^{\mathcal{N}_c} m_{\mathcal{N}_e}\xi_i^{\mathcal{N}_e} \sum_{j=1}^{\mathcal{N}_r}\left(\beta_i^j-\alpha_i^j\right)\mathfrak{R}_j \quad 0 \quad 0\right]^T \tag{4.70}$$

$$= \left[m_1 \sum_{j=1}^{\mathcal{N}_r}\mathfrak{R}_j \sum_{i=1}^{\mathcal{N}_c}\xi_i^1\left(\beta_i^j-\alpha_i^j\right) \quad \dots \quad m_{\mathcal{N}_e} \sum_{j=1}^{\mathcal{N}_r}\mathfrak{R}_j \sum_{i=1}^{\mathcal{N}_c}\xi_i^{\mathcal{N}_e}\left(\beta_i^j-\alpha_i^j\right) \quad 0 \quad 0\right]^T \tag{4.71}$$

$$= 0,$$

where the first $\mathcal{N}_e$ entries of the vector vanish due to (4.65). Qualitatively, we employed the fact that mass of the $\mathcal{N}_e$ elements that constitute the reactants and the products, total energy and the momentum of the fluid mixture has to be conserved to show that the source term lies in the null space of the projection matrix.

Case IIb: $\mathcal{N}_e > \mathcal{N}_c - \mathcal{N}_r$

Let $\mathcal{N}_d = \mathcal{N}_c - \mathcal{N}_r$ then the projection matrix is defined as

$$\mathbb{P} := \begin{bmatrix} \frac{\xi_1^1 m_1}{M_1} & \dots & \frac{\xi_{\mathcal{N}_c}^1 m_1}{M_{\mathcal{N}_c}} & 0 & 0 \\ \vdots & & \vdots & \vdots & \vdots \\ \frac{\xi_1^{\mathcal{N}_d-1} m_{\mathcal{N}_d-1}}{M_1} & \dots & \frac{\xi_{\mathcal{N}_c}^{\mathcal{N}_d-1} m_{\mathcal{N}_d-1}}{M_{\mathcal{N}_c}} & 0 & 0 \\ 1 & \dots & 1 & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & 0 & 0 & 1 \end{bmatrix}. \tag{4.72}$$

In this case, the simple system involves tracking the time evolution of the densities of $\mathcal{N}_d - 1$ elements, the total mass of the fluid mixture, the momentum and the total energy of the fluid mixture. The choice of elements is arbitrary since we assume that there is no reaction which is independent and the time evolution of the rest of the constituents is accounted in the conservation of the total density, ρ, of the fluid mixture.

In this case, relation (4.61) provides a nonlinear system of equations of size $\mathcal{N}_c - \mathcal{N}_r + 2$ which along with (4.64) results in a non-linear system of equations of size $\mathcal{N}_c + 2$. Since the mass of all the elements that make up the chemical constituents and the total mass of the fluid mixture ρ is conserved, it implies $\mathbb{P}\mathbf{R}(\mathbf{U}) = 0$.

Case IIc: $\mathcal{N}_e < \mathcal{N}_c - \mathcal{N}_r$

A generic defintion of the simple system cannot be provided in this case. In addition to mass of the elements that make up the constituents, momentum and total energy of the fluid mixture, i.e as done in Case *IIa*, the vector of variables needs to be further expanded by $\mathcal{N}_c - \mathcal{N}_e - \mathcal{N}_r$ number of conserved variables. As a result, (4.61) and (4.64) will provide $\mathcal{N}_c + 2$ number of equations. The additional variables needs to be defined on a case by case basis by looking at chemical species that is conserved irrespective of the fluid mixture being in equilibrium or not.

Remark 4.4.4. The projection matrices defined above are constant and have to be determined once in accordance with the chemical reactions that take place and the chemical constituents and chemical elements involved. Condition (2.2) on the rank of the projection matrix (2.2) and the non-degeneracy conditions (2.4) can be checked on a case by case basis. In general, the likelihood of the projection matrix being rank deficient are slim due to the varied nature of the chemical constituents present in chemically reacting fluid mixtures.

4.4.3 Entropy structure

Recalling that we assumed the existence of a strictly convex entropy $H(\mathbf{U})$ and entropy flux $Q(\mathbf{U})$ in Chapter 2 such that

$$-\,\mathrm{D}\,H(\mathbf{U})\cdot\mathbf{R}(\mathbf{U})\geq 0,\;\;\mathrm{D}\,H(\mathbf{U})\cdot\mathrm{D}\,\mathbf{F}(\mathbf{U})=\mathrm{D}\,Q(\mathbf{U}). \tag{4.73}$$

We showed that the physical entropy ρs is strictly concave with respect to $\mathbf{U}$ and the entropy production ξ is always non-negative in Section 4.3. Furthermore as a consequence of the balance law (4.3.1) it can be shown that, see [45] and [73],

$$\mathrm{D}\,(\rho s)\cdot\mathrm{D}\,\mathbf{F}(\mathbf{U})=\mathrm{D}\,(\rho s v)\,. \tag{4.74}$$

Hence $H:=-\rho s$ and $Q:=-\rho s v$ form compatible entropy and entropy flux functions in the sense as defined in Section 2.2.1, hence providing the desired entropy structure discussed in Chapter 2. The pair (H,Q) is referred to as the mathematical entropy and mathematical entropy flux and $(\rho s,\rho s v)$ as the physical entropy and physical entropy flux.

Remark 4.4.5. All that remains is to determine the constant ν in the condition (2.39). Since it is not possible to provide an analytical expression for ν, we calculate it numerically on a case by case basis by looking at the set of states the numerical solution takes values in.

Chapter 5

Numerical Results

Outline

The outline of the chapter is as follows

Section 5.1: In this section, the numerical setup and the test case of chemically reacting flows with which we demonstrate and test the proposed mesh and model adaptation strategy is described.

Section 5.2: In this section, the numerical tests conducted to test the proposed coupling method are presented and their results are discussed. The behaviour of rarefactions, contact discontinuities and shocks as they impinge on a coupling interface is studied. To this end, we consider the case of a shock tube.

Section 5.3: In this section, the numerical tests conducted to test the model adaptation strategy are presented. We consider the case of a shock tube and an advecting reaction front.

Section 5.4: In this section, the numerical tests conducted to test the mesh adaptation strategy are presented. We consider the case of a shock tube.

Section 5.5: In this section, we discuss the results of simultaneous mesh and model adaptation simulations for the case of a shock tube.

5.1 Setup

5.1.1 Chemically reacting flows

We conduct the numerical experiments for the simple case of dissociation of oxygen, where the molecular oxygen dissociates into atomic oxygen with nitrogen acting as a catalyst:

$$O_2 + N_2 \rightleftharpoons 2O + N_2 \ .$$

The number of reactions, the number of constituents, the number of (non catalyst) reacting constituents and the number of elements is given by $\mathcal{N}_r = 1, \mathcal{N}_c = 3, \mathcal{N}_{cr} = 3$ and

$\mathcal{N}_e = 2$ respectively. Note that the problem at hand falls under the Case *IIa* defined in Section 4.4.2, with $N = 5, n = 4$.

The vector of variables of the complex system is given by $\mathbf{U} := \begin{bmatrix} \rho_{O_2} & \rho_O & \rho_{N_2} & \rho v & \rho E \end{bmatrix}^T$. According to the definition of model hierarchy for chemically reacting flows, see Section 4.4, the projection matrix is given by

$$\mathbb{P} := \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.1}$$

The vector of the conserved variables of the simple system is given by $\mathbf{u} := \begin{bmatrix} \rho_{O_2} + \rho_O & \rho_{N_2} & \rho v & \rho E \end{bmatrix}^T$.

The governing equations and the closure employed, i.e. the equation of state and the chemical reactions, is as discussed in Section 4.1 and Section 4.2 respectively. The thermodynamic and physical constants of the chemical constituents at hand are listed in Table 5.1, cf. [16] and [63].

Table 5.1: Physical and thermodynamics constants

	ρ_{O_2}	ρ_O	ρ_{N_2}
$M\ (kg)$	0.032	0.016	0.028
$e_0\ (J\ K^{-1}\ mol^{-1})$	249200	0	0
$c_v\ (J\ mol^{-1}\ )$	$\frac{5}{2M_{O_2}}R$	$\frac{3}{2M_{O_2}}R$	$\frac{5}{2M_{O_2}}R$
ρ_R (kg)	1145	1141	1308
$s^R\ (J\ K^{-1}\ mol^{-1}\ )$	205.15	161.1	191.61

The mass of atomic oxygen and atomic nitrogen is 0.016 kg and 0.014 kg respectively. The value of the specific gas constant is $R = 8.314\ J\ K^{-1}\ mol^{-1}$ and the reference temperature and pressure are assumed to be $T_R := 2000K, p_R = 1.01325 \times 10^5 m^{-1}\ kg \cdot s^{-2}$ respectively.

The forward reaction rate constant is assumed to be of the form

$$k_f := C \cdot T^{-2} \cdot \exp\left(\frac{-E}{T}\right), \tag{5.2}$$

where the constants $C\ (m^3\ mol^{-1}\ s^{-1})$ and $E\ (K)$ need to be specified. Forward reaction rate constant of the form (5.2) is commonly employed in non-equilibrium hypersonic airflows, cf. [46].

5.1.2 Numerical setup

We employ a third order RK-DG scheme, employing quadratic polynomials and a third order SSP-RK scheme. The `Multiwave` library [70] along with the `GNU Scientific`

`Library (GSL)` [35] is employed for matrix, vector operations and to solve non-linear system of equations for computing the Maxwellian.

The computational domain is assumed to be $\Omega := [-1, 1]$ and periodic boundary conditions are employed. The maximum mesh refinement level is set to $L = 7$. Furthermore, the number of cells in the mesh on the smallest level, i.e. $l = 0$ in the grid hierarchy, is assumed to be 10. Hence, the non mesh adaptive computations are done on a uniform mesh with $\mathcal{N}_E := 10 \times 2^7 = 1280$ cells and the number of cells in mesh adaptive simulations can be between 10 and 1280. The size of the time step is fixed throughout the computations and is such that it satisfies the CFL condition (3.14) with $c \approx 0.1$.

5.2 Coupling

In this section, the numerical simulations conducted to test the coupling proposed in Section 3.3.6 are presented and their results are discussed.

The objective of model adaptation is to locally employ the simple system in order to reduce the requisite computational resources with only a small drop in the accuracy of the resulting simulation. The placement of coupling interfaces and the employed coupling should be in accordance with this objective. To this end, we conduct numerical simulations to study the following:

(i) The first aspect to be investigated is the behaviour of waves, i.e. shocks, contact discontinuities and rarefaction waves, as they impinge coupling interfaces and the effect on the accuracy of the numerical simulations.

(ii) The second aspect to be investigated is the difference in the behaviour observed at different reaction speeds. The faster the reaction speed is, the closer the numerical solution will be to the equilibrium manifold as the waves impinge an interface. This is expected to lead to different behaviour of the coupling.

The above two aspects will help us asses how well suited the proposed coupling method is to the objective of model adaptation and inform us about reasonable placements of the interfaces in the computational domain in the model adaptive simulations.

To test the proposed coupling, we consider shock tubes at different reaction speeds. The system of equations is initialized as follows.

Initial data

The velocity and temperature is set to $T = 2000, v = 0$ in the entire computational domain. Pressure and density of atomic oxygen is set to

$$p = 2 \cdot 10^6, \rho_0 = 0.01, \text{ for } x < 0 \tag{5.3}$$

and

$$p = 10^6, \rho_0 = 0.005, \text{ for } x \geq 0. \tag{5.4}$$

Based on the values of the temperature, the pressure, the velocity and the density of atomic oxygen, $\mathbf{U}_{eq} = \begin{bmatrix} \rho_{O_2} & \rho_O & \rho_{N_2} & \rho v & \rho E \end{bmatrix}^T$ is calculated. Here $\mathbf{U}_{eq}$ corresponds to the equilibrium value and is calculated by employing the equation of state and the fact that the reaction term in the balance laws of partial densities should vanish, see (4.21)-(4.25).

We study two cases; waves impinging coupling interfaces, where the waves have been generated in a sub-domain where the complex system is employed and waves impinging coupling interfaces, where the waves have been generated in a sub-domain where the simple system is employed. To this end, the model employed in the computational domain for some cell $\mathcal{V}_k$ is as follows:

- *Case I:*

$$\Theta_k := \begin{cases} \text{complex model} & \text{if } 0.05 \leq x_{k+1} \leq 0.08125 \text{ and } -0.1 \leq x_{k+1} \leq -0.128125 \\ \text{simple model} & \text{otherwise} \end{cases}. \tag{5.5}$$

- *Case II:*

$$\Theta_k := \begin{cases} \text{complex model} & \text{if } -0.1 \leq x_{k+1} \leq 0.05 \\ \text{simple model} & \text{otherwise} \end{cases}. \tag{5.6}$$

The system of equations are initialized by setting the initial values to $\mathbf{U}_{eq}$ for cells in Ω_c and $\mathbb{P}\mathbf{U}_{eq}$ for cells in Ω_s. The simulations are conducted until time $T = 3 \cdot 10^{-4}$.

To study the behaviour at different reaction speeds, the shock tube problem is considered at three reaction speeds with constants C and E set to

$$\text{Slow reaction: } C = 2.9 \cdot 10^9, E = 597.5, \tag{5.7}$$
$$\text{Intermediate reaction: } C = 2.9 \cdot 10^{10}, E = 597.5, \tag{5.8}$$
$$\text{Fast reaction: } C = 2.9 \cdot 10^{12}, E = 597.5. \tag{5.9}$$

The initial discontinuity leads to the formation of a shock and a contact discontinuity travelling to the right and a rarefaction wave expanding as it travels to the left.

Remark 5.2.1. Figures 5.1 and 5.10 show plots of the variables at time $t = 3.725 \cdot 10^{-5}$ for Case *I* and *II* respectively. In these figures and in rest of the plots, the sub-domain where the simple system is employed is shown with a green background and the sub-domain where the complex system is employed is shown with a white background. The Maxwellian is employed to calculate the full set of densities, momentum and energy of the fluid mixture for cells in Ω_s.

Notation and plots

The numerical solutions of the coupling simulations, i.e. simulations where the domain is decomposed as defined in (5.5) and (5.6), are denoted by $\mathbf{U}_h^c$. The reference numerical solutions, i.e. numerical solutions of simulations with complex system being employed in the entire computational domain and throughout the entire simulation, but with the same numerical setup including initialization, mesh and time step size, is denoted by $\mathbf{U}_h^r$. Naturally, the Maxwellian is employed to plot the values of $\mathbf{U}_h^c$ for the cells in Ω_s.

To study the results of the simulations, the distance to the equilibrium manifold is quantified by calculating and plotting the scaled integral of the relative entropy between the numerical solution and its projection on the equilibrium manifold, i.e. $\mathbf{U}_h^c$ and $M\left(\mathbb{P}\mathbf{U}_h^c\right)$. To this end, for cell $\mathcal{V}_j$ let

$$\kappa_\varepsilon^{\mathfrak{n},j} := \left(\frac{1}{\Delta x_j} \cdot \int_{\mathcal{V}_j} H\left(\mathbf{U}_{h,j}^c | M\left(\mathbb{P}\mathbf{U}_{h,j}^c\right)\right) \mathrm{dx}\right)^{0.5}. \tag{5.10}$$

Analogously, the error incurred in the coupling simulation for $\mathcal{V}_j$ is quantified by

$$\kappa_r^{\mathfrak{n},j} := \left(\frac{1}{\Delta x_j} \cdot \int_{\mathcal{V}_j} H\left(\mathbf{U}_{h,j}^r | \mathbf{U}_{h,j}^c\right) \mathrm{dx}\right)^{0.5}. \tag{5.11}$$

Furthermore, to investigate the behaviour of the waves as they hit the interface, primitive variables, i.e. the velocity, the temperature and the pressure of the fluid mixture is plotted.

In the next section, we describe some observations made and make some inferences based on the results of the coupling simulations.

5.2.1 Waves travelling from the complex to the simple system

Figure 5.1 shows the plot of the full set of the variables for the case of the slow reaction. In the subsequent sections, we study the behaviour of the contact discontinuity and the shock as it hits the interface at $x = 0.05$ and as the rarefaction hits the interface at $x = -0.1$.

Impingement of shock on a coupling interface

Figure 5.2 shows the plots of the density of molecular oxygen (ρ_{O_2}) and $\kappa_\varepsilon^{\mathfrak{n}}$ and $\kappa_r^{\mathfrak{n}}$ for all reaction speeds at $t = 6.2005 \cdot 10^{-5}$. Figure 5.3 and Figure 5.4 shows the plots of the primitive variables and the density of oxygen for the coupling simulation (ρ_{O_2}) and the reference simulation ($\rho_{O_2}^r$) for the slow and fast reaction at $t = 6.2005 \cdot 10^{-5}$ respectively.

We can observe the following:

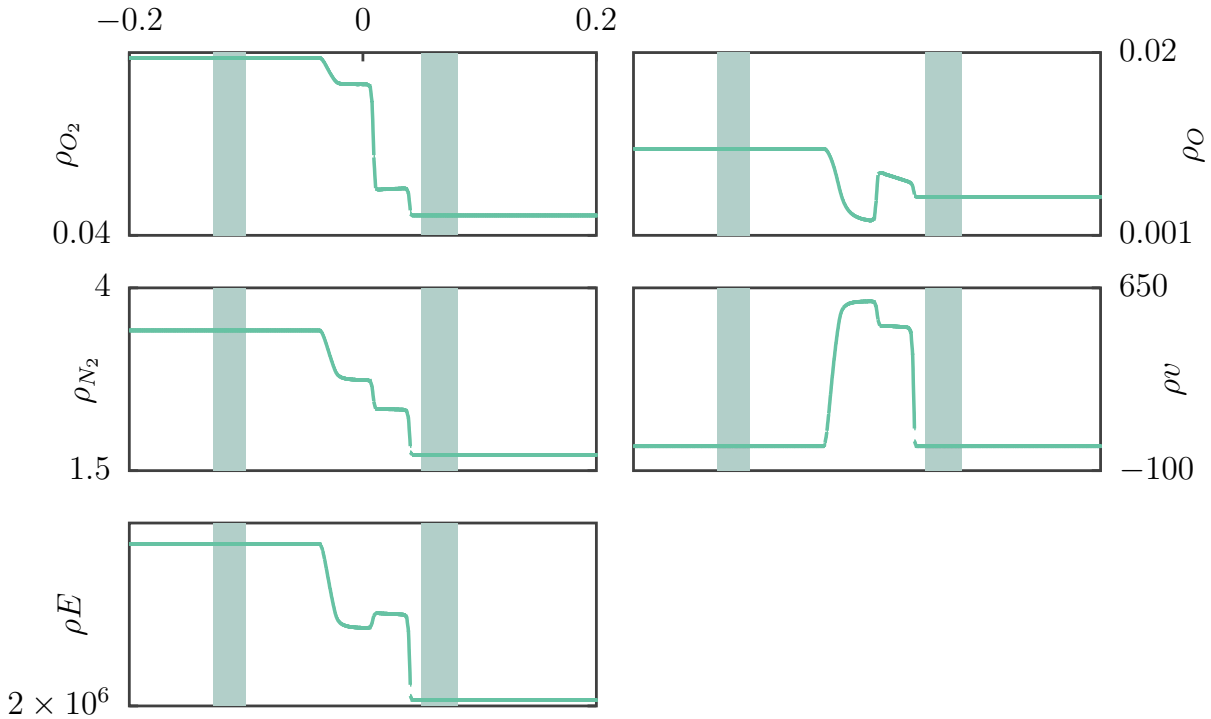

Figure 5.1: Shock tube: $t = 3.725 \cdot 10^{-5}$ for slow reaction

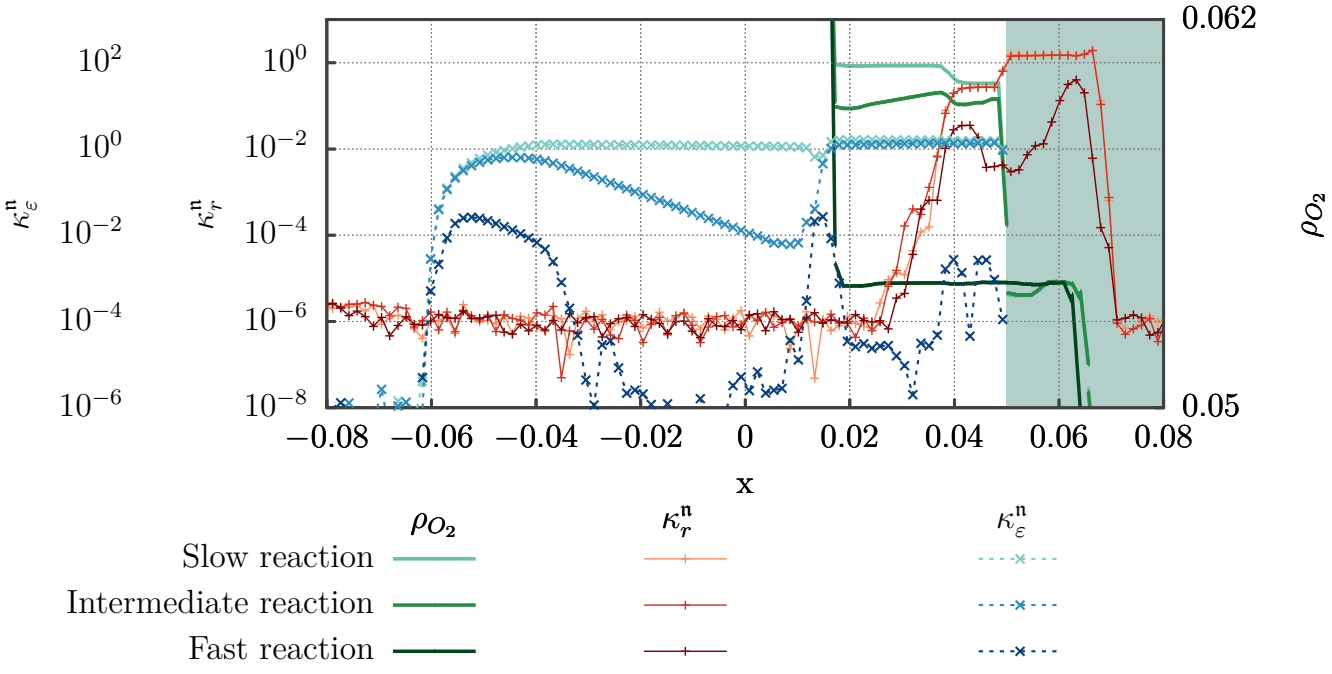

Figure 5.2: Plot of shock at $t = 6.2005 \cdot 10^{-5}$ for all reaction speeds

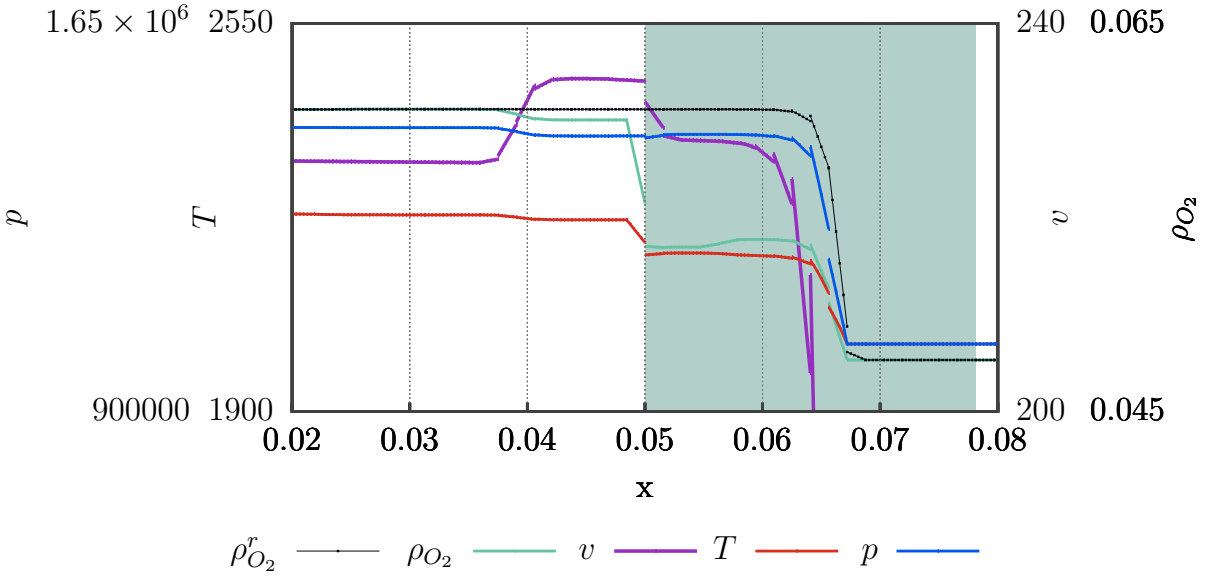

Figure 5.3: Plot of primitive variables at $t = 6.2005 \cdot 10^{-5}$ slow reaction

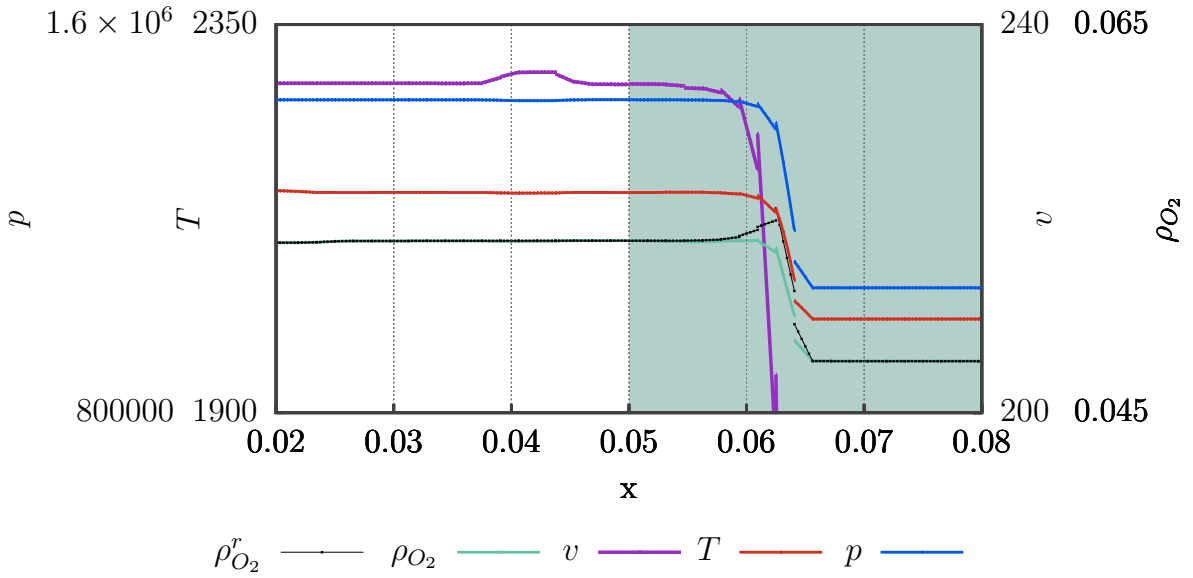

Figure 5.4: Plot of primitive variables at $t = 6.2005 \cdot 10^{-5}$ fast reaction

- In Figure 5.2, we can see that the shock splits into a shock travelling ($x \approx 0.06$) to the right a rarefaction travelling to the left ($x \approx 0.04$) and a jump discontinuity at the interface ($x = 0.05$) for slow and intermediate speed of reaction. The rarefaction and the shock wave is more apparent in Figure 5.3, which shows the the plots of the primitive variables for slow reaction speed.

- By comparing Figures 5.3 and 5.4, we can observe that the strength of the waves generated is smaller for the fast reaction than for the slow reaction. The rarefaction is distinguishable in the plot of velocity in Figure 5.4 for fast reaction speed, while not easily distinguishable in plots of other primitive variables. On the other hand, all of the waves are easily distinguishable in Figure 5.3 in the case of slow reaction speed.

- The coupling simulation ($\rho^c_{O_2}$) agrees with the reference solution ($\rho^r_{O_2}$) to a larger extent as the reaction speed increases, albeit the sharp peak at the shock is smoothened out in the coupling simulation of the fast reaction, as observed in Figures 5.3 and 5.4. The better agreement between the reference and the coupling simulation at faster reaction speeds is observed in the plots of $\kappa^{\mathfrak{n}}_r$ and $\kappa^{\mathfrak{n}}_\varepsilon$ in Figure 5.2, where the distance to the equilibrium manifold decreases at faster reaction speeds and so does the difference between the reference and the coupling simulation.

The behaviour observed above can be explained by noting the following:

- Let $\mathbf{U}_l$ and $\mathbf{U}_r$ be the left and right states that make up the shock, then for the case of slow speed of reaction, the numerical solution is far away from the equilibrium manifold, i.e. $|\mathbf{U}_l - M(\mathbb{P}\mathbf{U}_l)| \gg 1$ or $|\mathbf{U}_r - M(\mathbb{P}\mathbf{U}_r)| \gg 1$. On the other hand, for the case of the fast reaction, the numerical solution is close to the equilibrium manifold, i.e. $|\mathbf{U}_l - M(\mathbb{P}\mathbf{U}_l)| \ll 1$ and $|\mathbf{U}_r - M(\mathbb{P}\mathbf{U}_r)| \ll 1$.

- Furthermore, $M(\mathbb{P}\mathbf{U}_l)$ and $M(\mathbb{P}\mathbf{U}_r)$ cannot be connected by a shock. This can be reasoned by checking if the jump discontinuity produced by the states $M(\mathbb{P}\mathbf{U}_l)$ and $M(\mathbb{P}\mathbf{U}_r)$ satisfies the Rankine Huignot jump condition [62]. It was observed that the jump discontinuity does not satisfy the Rankine Huignot jump condition. In particular, the mismatch in the speed of the wave being in the one calculated using the values and the fluxes of ρ_{O_2} and ρ_O. This is expected since the values of the density of atomic and molecular oxygen, i.e. ρ_{O_2} and ρ_O, differ from their equilibrium counterpart, i.e the values calculated by projecting the vector of variables of the complex system using the projection matrix followed by computing the Maxwellian. On the other hand, the values of $\rho_{N_2}, \rho v$ and ρE are unchanged. Hence, shocks formed in Ω_c split up when they hit the interface. In particular, it was numerically verified that the discontinuities generated once the shock hits the interface satisfy the Rankine Huignot jump condition of the simple system.

Impingement of a contact discontinuity on a coupling interface

Figure 5.5 shows the plots of the density of oxygen (ρ_{O_2}), $\kappa_\varepsilon^{\mathfrak{n}}$ and $\kappa_r^{\mathfrak{n}}$ at $t = 1.25 \cdot 10^{-4}$ for all the reaction speeds. Figure 5.6 and Figure 5.7 show the plots of the primitive variables and the density of oxygen for the coupling simulation ($\rho_{O_2}^c$) and the reference simulation ($\rho_{O_2}^r$) for the slow and fast reaction at $t = 1.25 \cdot 10^{-4}$. At this point in time, the contact discontinuity has hit the interface and is at $x \approx 0.06$.

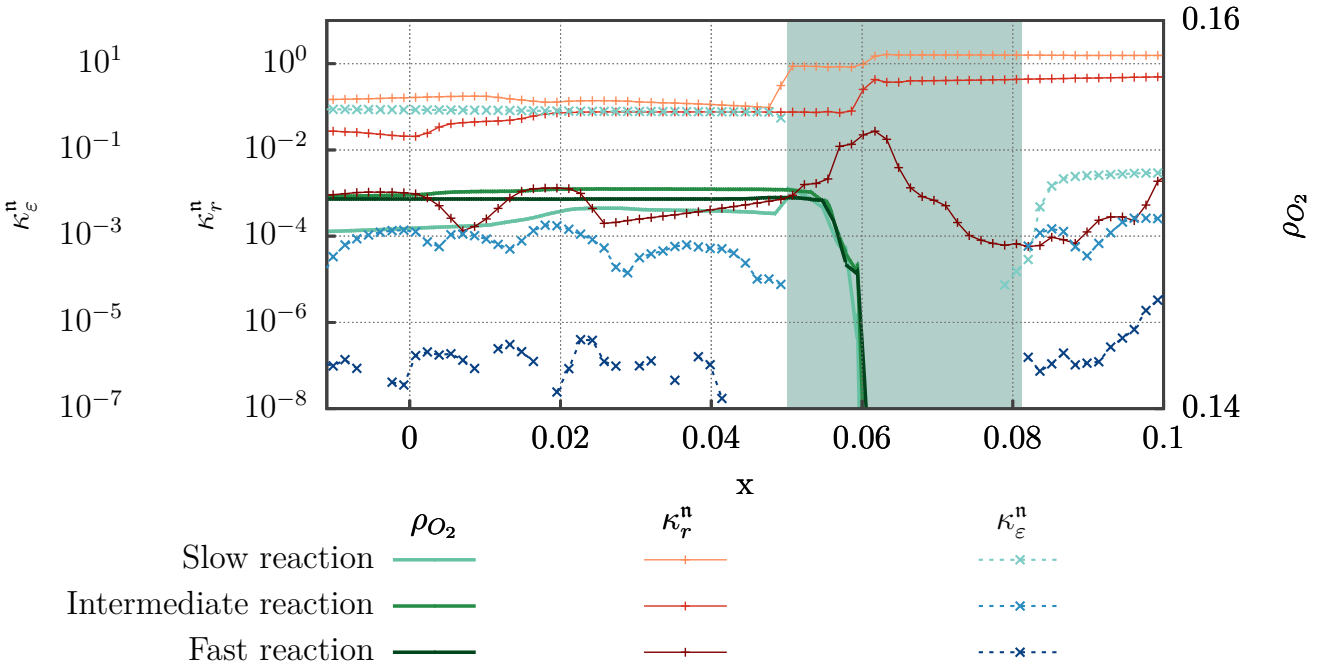

Figure 5.5: Shock tube: $t = 1.25 \cdot 10^{-4}$

The following can be observed when the contact discontinuity hits the coupling interface:

- The contact discontinuity splits into a rarefaction wave travelling to the left and a contact discontinuity travelling to the right at $x \approx 0.02$ and $x \approx 0.06$ respectively and a small jump discontinuity at the interface. The waves are more apparent for the case of slow reaction in the plot of the primitive variables in Figure 5.6.

- No easily distinguishable splitting of the contact discontinuity is observed in the simulation of the fast reaction and the coupling simulation closely agrees with the reference simulation.

Impingement of a rarefaction on a coupling interface

Figure 5.8 shows the plots of the density of oxygen (ρ_{O_2}), $\kappa_\varepsilon^{\mathfrak{n}}$ and $\kappa_r^{\mathfrak{n}}$ at $t = 2.67 \cdot 10^{-4}$ for all reaction speeds. At this point in time, the rarefaction wave has crossed the interface at $x = -0.1$. Once the rarefaction crosses the interface, it leads to a jump across the interface

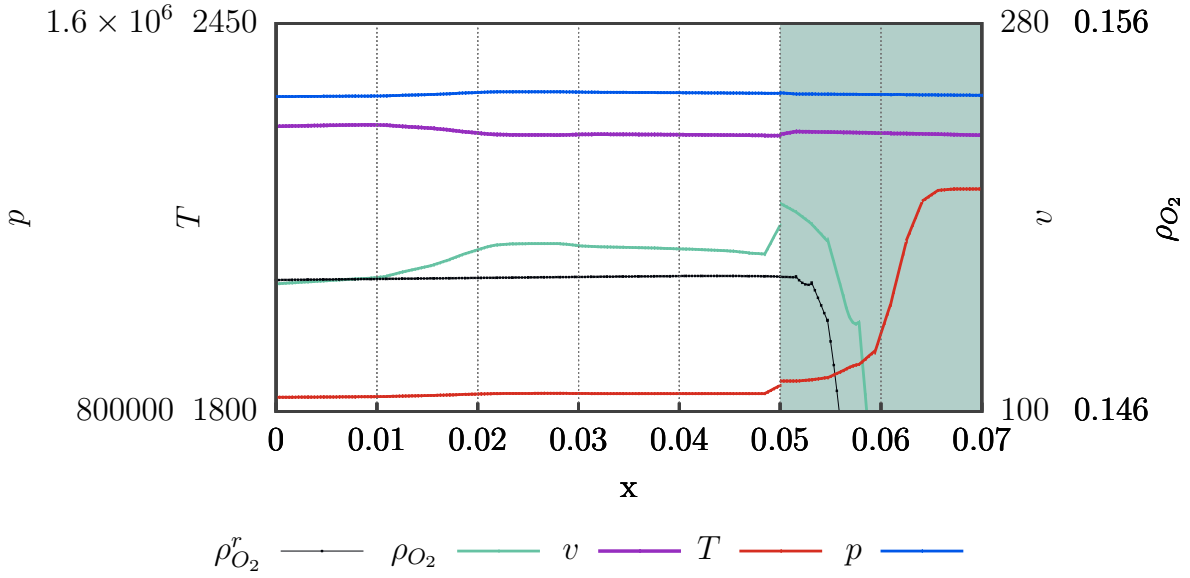

Figure 5.6: Plot of primitive variables at $t = 1.25 \cdot 10^{-4}$ slow reaction

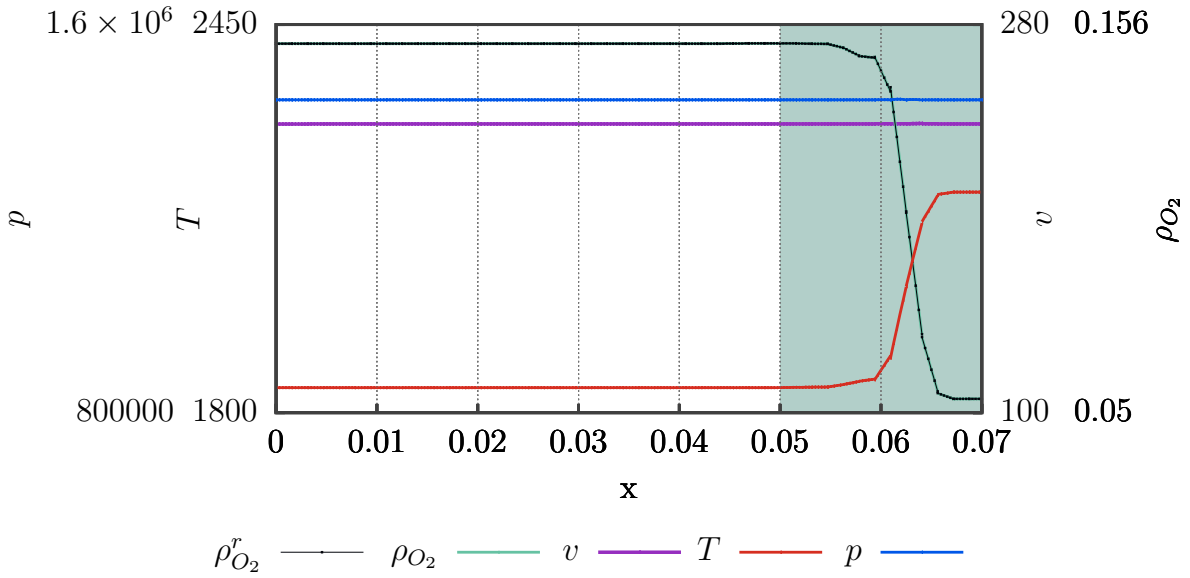

Figure 5.7: Plot of primitive variables at $t = 1.25 \cdot 10^{-4}$ fast reaction

since the states are constrained to the equilibrium manifold for cells in Ω_s. Furthermore, we can observe that the slower the reaction speed is, the larger the jump produced across the interface is. This is due the fact that the slower the reaction speed is, the further away the numerical solution is from the equilibrium manifold.

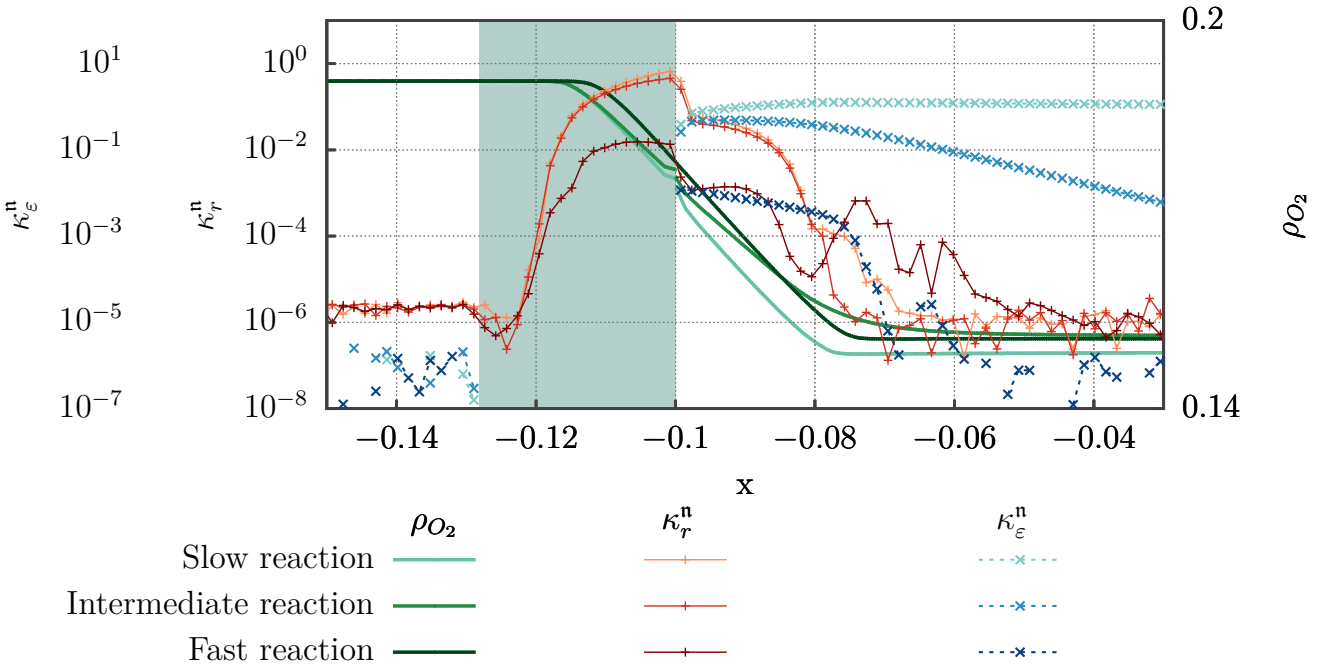

Figure 5.8: Shock tube: $t = 2.67005 \cdot 10^{-4}$

Comparison with reference simulations

Figure 5.9 shows the plots of the square root of the integral of relative entropy between the coupling simulation and the reference simulation. We can observe that the slower the reaction speed is, the larger relative entropy is between the reference and the coupling simulation. This is correlates to the observation that the slower the speed of the reaction is the further away from the equilibrium manifold the numerical solution is. Consequently, this leads to creation of larger artefacts. Furthermore, for the fast reaction, the artefacts created are quickly dampened due to the fast speed of the reaction, leading to a small reduction in the relative entropy at $t \approx 7.5 \times 10^{-5}$.

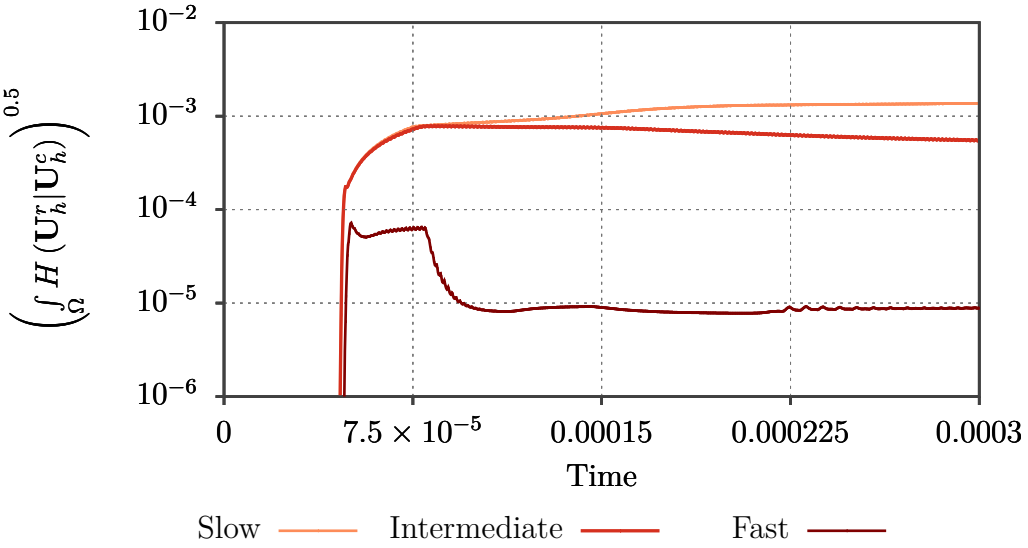

Figure 5.9: Relative entropy for all reaction speeds

5.2.2 Waves travelling from the simple to the complex system

Next, we study the behaviour of waves created in Ω_s as they hit coupling interfaces. To this end, we consider Case *II* at reaction speeds defined in Section 5.2.

Figure 5.10 shows the plots of the variables at time $t = 3.725 \times 10^{-5}$. In the subsequent sections, we study the behaviour as the waves hit the interface at $x = -0.1$ and at $x = 0.05$.

Impingement of a shock on a coupling interface

Figure 5.11 shows the plot of ρ_{O_2}, $\kappa_\varepsilon^{\mathrm{n}}$ and κ_r^{n} for all reaction speeds. Figures 5.12 and 5.13 show the plots of the primitive variables and the density of O_2 at time $t = 6.2005 \times 10^{-5}$ for slow and fast reaction speed respectively. At this point in time, the shock has hit the interface at $x = 0.05$ and crossed into the sub-domain where the complex system is employed.

We observe and note the following:

- The shock splits into an easily distinguishable shock at $x \approx 0.065$, a contact discontinuity at $x \approx 0.05$ and a rarefaction wave at $x \approx 0.04$ for slow and intermediate reaction speed. The waves can be observed in Figure 5.12 showing plots of the primitive variables for slow reaction. The waves are not easily distinguishable for the case of fast speed of reaction, see Figure 5.13.

- It is expected that a shock generated in Ω_s splits once it impinges on an interface

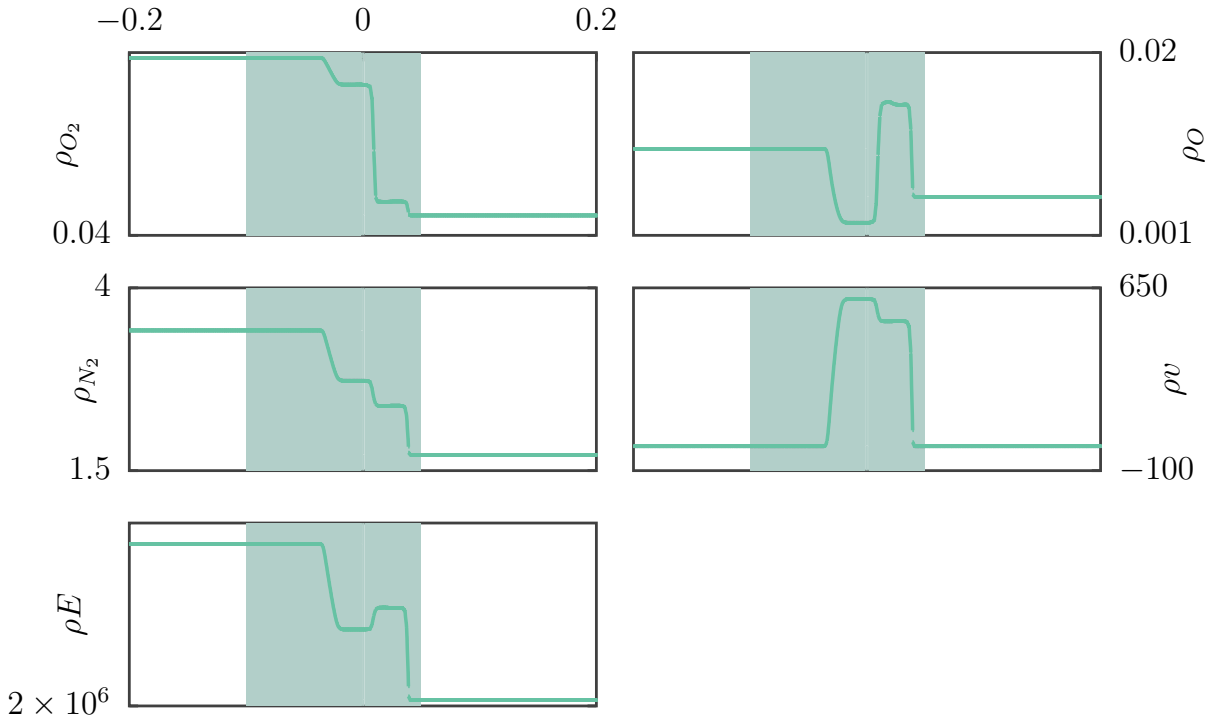

Figure 5.10: Shock tube: $t = 3.725 \cdot 10^{-5}$

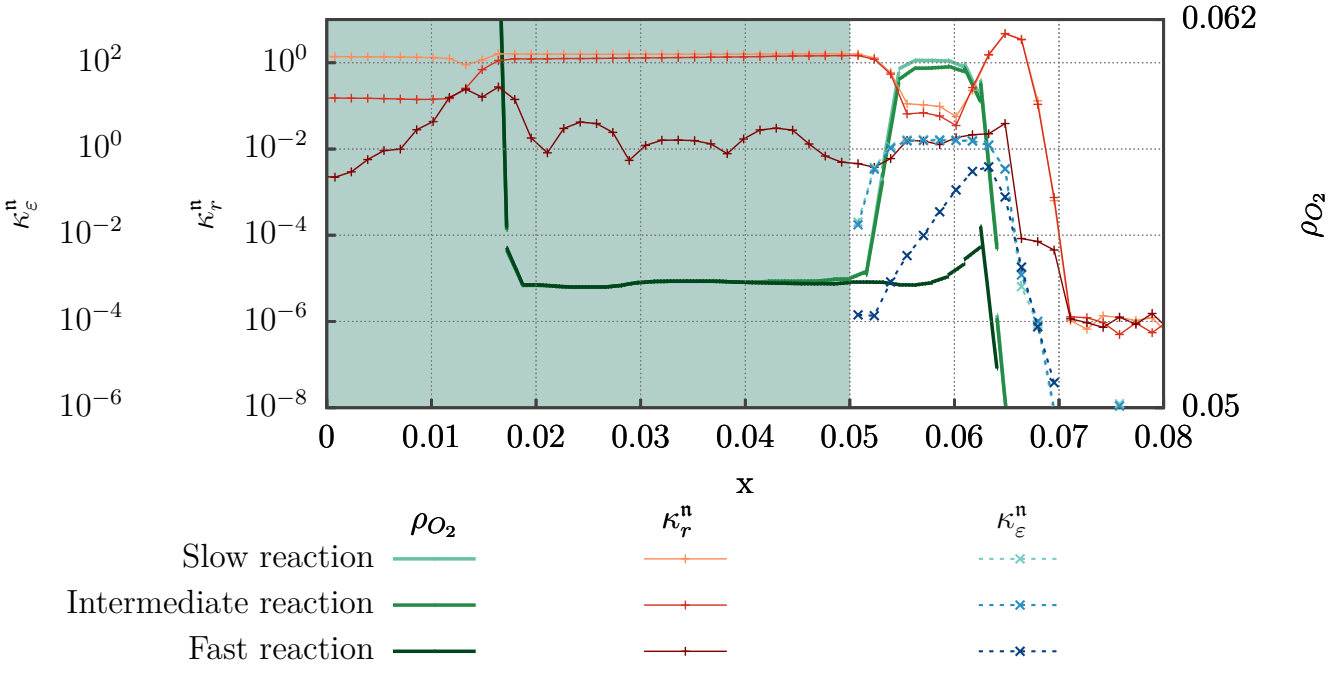

Figure 5.11: Plot of shock at $t = 6.2005 \cdot 10^{-5}$ for all reaction speeds

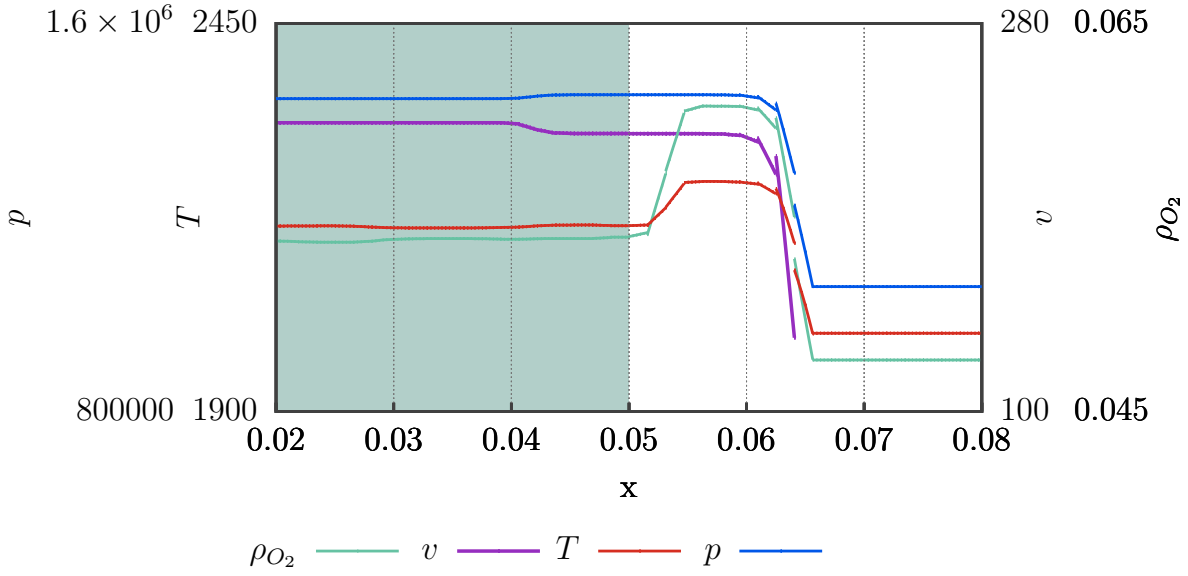

Figure 5.12: Plot of primitive variables at $t = 6.2005 \cdot 10^{-5}$ slow reaction

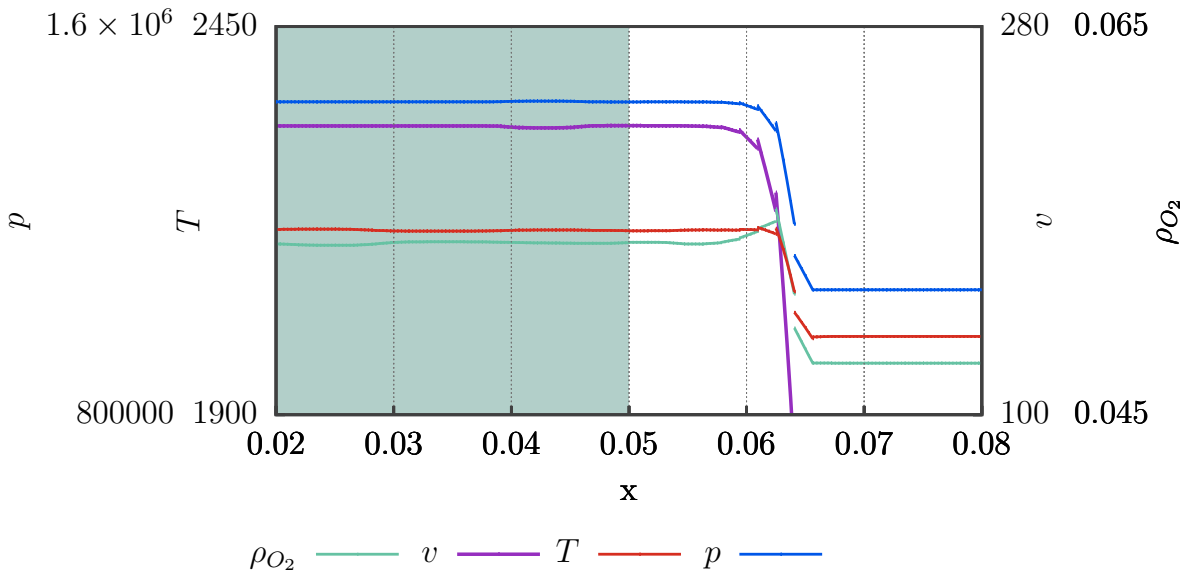

Figure 5.13: Plot of primitive variables at $t = 6.2005 \cdot 10^{-5}$ fast reaction

since a shock in Ω_s is not necessarily a shock in Ω_c. This can be explained by the Rankine Huignot condition. Let $\mathbf{u}_l$ and $\mathbf{u}_r$ be the left and the right states that make up the shock in Ω_s, then it was numerically verified that the Rankine Huignot jump condition is satisfied with the states and fluxes of the simple system, i.e. $\mathbf{u}_l, \mathbf{u}_r$ and $\mathbf{g}(\mathbf{u}_l), \mathbf{g}(\mathbf{u}_r)$ respectively. On the other hand, the states and the flux when converted to the complex system, i.e. $M(\mathbf{u}_l), M(\mathbf{u}_r)$ and $\mathbf{F}(M(\mathbf{u}_l)), \mathbf{F}(M(\mathbf{u}_r))$ violate the Rankine Huignot jump condition, indicating that these states cannot be connected by a shock in Ω_c leading to a generation of new waves.

Impingement of a contact discontinuity on a coupling interface

Figure 5.14 shows the plot of ρ_{O_2}, $\kappa_\varepsilon^{\mathfrak{n}}$ and $\kappa_r^{\mathfrak{n}}$ at time $t = 1.25 \cdot 10^{-4}$ for all reaction speeds. Furthermore, Figure 5.15 shows the plots of the primitive variables and the density of O_2. At this point in time the contact discontinuity has hit the interface at $x = 0.05$ and crossed into the sub-domain where the complex system is employed.

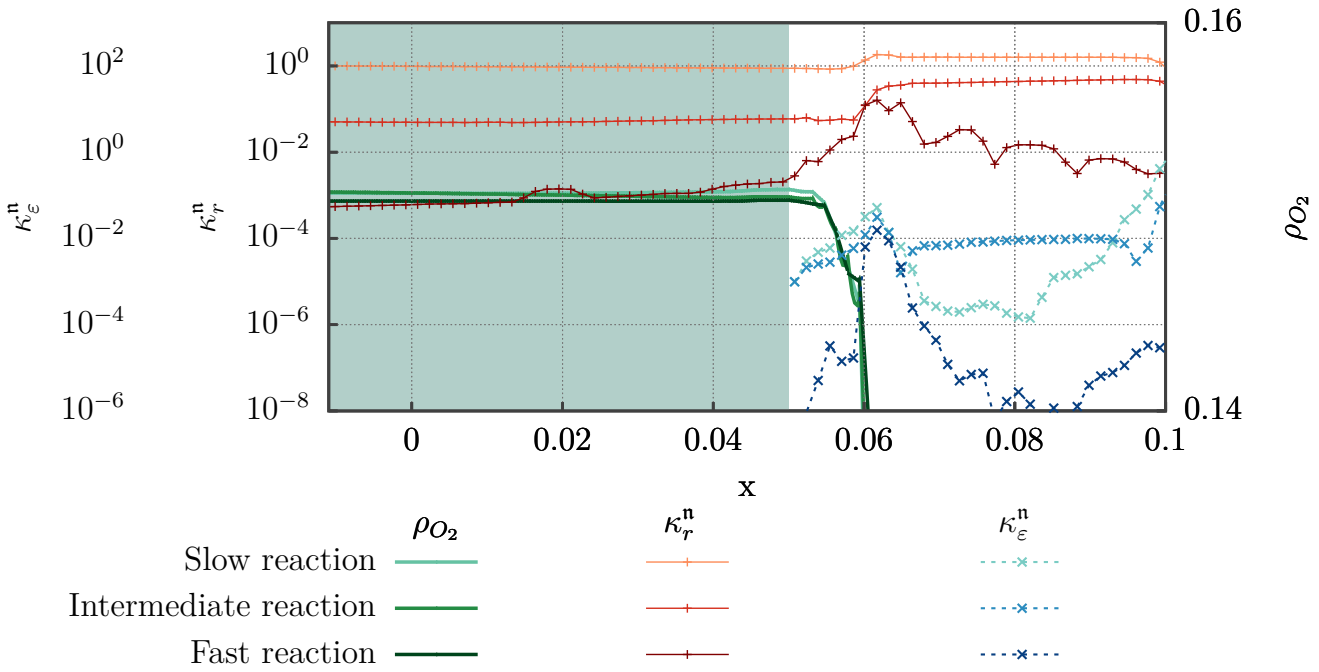

Figure 5.14: Shock tube: $t = 1.25 \cdot 10^{-4}$

We can observe and note the following:

- The contact discontinuity does not split into new waves for any of the reaction speeds. New waves are not created due to the fact that the existing contact discontinuity is also a valid contact discontinuity for the complex system, i.e. the jump discontinuity satisfies the Rankine Huignot jump condition for the complex system.

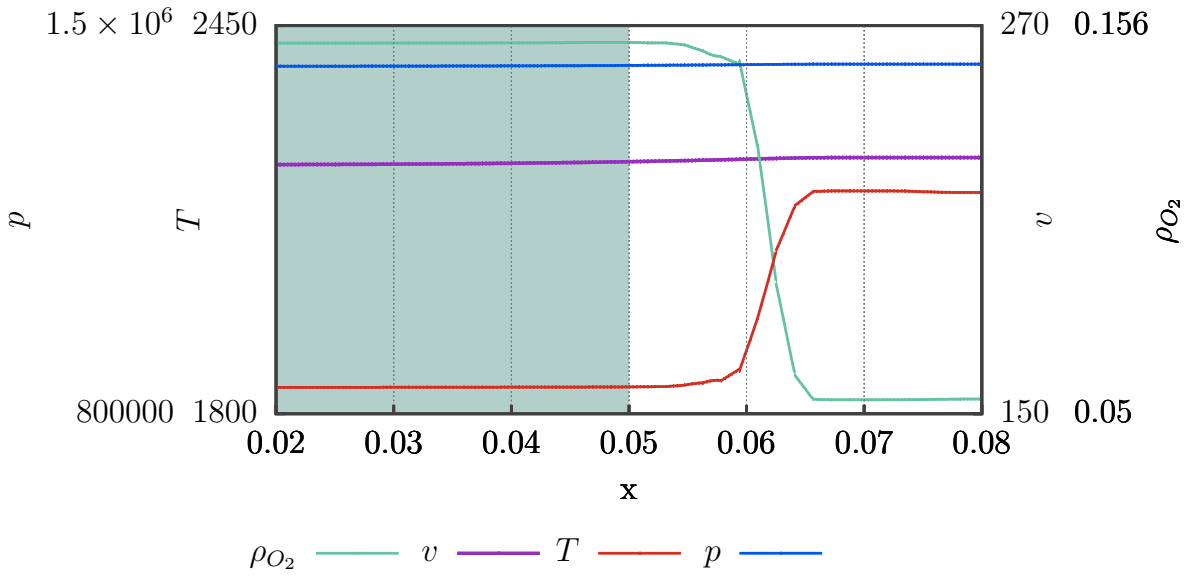

Figure 5.15: Plot of primitive variables at $t = 1.25 \cdot 10^{-4}$ slow reaction

- The difference in behaviour of a shock and a contact discontinuity is due to the fact that pressure and velocity is continuous across the jump for a contact discontinuity and discontinuous for a shock. Consequently, a contact discontinuity travels with the speed of the fluid mixture. Hence a discontinuity which is a contact discontinuity for the simple system is always a valid contact discontinuity for the complex system, while this is not true for a shock.

Impingement of a rarefaction on a coupling interface

Figure 5.16 shows the plots at $t = 2.67 \cdot 10^{-4}$ of the density of molecular oxygen, $\kappa_\varepsilon^{\mathrm{n}}$ and κ_r^{n} for all reaction speeds. At this point in time the rarefaction wave has crossed the interface at $x = -0.1$. For the slow and the intermediate reaction, the rarefaction keeps on expanding as it travels to the left, it leads to reactions since the states are no longer constrained to the equilibrium manifold in Ω_c.

5.2.3 Conclusions

We can conclude from the numerical experiments conducted that the coupling works reasonably well, as long as the states that make up the waves generated in Ω_c and impinging on a coupling interface are made up of states that are close to the equilibrium manifold. If this is not the case, then the coupling leads to artefacts. Furthermore, impingement of shocks travelling from Ω_s on coupling interfaces should be avoided since it may lead to

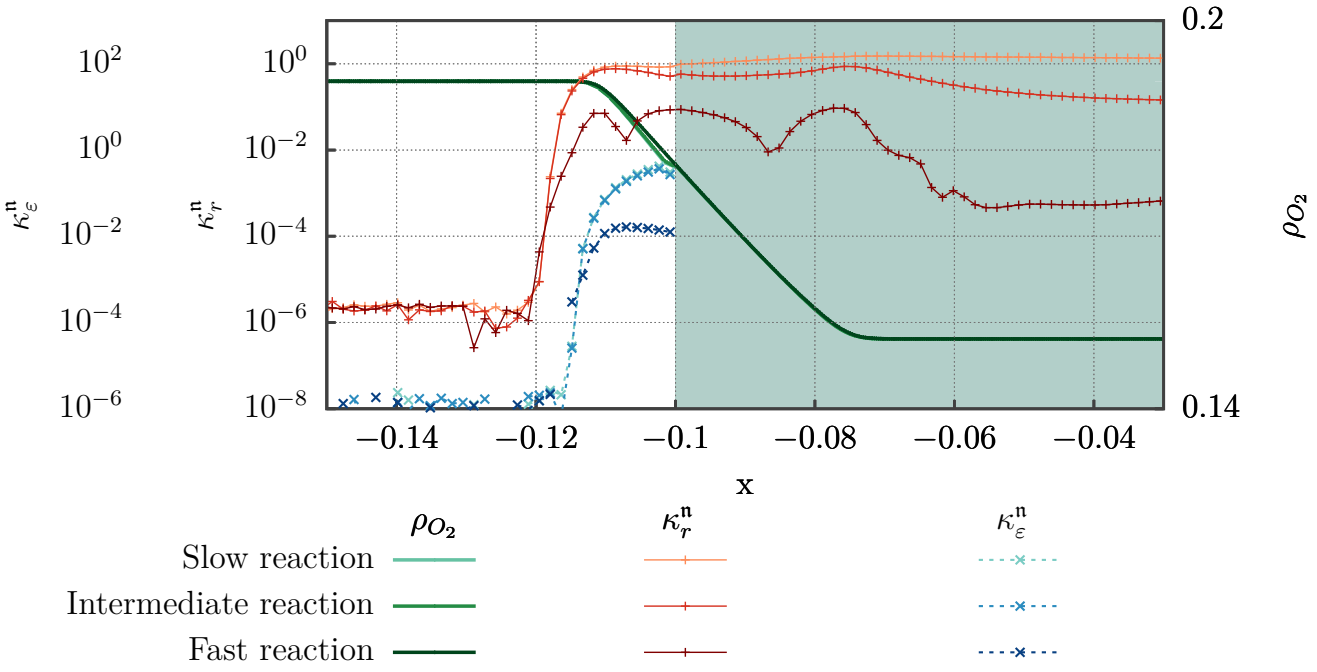

Figure 5.16: Shock tube: $t = 2.67 \cdot 10^{-4}$

artefacts, in particular for slow reaction speeds.

Next, we give a heuristic argument as to why the proposed model adaptation strategy should in most cases lead to suitable placements of coupling interfaces and should avert production of artefacts. Recall that the model adaptive simulations are carried out by starting from the complex system and given suitable conditions hold switching to the simple system. Qualitatively, the model adaptation strategy ensures that the numerical solution is close to the equilibrium manifold and that the dynamics is close to the equilibrium dynamics in regions where we switch to the simple system, before we switch to the simple system. We employ the simple system, if and only if both these criteria hold. Moreover, when decomposing the computational domain, safety cells are appended to the left and right of a patch of cells where the complex system is employed. In addition, even if the two states that make up a jump discontinuity lie on the equilibrium manifold, the numerical solution will be spread over a few cells. The smeared shock will in most cases be constituted of states away from the equilibrium manifold. As a result, the model coarsening distance in most cases will be larger than the tolerance τ_κ. This should ensure that the complex system is employed in regions near jump discontinuities, averting impingement of shocks and contact discontinuities in Ω_c with coupling interfaces.

5.3 Model adaptation

In this section, we discuss the results of model adaptive simulations. We consider two cases: shock tube and a travelling reaction front. The tests are conducted with a uniform mesh with $L = 7, \mathcal{N}_E = 1280$. In each test case, we first make some observations and then draw conclusions based on the observations.

5.3.1 Shock tube

Initial data

The temperature and velocity is set to $T = 2000, v = 0$ in the entire computational domain. The pressure of the fluid mixture and the density of atomic oxygen is set to

$$p = 2 \cdot 10^6, \rho_0 = 0.01, \text{ for } |x| \leq 0.5 \tag{5.12}$$

and

$$p = 10^6, \rho_0 = 0.005, \text{ for } |x| \geq 0.5. \tag{5.13}$$

Based on the values of the temperature, the pressure and the velocity, the equilibrium value $\mathbf{U}_{eq}$ is calculated. Note that in the first time step, the complex system is employed in the entire computational domain. After which, the model to be employed and the decomposition of the domain is done at the end of each time step, employing the model adaptation strategy described in Section 3.3. The constants determining the speed of the reactions are set to $C = 2.9 \cdot 10^{13}, E = 597.5$. Fast reaction speed is employed as this allows the simple system to be employed in large regions of the computational domain.

The initial discontinuity gives rise to a rarefaction wave, a shock and a contact discontinuity. The model adaptive simulations are conduced at four different tolerances, the simulation with the smallest tolerance being

$$\tau_r^s = 0.04, \tau_\kappa^s = 0.0004. \tag{5.14}$$

Subsequent simulations are carried out by scaling the tolerances by a factor of 2. The factor of safety is set to $\mathfrak{f}_\varepsilon = 0.25$. The conversion of the numerical solution from one system to the other is done employing the algorithms described in Section 3.3.4 and the simulations are carried out until time $T = 6 \cdot 10^{-5}$.

Notation

The numerical solution of the model adaptive simulation is denoted by $\mathbf{U}_h^\varepsilon$ and the reference numerical solution by $\mathbf{U}_h^r$.

The error incurred in the model adaptive simulations for some cell $\mathcal{V}_j$ is quantified by

$$\kappa_r^{\mathfrak{n},j} := \left(\frac{1}{\Delta x_j} \cdot \int_{\mathcal{V}_j} H\left(\mathbf{U}_{h,j}^r | \mathbf{U}_{h,j}^\varepsilon\right) \mathrm{dx} \right)^{0.5}. \tag{5.15}$$

Furthermore, we plot the model coarsening distance $\kappa_\varepsilon^{\mathtt{n}}$ and the modelling error indicators $\mathtt{M}_s^{\mathtt{n}}, \mathtt{M}_c^{\mathtt{n}}$ as defined in (3.59), (3.58) and (3.63) respectively. Recall that these quantities are employed when making a decision about the model to be employed. Note that the modelling error indicators are denoted by $\mathtt{M}_{s/c}^{\mathtt{n}}$ in the plots, where $\mathtt{M}_c^{\mathtt{n}}$ and $\mathtt{M}_s^{\mathtt{n}}$ is plotted in the sub-domains where the complex system and the simple system is employed respectively. Furthermore, the absolute value of the first component of the source term in the reference simulation, i.e. $\frac{1}{\varepsilon}|R_1\left(\mathbf{U}_h^r\right)|$ is plotted.

The tolerances $\tau_r, \mathfrak{f}_\varepsilon \cdot \tau_r$ and τ_κ employed in the model adaptive simulations are shown in the plots with horizontal lines as can be seen in Figure 5.17. In particular, $\tau_r, \mathfrak{f}_\varepsilon \cdot \tau_r$ are shown with yellow horizontal lines and τ_κ with a blue horizontal line.

Note that when comparing model adaptive simulations, for the sake of conciseness, the model adaptive simulations at different tolerances are denoted by τ at tolerance specified in (5.18), 2τ for the model adaptive simulation where the tolerances in (5.18) are scaled by a factor of 2 and so on.

Observations

We first discuss the results of the model adaptive simulation with tolerance $4\tau, \tau$ and then compare the behaviour of the model adaptation at different tolerances. We note the following:

- Initially, after the first time step, the model adaptation strategy leads to the simple system being employed everywhere, except near the waves created due to the initial discontinuity, see Figure 5.17.

- As the contact discontinuity and the shock travels to the right and the rarefaction travels to the left, the numerical solution approaches equilibrium in the two plateaus between the three waves. As a result, the simple system is employed between the contact discontinuity and the shock from time $2.6125 \cdot 10^{-4}$ onward, as seen in Figure 5.18, and between the rarefaction and the contact discontinuity from time $t = 3.725 \cdot 10^{-4}$ onward. The switch to the simple system is made once the modeling error indicator and the model coarsening distance are smaller than the prescribed tolerances and the number of cells in the patch where the simple system is to be employed is more than one cell.

- The complex system has to be employed near the contact discontinuity, the shock and the rarefaction. In the case of the rarefaction, this is due to the fact that the rarefaction continues to expand as it travels to the left. Hence, the states that make up the rarefaction are away from the equilibrium manifold. In the case of the contact discontinuity and the shock, since the jump discontinuity is smeared over a few cells, the states that make up the discontinuity are away from the equilibrium manifold.

- The size of the two patches of cells, where the simple system is employed, increases

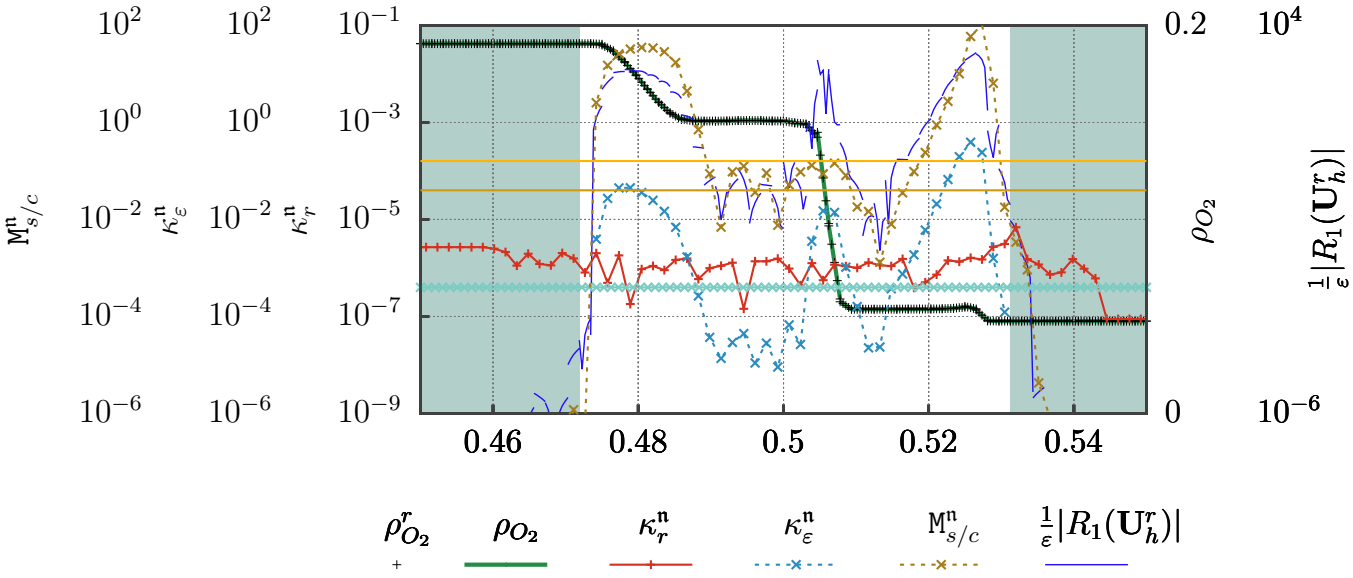

Figure 5.17: Shock tube: $t = 2.6 \cdot 10^{-4}$

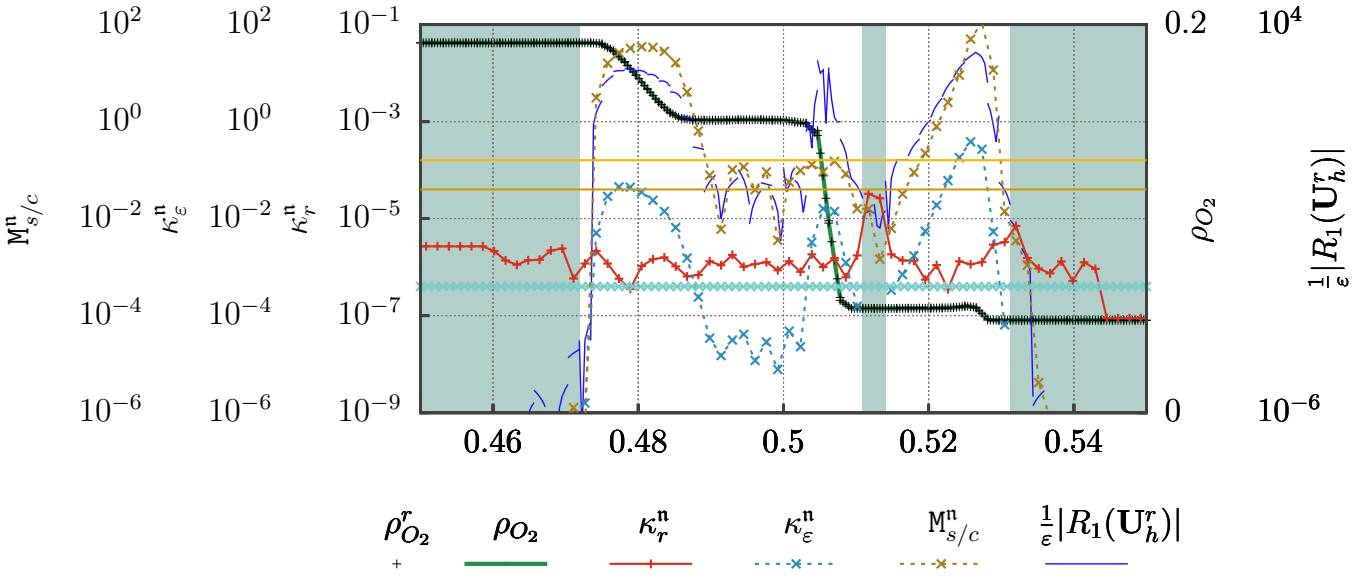

Figure 5.18: Shock tube: $t = 2.6125 \cdot 10^{-4}$

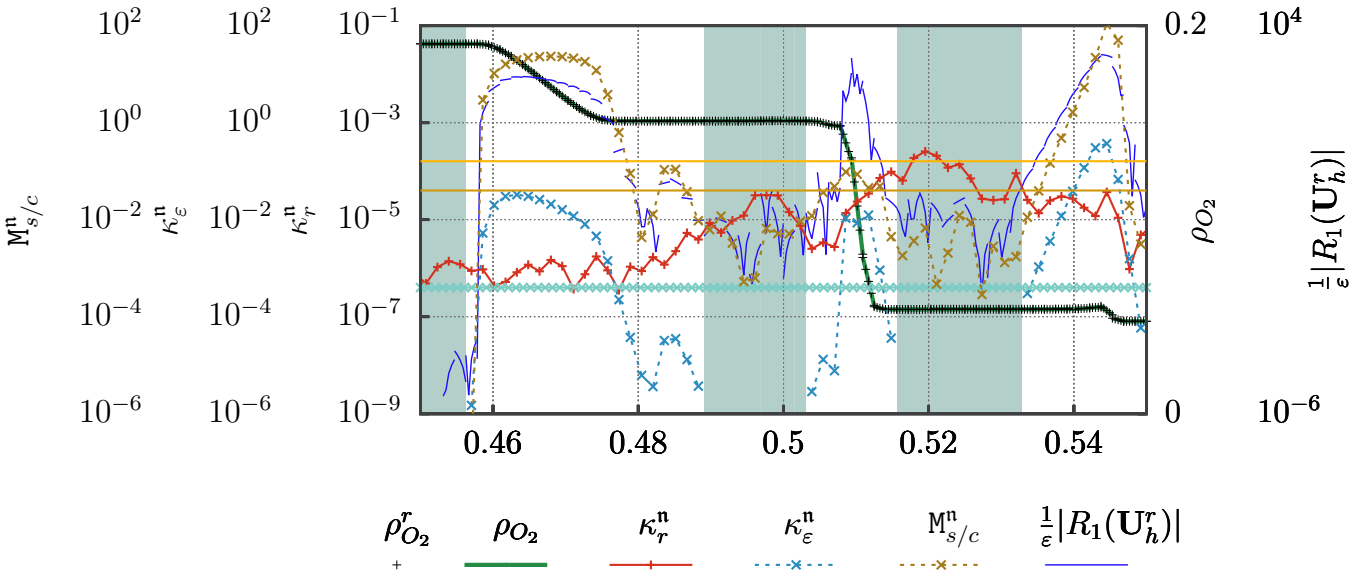

Figure 5.19: Shock tube: $t = 4.375 \cdot 10^{-4}$

in the long run. This is due to the fact that the numerical solution approaches equilibrium in a larger number of cells, see Figure 5.19.

- We can observe, in Figures 5.17-5.19, that the model adaptive solution qualitatively agrees to a large extent with the reference simulation and no artefacts are produced in the model adaptive simulations. We can also observe that the order of the source term in the reference simulation is relatively small in the regions where the simple system is employed. A cursory glance at the results indicates that the model adaptation strategy works reasonably well and passes sanity checks.

- Figures 5.20-5.22 show the plots of the model adaptive simulations for tolerance τ at times $3.2 \cdot 10^{-5}$, $3.225 \cdot 10^{-5}$ and $3.2375 \cdot 10^{-4}$ respectively. In this case, we switch to employing the simple system in between the contact discontinuity and the shock at $t = 3.2 \cdot 10^{-4}$, see Figure 5.20, and after three time steps we switch back to the complex system at time $t = 3.2375 \cdot 10^{-4}$, see Figure 5.22. This is a consequence of the model adaptation strategy giving rise to a safety cell. We can observe in Figure 5.21 that the model coarsening distance increases to a value above τ_κ in the first cell to the right of the patch at $x \approx 0.52$. As a result, with the added safety cell, the patch where the simple system can be potentially employed is just a single cell, which, according to our model adaptation strategy is not permitted. This results in a switching back to the complex system as seen in Figure 5.22.

- We can also observe, see Figure 5.23, that the size of the patch of cells where the simple system is employed does not significantly differ when the tolerances

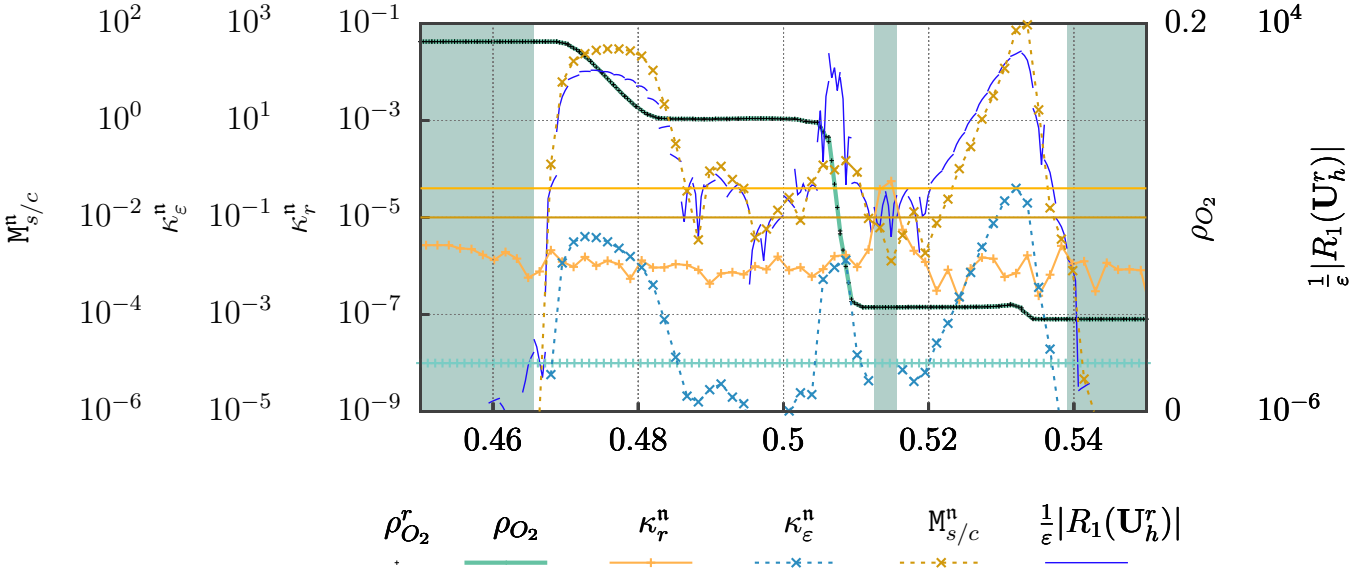

Figure 5.20: Shock tube: $t = 3.2 \cdot 10^{-4}$

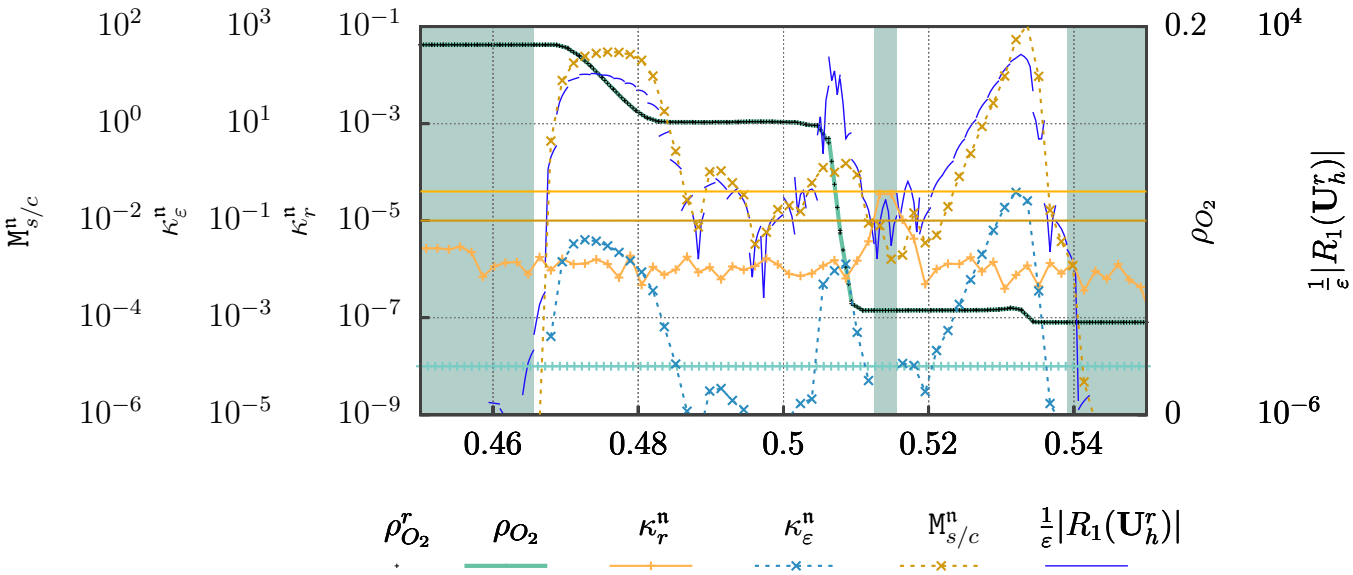

Figure 5.21: Shock tube: $t = 3.225 \cdot 10^{-4}$

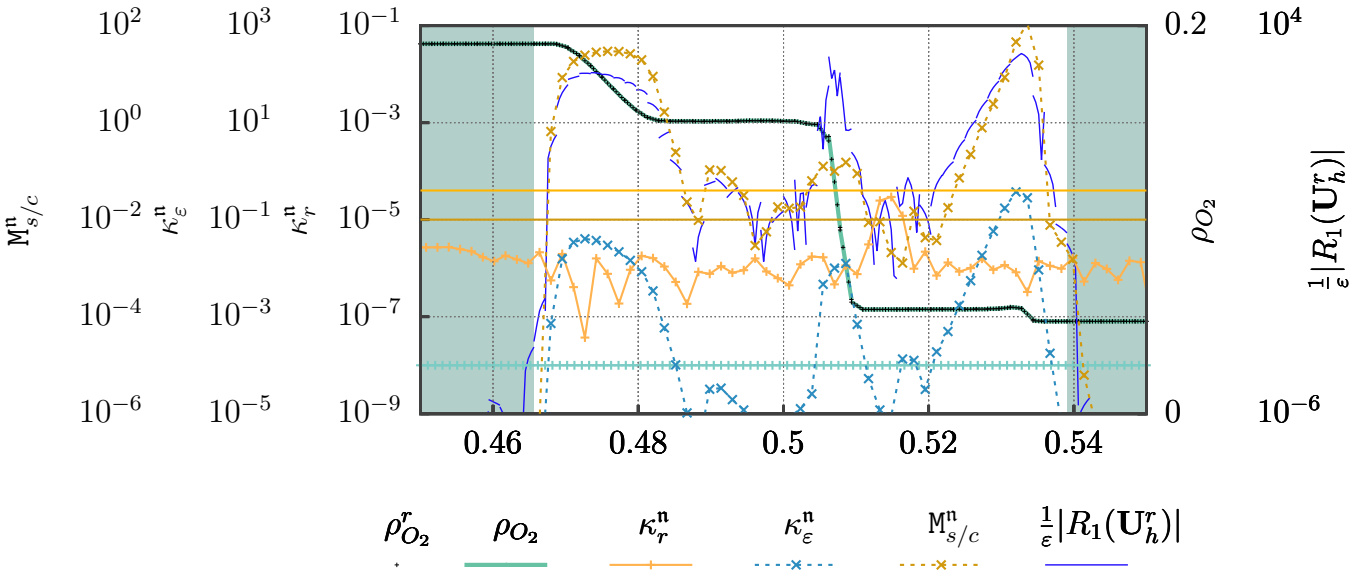

Figure 5.22: Shock tube: $t = 3.2375 \cdot 10^{-4}$

employed are increased. This is due to the fact that the indicators employed steeply increase near discontinuities. Hence, the distance between the interface and the discontinuities does not linearly scale with the tolerances. The differences between the model adaptive simulations at different tolerances are mainly observed in the time at which we begin to employ the simple system near the plateaus. Differences are also observed in the times at which the patches shift as the waves move, with the patches shifting earlier at lower tolerances, albeit within a margin of a few time steps.

- Figure 5.24 shows the plot of the square root of the integral of the relative entropy between the reference simulation and the model adaptive simulations at different tolerances. We can observe that the first peaks in the plots linearly scale with the tolerances. On the other hand, the long time behaviour does not. This can be attributed to the observation that, over a long time, the size of the sub-domains where the simple system is employed, does not significantly differ for different tolerances, but the time at which we first switch to the simple system does.

- We can also observe a cyclic behaviour in the relative entropy. The relative entropy increases and gives rise to crests and then decreases over a few time steps, giving rise to troughs. Consider the patch of cells where the simple system is employed between the contact discontinuity and the shock. The peaks can be attributed to the fact that the contact discontinuity approaches the interface leading to a small rise in the error. The patches shift after a few time steps as the contact discontinuity advances towards the interface. The decrease in the relative entropy can be observed

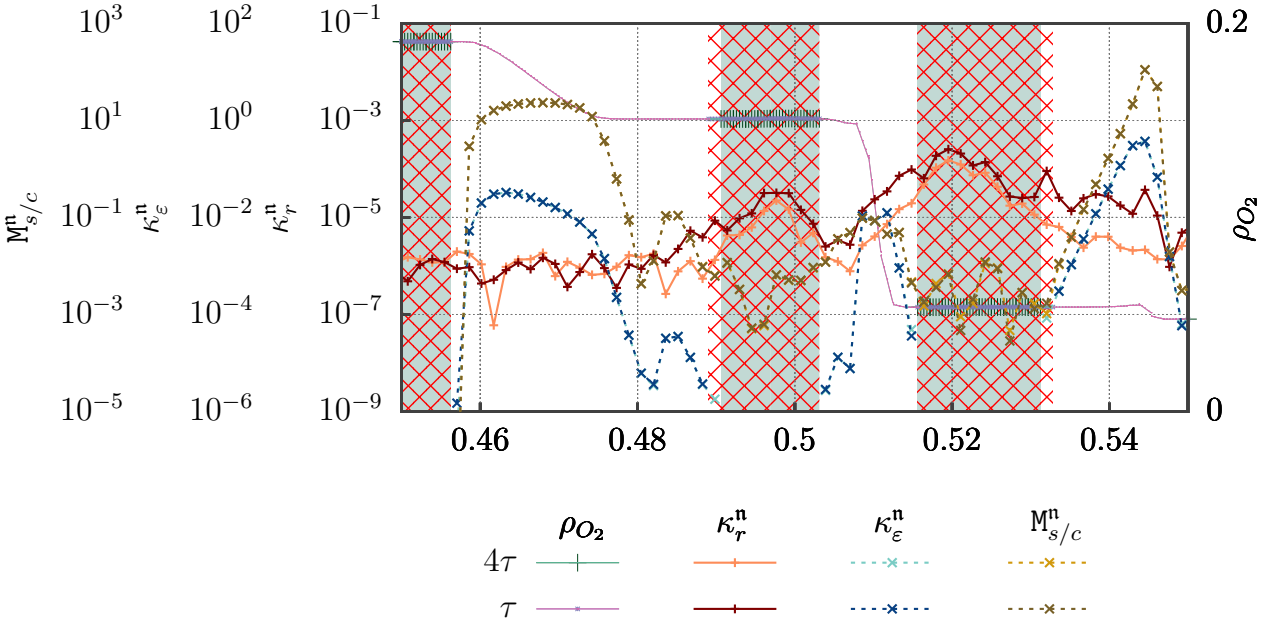

Figure 5.23: Shock tube: $t = 4.375 \cdot 10^{-4}$

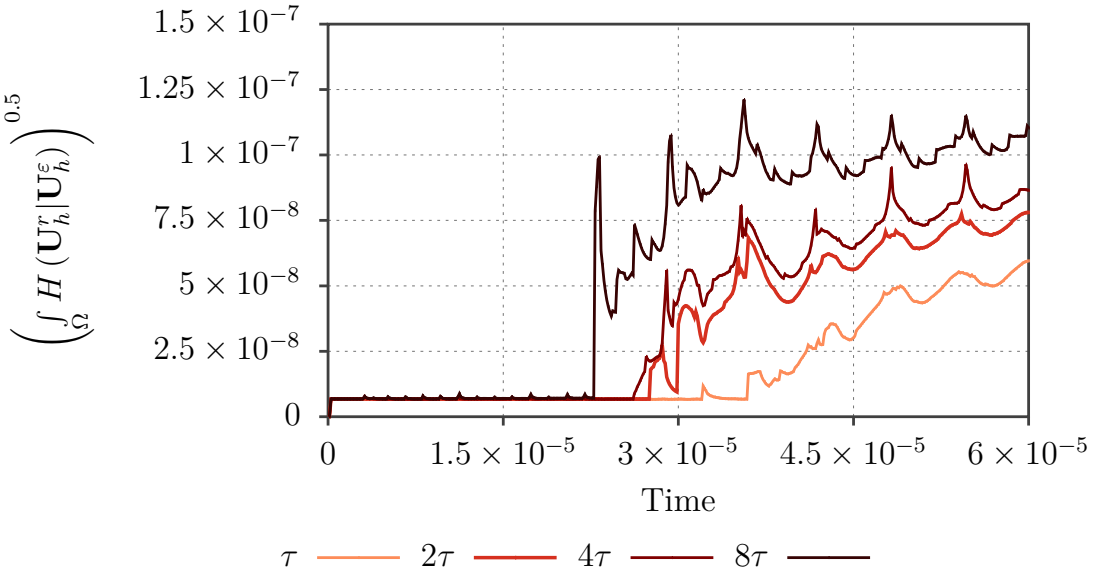

Figure 5.24: Relative entropy: Shock tube

subsequently since the numerical solution in the reference simulation approaches chemical equilibrium in the region where the simple system is employed in model adaptive simulations.

5.3.2 Reaction front

In this section, we discuss the results of the model adaptive simulations for an advecting reaction front.

Initial data

The values of the primitive variables, in the entire computational domain, are set to $p = 10^6, T = 2000, v = 500$ and the density of atomic oxygen is set to $\rho_O = 0.01$. Based on these values, the equilibrium value $\mathbf{U}_{eq}$ is calculated. Then, for $x \in \Omega$ the density of molecular oxygen is perturbed to $\rho_{O_2}(x) := \rho^{eq}_{O_2} \cdot (1 + 0.01 \cdot \exp(-80 \cdot x^2))$, where $\rho^{eq}_{O_2}$ is the equilibrium value of the density of molecular oxygen. Momentum, total energy and the total density of the fluid mixture is accordingly re-calculated. The complex system is employed everywhere in the first time step and the models to be employed and the decomposition of the domain is done at the end of each time step by employing the proposed model adaptation strategy. We present the results of the simulations at four different tolerance, the lowest tolerance employed being $\tau_r = 0.002, \tau_\kappa = 0.00002$. In the subsequent simulations, the tolerances are scaled by a factor of two and the factor of safety is set to $\mathfrak{f}_\varepsilon = 0.25$.

We employ the same notations introduced in the previous section to make observations and draw conclusions.

Observations

We first look at the results of the numerical solutions for tolerance 4τ and then compare the results of model adaptive simulations at different tolerances.

We note the following:

- The initialization gives rise to a reaction front travelling to the right. The simple system is employed everywhere, except the reaction front as seen in Figure 5.25.
- The size of the patch of cells, where the complex system is employed, decreases as the fluid mixture in the reaction front approaches chemical equilibrium, as seen by comparing the numerical solution at time $t = 1.5 \cdot 10^{-5}$ in Figure 5.25 and time $t = 6.75 \cdot 10^{-5}$ in Figure 5.26. From $t = 1.44 \cdot 10^{-4}$ onward, the simple system is employed in the entire computational domain, see Figure 5.27.
- We can also observe that the source term in the reference simulation is significantly smaller in the regions where the simple system is employed than in the regions

where the complex system is employed, which informs us that the model adaptation strategy leads to suitable domain decomposition.

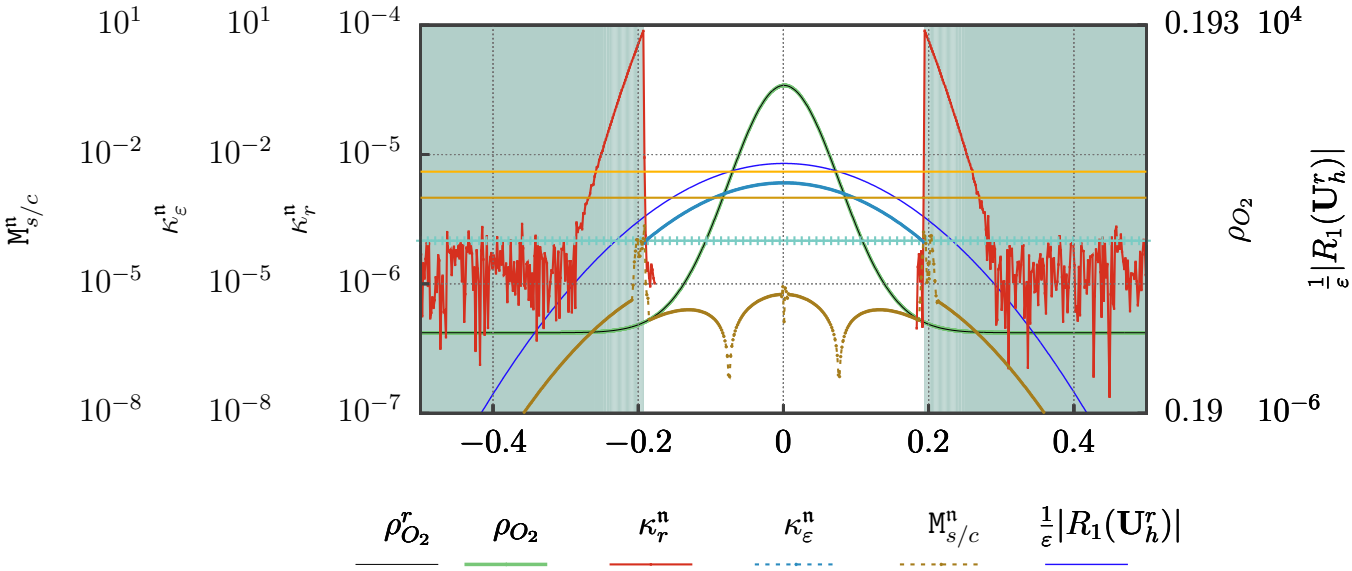

Figure 5.25: Advecting reaction front: $t = 1.5 \cdot 10^{-5}$

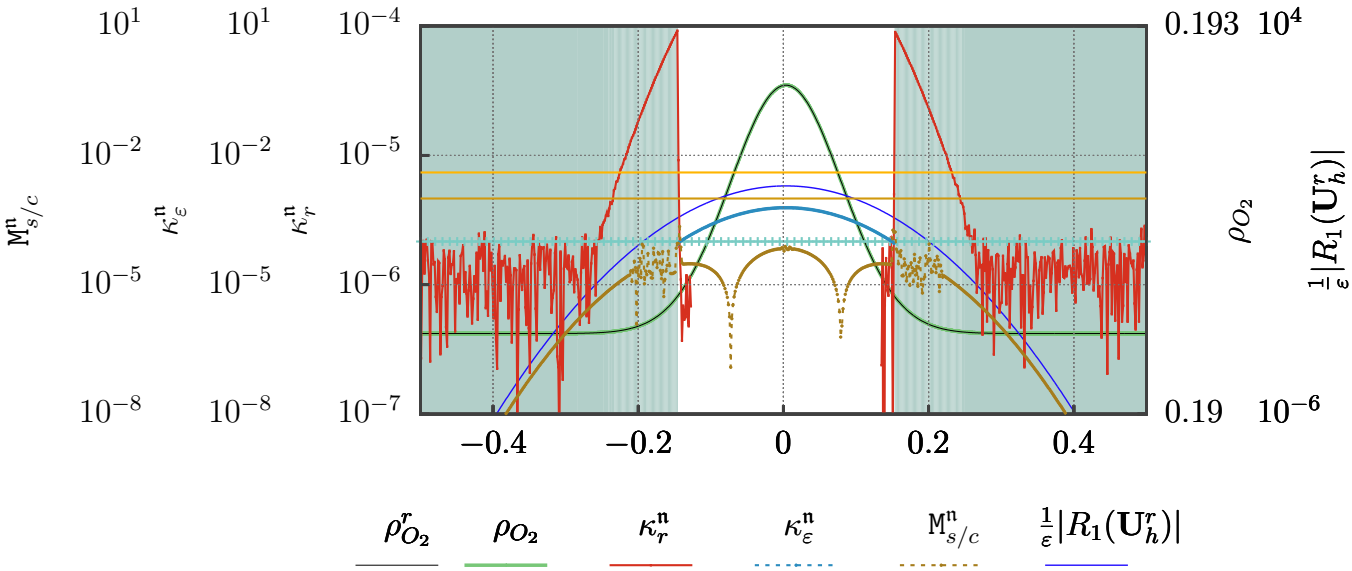

Figure 5.26: Advecting reaction front: $t = 6.75 \cdot 10^{-5}$

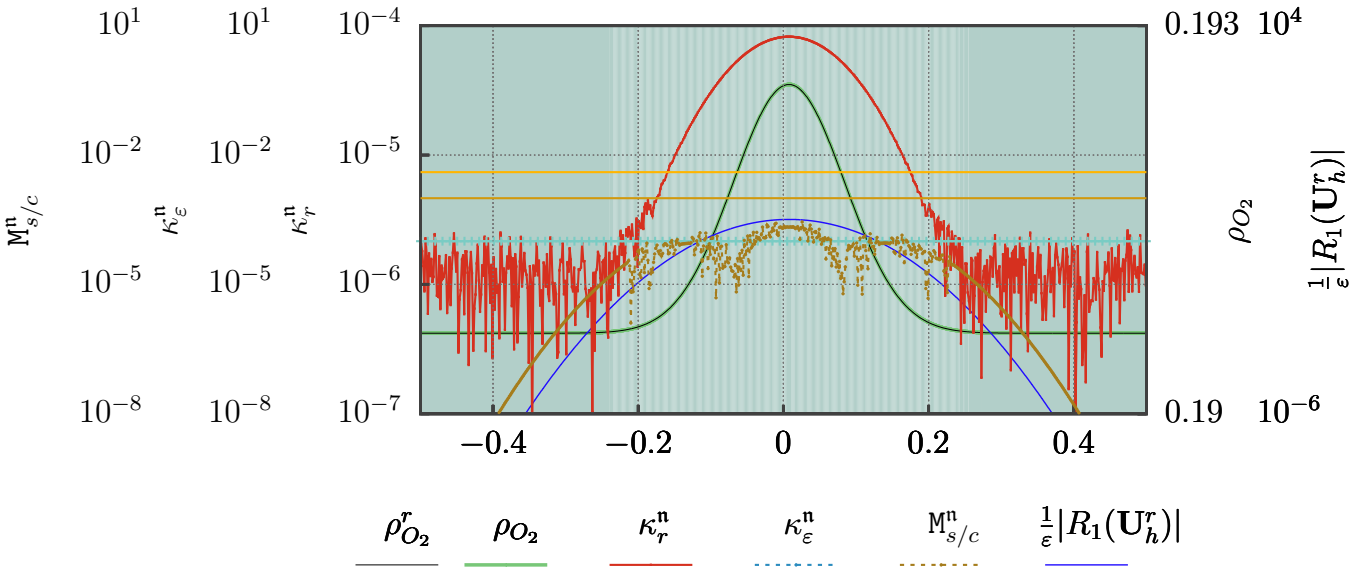

Figure 5.27: Advecting reaction front: $t = 1.44 \cdot 10^{-4}$

- Figure 5.28 shows the plots of the square root of the integral of the relative entropy between the reference simulation and the model adaptive simulation over the entire computational domain. The relative entropy between the model adaptive simulations and the reference simulations scale by a factor of two between two consecutive simulations, i.e. when comparing model adaptive simulations where the tolerances are twice in one simulation than in the other. This can be observed by looking at the initial peaks in the relative entropy and the peaks corresponding to the highest value for all the simulations in Figure 5.28. The difference in the long time behaviour when comparing model adaptive simulations of the shock tube and the reaction front can be attributed to the fact that in the case of reaction front, away from the reaction front, the modelling error indicator and the model coarsening distance decrease gradually. On the other hand, in the case of the shock tube, the numerical indicator, away from the waves, decrease sharply.

- In the long run, the relative entropy between the model adaptive simulation and the reference simulation decreases. This is due to the fact that the numerical solution in the reference simulation approaches chemical equilibrium.

5.3.3 Summary

In this section, we summarize some notable aspects of the model adaptation strategy and make some inferences based on them.

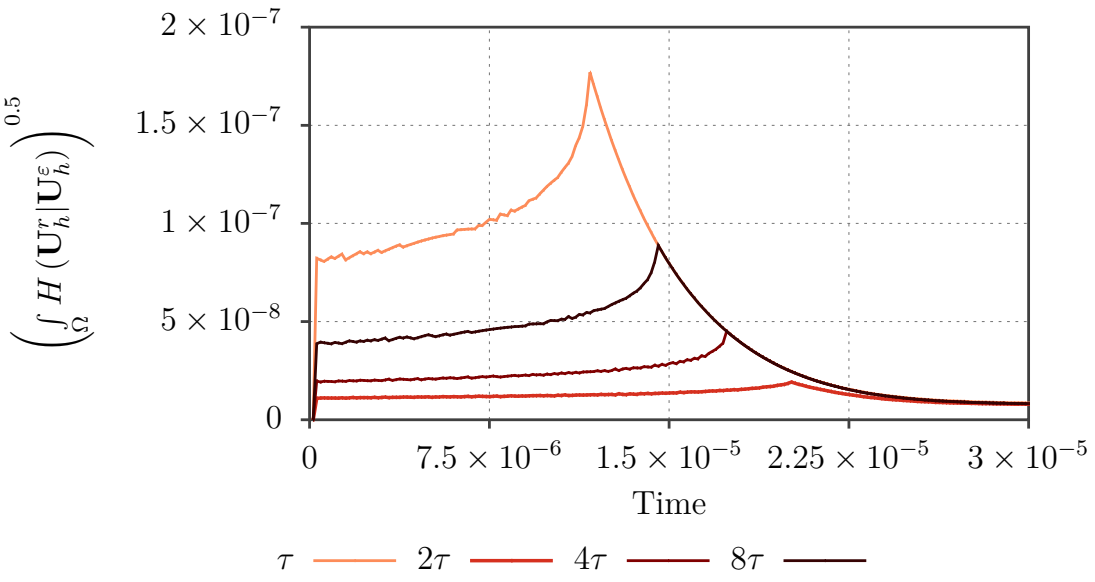

Figure 5.28: Relative entropy: advecting reaction front

We note the following:

- The model adaptation strategy leads to employment of the simple system, locally in time and space, in regions which are suitable empirically. The model adaptation strategy does not result in undesirable back and forth switch between the simple system and the complex system. The switch back to the complex system is only triggered if the number of cells in a patch where the simple system is employed falls below one. We can infer that the model adaptation strategy works well by ensuring that the numerical solution is close to the equilibrium manifold and ensuring that the dynamics is close to that of the equilibrium dynamics before switching to the simple system. This is also evidenced by the fact that the source term in the reference simulations is of a smaller magnitude in the regions where the simple system is employed.

- The model adaptation strategy automatically places the interfaces at points in the computational domain where the proposed coupling is expected to work reasonably well. This averts the production of artefacts which could negatively affect the accuracy of the model adaptive simulations.

5.4 Mesh adaptation

In this section, we discuss the results of mesh adaptive numerical simulations. We consider the case of a shock tube.

5.4.1 Shock tube

Numerical setup

The numerical setup is similar to that as described in Section 5.3.1, except that the complex system is used in the entire computational domain for the entirety of the simulation. The constants determining the speed of the reaction are set to $C = 2.9 \cdot 10^{13}, E = 597.5$. A uniform mesh is employed in the first time step and subsequently the mesh to be employed is determined at the end each of time step in accordance with the mesh adaptation strategy outlined in Section 3.4.4. The mesh adaptive simulations are compared with the reference simulation, i.e. the simulation with the same numerical setup but where a uniform mesh is employed throughout the simulation. Consequently, the mesh employed in the first time step in the reference and the mesh adaptive simulations is the same.

The numerical solutions of the mesh adaptive simulations are denoted by $\mathbf{U}_h^\delta$ and the reference numerical solution is denoted by $\mathbf{U}_h^r$. As done in the previous sections, the difference between the mesh adaptive simulations and the reference simulation in cell $\mathcal{V}_j$ is quantified by

$$\kappa_r^{\mathfrak{n},j} := \left(\frac{1}{\Delta x_j} \cdot \int_{\mathcal{V}_j} H\left(\mathbf{U}_{h,j}^r | \mathbf{U}_{h,j}^\delta\right) \mathrm{dx} \right)^{0.5}. \tag{5.16}$$

The mesh adaptive simulations are conducted at four different tolerances with the smallest tolerances being

$$\tau_r^s = 0.02 \ \tau_\kappa^s = 0.0002. \tag{5.17}$$

The tolerances in the subsequent simulations are scaled by a factor of 2. The mesh adaptive simulation at tolerances specified in (5.17) is informally referred to as the mesh adaptive simulation with tolerance τ and the subsequent simulations are referred to as mesh adaptive simulations with tolerances $2\tau, 4\tau$ and 8τ. The values τ_r and $2^{(p+1)\cdot L} \cdot \tau_r$, which are employed to determine the level to be employed in each cell, see Step 2 of Algorithm 4, are shown in the plots with green horizontal lines and the tolerance employed for the mesh coarsening distance, i.e. τ_κ, is shown with a blue horizontal line. The factor of safety in the mesh adaptive simulations is set to $\mathfrak{f}_\delta = 2$.

To study the simulation results, the numerical indicators employed in the mesh adaptive algorithm are plotted, namely the discretization error indicators for the complex system $(\mathtt{D}_c^{\mathfrak{n}})$, the mesh coarsening distance $\kappa_\delta^{\mathfrak{n}}$ and the mesh refinement level in each cell $\mathcal{L}^{\mathfrak{n}}$.

Recall that the numerical simulations are limited to remove oscillations. It was observed, and has also been previously noted in [57], that the limiting process interacts with mesh adaptation. To enable the discussion on the interaction between mesh adaptation and limiting, we employ limiting indicators Δ^r and Δ^δ for the reference and the mesh adaptive simulations respectively. The limiting indicators take the value 1 if the numerical solution is limited in a cell and the value 0 if it is not.

Observations

The following observations were made in the mesh adaptive simulations:

- Initially, the highest level is employed in the region where the contact discontinuity, the shock and the rarefaction wave is. The mesh employed gradually transitions from the most refined to the most coarse mesh away from the waves. As the waves move away from each other, regions of plateaus are created between the contact discontinuity and the shock and the shock and the rarefaction wave. Once the mesh coarsening distance and the discretization error indicator is small enough, coarser mesh is employed at the plateaus, see Figures 5.29 and 5.30, which show the plots of the density of molecular oxygen and the numerical indicators employed in mesh adaptation at time $1.875 \cdot 10^{-5}$ and $6 \cdot 10^{-5}$ respectively for mesh adaptive simulation with tolerance 2τ.

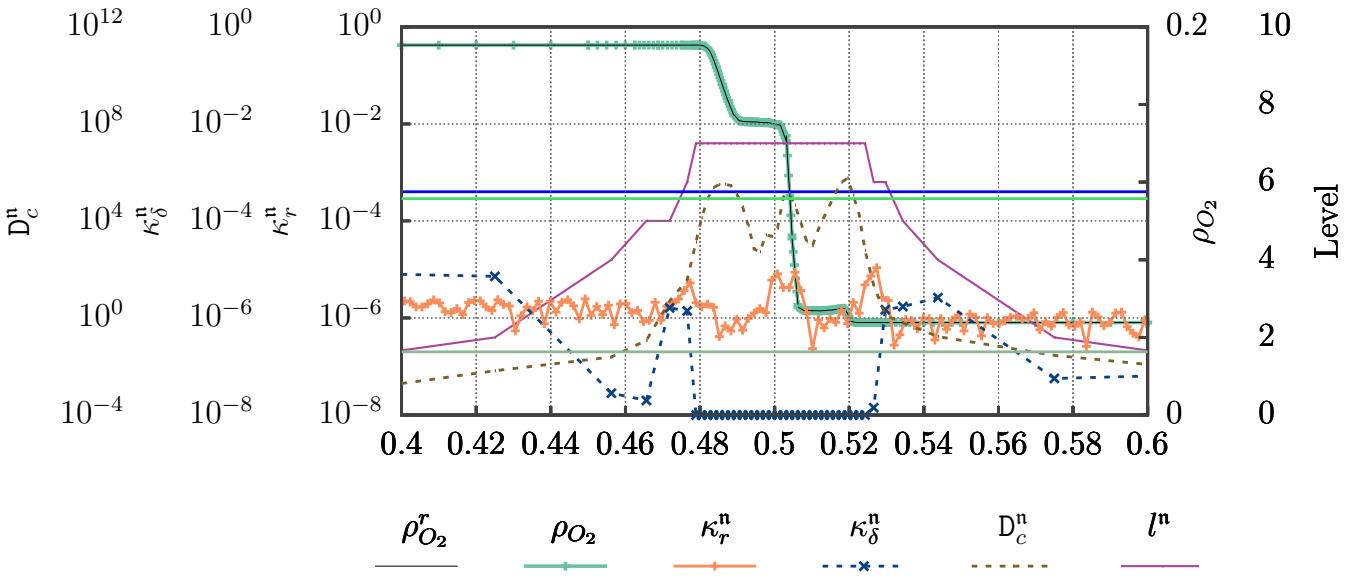

Figure 5.29: Shock tube at $t = 1.875 \cdot 10^{-5}$

- Recall that limiting is done in accordance with Algorithm 1, where the so called Shu constant $\mathfrak{m}$ is employed in order to avoid limiting in regions where the numerical solution is locally smooth. This is ensured by limiting the numerical solution in some cell $\mathcal{V}_i$, if and only if the slope of the characteristic variables is greater than $\mathfrak{m}\Delta x_i^2$, where Δx_i is the mesh width in that cell. When comparing the numerical solution of a mesh adaptive solution in a cell to that of the reference solution in the same cell, i.e. a cell on the highest level in the mesh adaptive solution and the cell in the same spatial position in the reference simulation, occasionally it can be observed that the slope of the characteristic variables falls marginally below the value $\mathfrak{m}\Delta x_i^2$ in one of the simulations and marginally above in the other. As a result, the numerical solution is limited in the simulation in which the slope is larger than the value $\mathfrak{m}\Delta x_i^2$ and not the other. This can be observed in Figures 5.31-5.34, which show the plots of mesh adaptive simulation with tolerance 2τ at two consecutive time

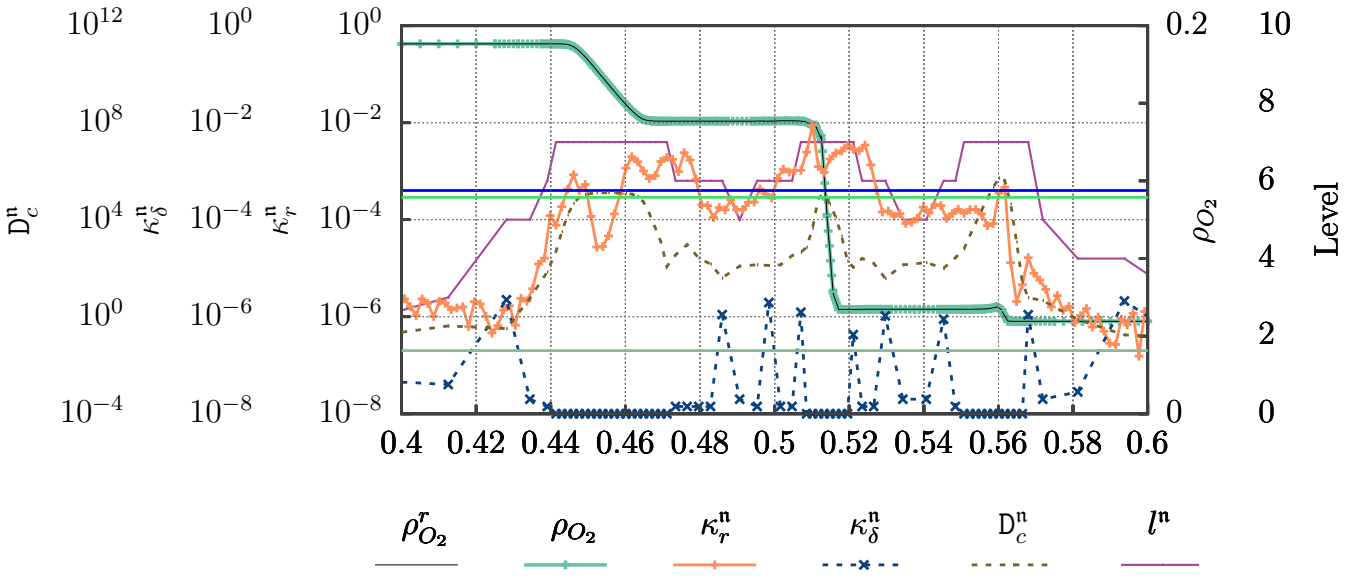

Figure 5.30: Shock tube at $t = 6 \cdot 10^{-5}$

steps. We can observe that the numerical solution in the mesh adaptive simulation is limited at $x \approx 0.485$ and not in the reference simulation, leading to a sharp increase in the value of κ_r^{n} in Figure 5.32. Similarly in Figure 5.34, the reference simulation is limited in the cell at $x \approx 0.485$ while the numerical solution in the mesh adaptive simulation is not. The difference in behaviour of the limiting is a result of perturbations introduced in the mesh adaptive simulations when compared to the reference simulation. For more information about the interplay between limiting and mesh adaptation, the reader is referred to [57].

- Recall that the residual employed to calculate the discretization error indicators in some cell $\mathcal{V}_i$ scales according to $\mathcal{O}\left(\Delta x_i^p\right)$, p being the order of the polynomial employed in that cell, see Section 3.4. Hence, limiting the numerical solution leads to an increase in the residual and as a result in the discretization error indicator. This can trigger local refinement of the mesh. Practically, this manifests itself in two ways:

  1. When the mesh is to be coarsened, i.e. two fine cells are to be replaced by a coarser cell with twice the mesh width of the fine cell, it can happen that the numerical solution in the fine cells is such that it is not limited, but the numerical solution in the coarsened cell is limited. This results in an increase in the discretization error indicator above the tolerance prescribed for cells with the level of mesh refinement that particular cell has. This is due to the fact that the order of polynomial falls to $p = 1$, once the numerical solution is limited. The increase in the discretization error indicator leads to refining the

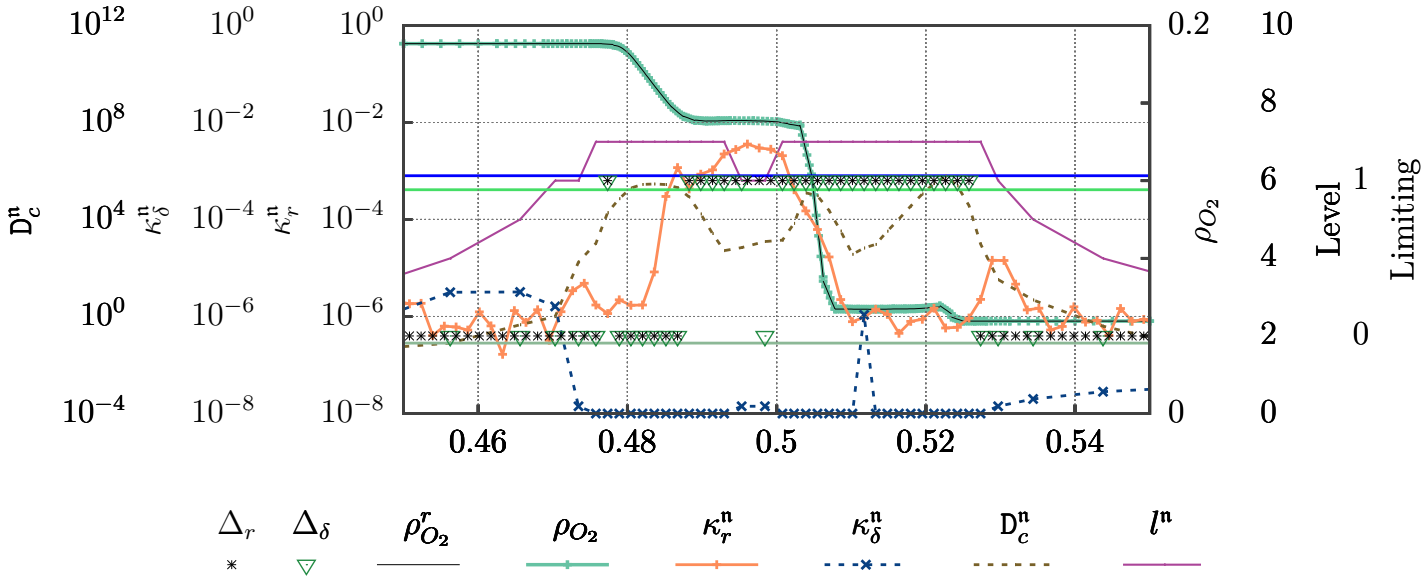

Figure 5.31: Shock tube at $t = 2.225 \cdot 10^{-5}$

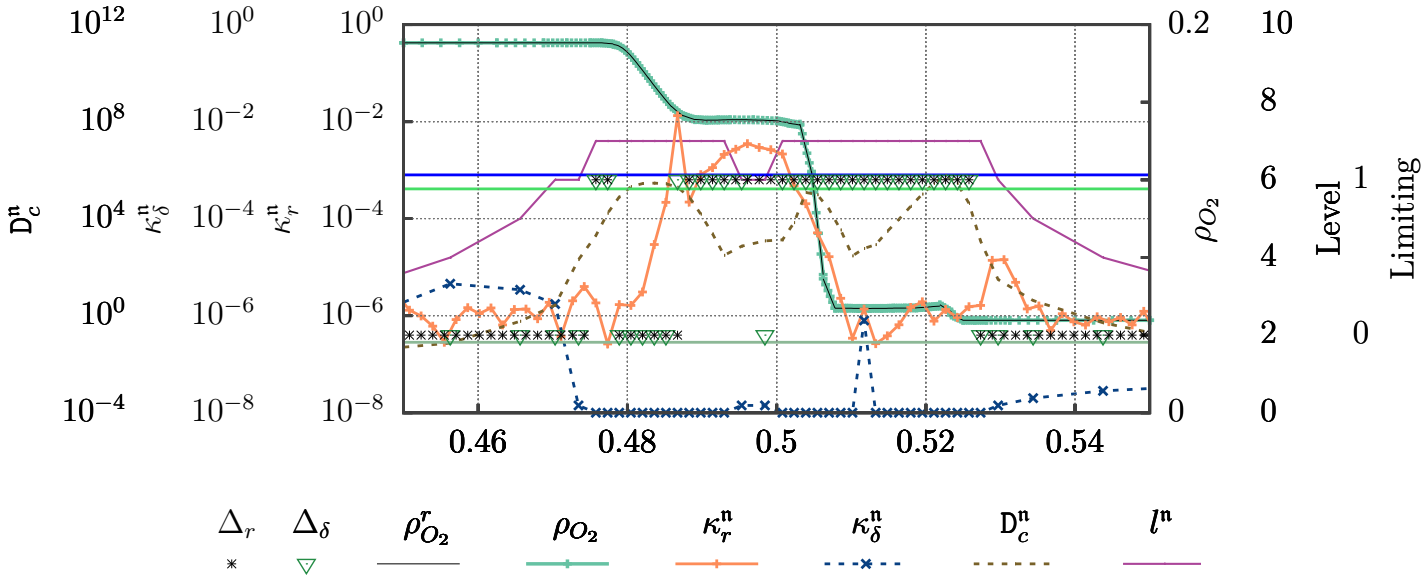

Figure 5.32: Shock tube at $t = 2.2375 \cdot 10^{-5}$

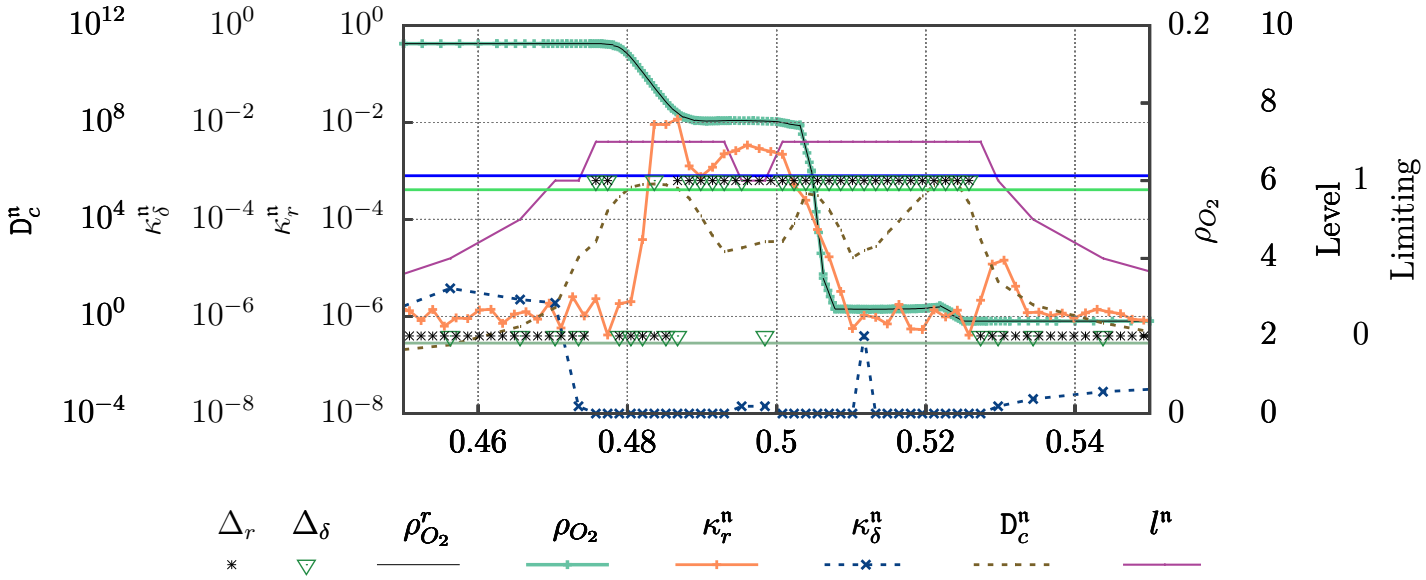

Figure 5.33: Shock tube at $t = 2.25 \cdot 10^{-5}$

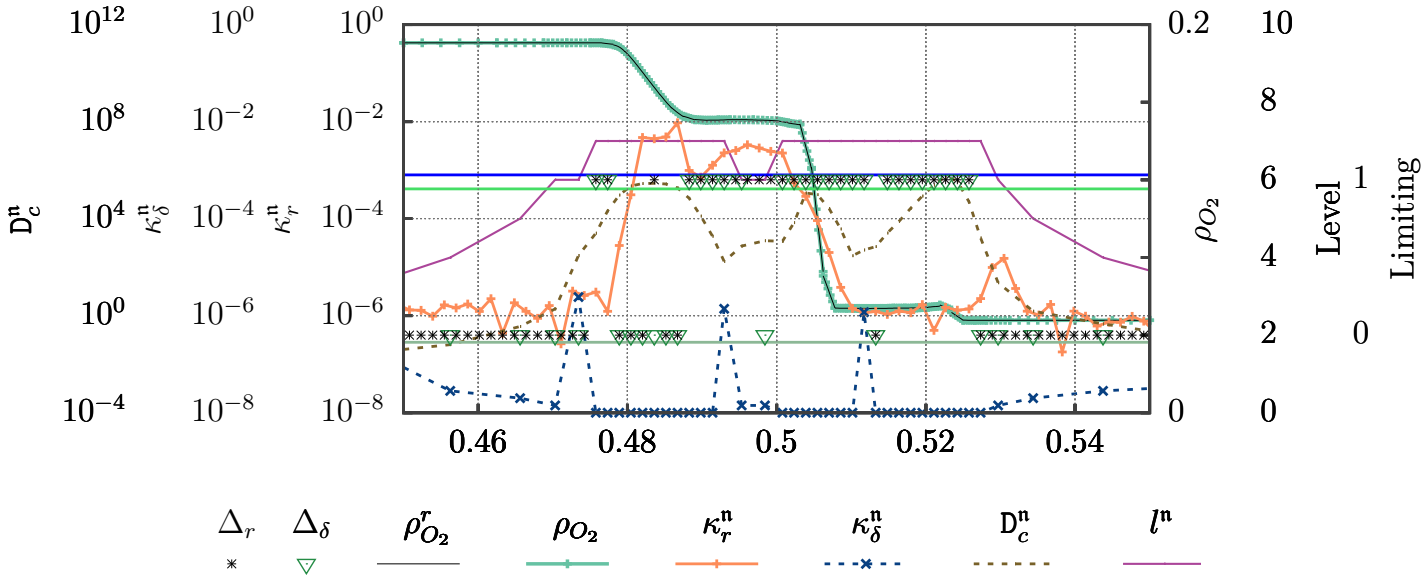

Figure 5.34: Shock tube at $t = 2.22625 \cdot 10^{-5}$

cell which was momentarily coarsened. This can be observed in Figures 5.35-5.37 showing plots for the mesh adaptive simulation with tolerance τ, where the numerical solution in the cell (on level 6) at $x \approx 0.52$ is limited in the cell in Figure 5.36, which leads to mesh refinement in Figure 5.37.

2. If the numerical solution is limited in a safety cell, i.e. one of the cells added to the left and right of a patch of cells on the highest level (see Section 3.4.4), then limiting the solution in that cell increases the discretization error indicator in that cell above the tolerance of the level $L-1$, and as a result it is no longer a safety cell. The increase in the discretization numerical indicator results in cells to be additionally appended to the left and right of the current patch where cells are on the highest level. These cells become the new safety cells. This can be observed in Figures 5.35-5.37 where the numerical solution in the mesh adaptation simulation is limited in Figure 5.36 at $x \approx 0.495$ leading to appending additional cells in the next time step in Figure 5.37.

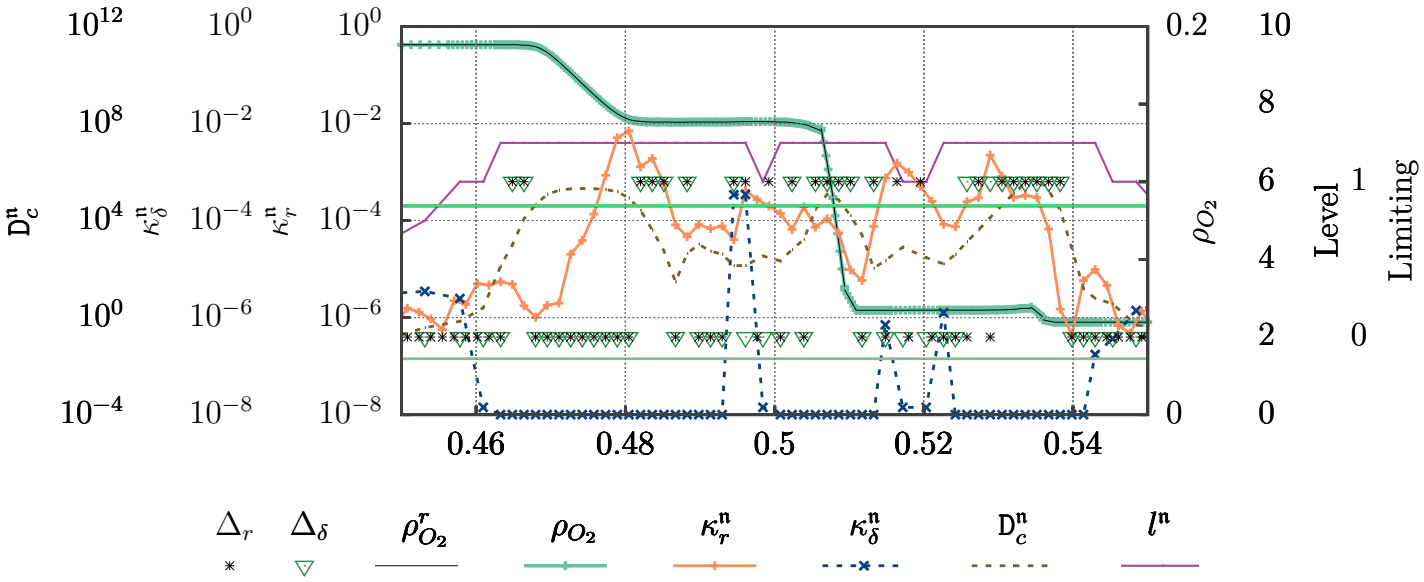

Figure 5.35: Shock tube at $t = 2.675 \cdot 10^{-5}$

- Figure 5.38 shows the plot of the square root of the relative entropy between the mesh adaptive simulations and the reference simulation for all tolerances. We can observe, as expected, that the long time behaviour scales linearly with the tolerances.

- Figure 5.39 shows the plot of the total number of cells employed throughout the mesh adaptive simulations. In the simulations conducted, the number of cells employed at the end of the simulations is 170 and 132 in the mesh adaptive simulations with the smallest and highest tolerance as compared to 1280 cells in the reference simulation. Note that in the mesh adaptive simulations, the number of cells reduces

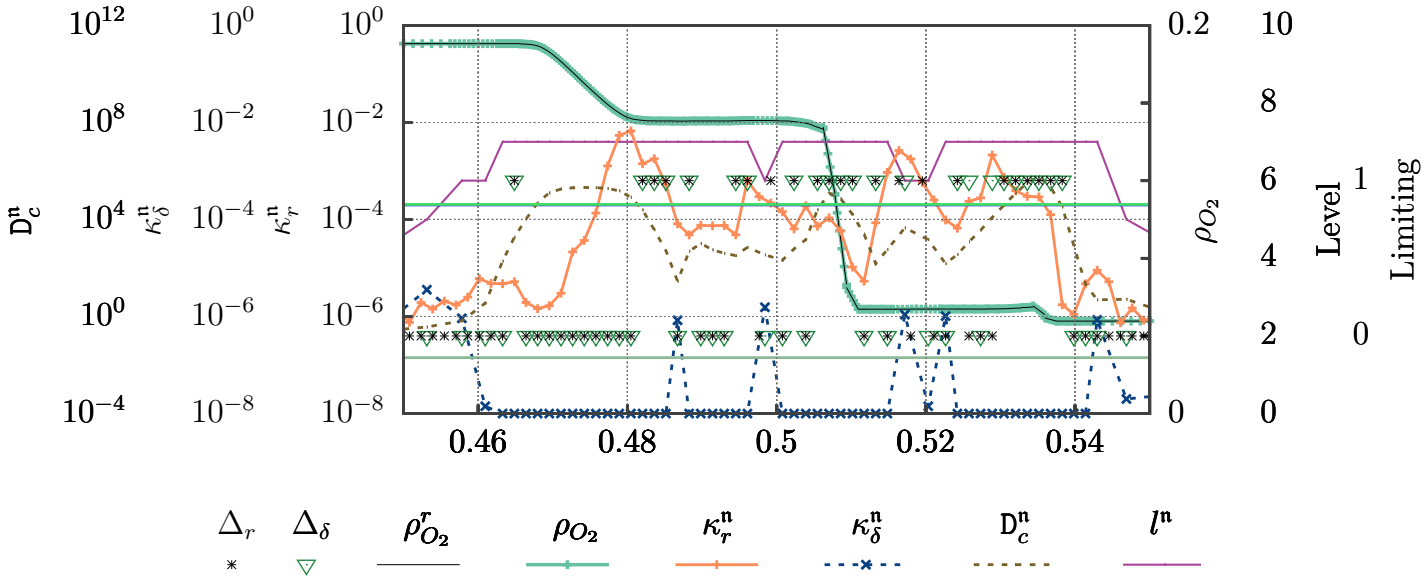

Figure 5.36: Shock tube at $t = 2.5625 \cdot 10^{-5}$

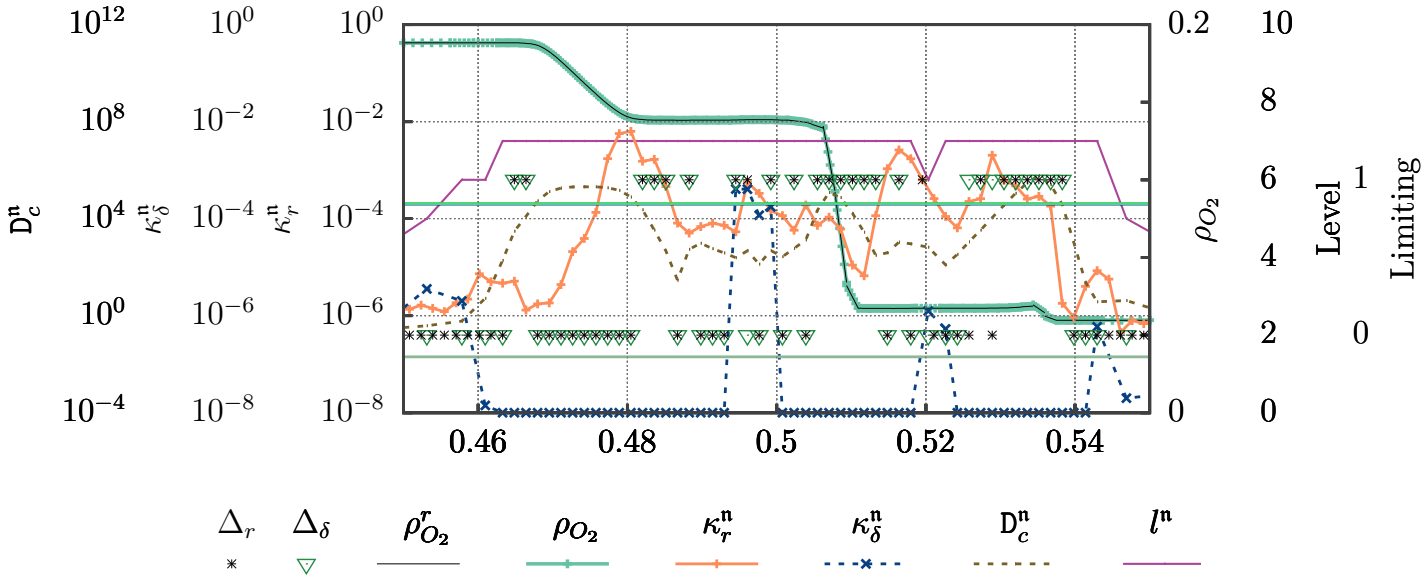

Figure 5.37: Shock tube at $t = 2.575 \cdot 10^{-5}$

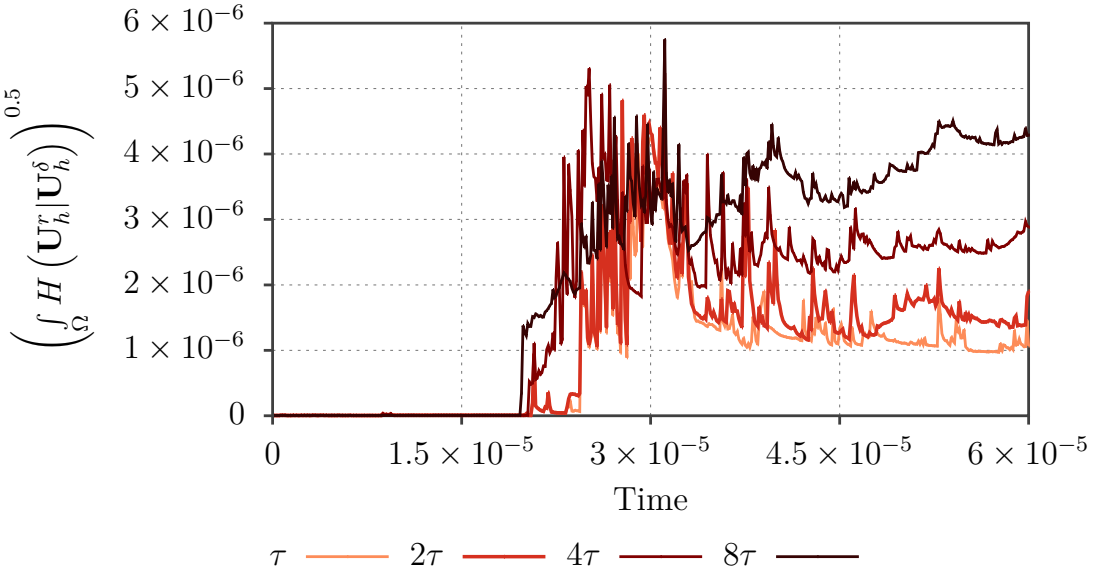

Figure 5.38: Relative entropy: Shock tube

over a few time steps from 1280 (not shown) to about 48 cells, after which the number of cells increases again as the waves separate requiring fine mesh at kinks and discontinuities.

5.4.2 Summary

Based on the numerical experiments conducted, we can note the following about the proposed mesh adaptation strategy:

- The mesh adaptation strategy works reasonably well and ensures the jump discontinuities are adequately resolved. The information used to decide upon the mesh refinement to be employed is the residual, which provides information about the extent to which the numerical solution satisfies the system of equations. The mesh coarsening distance provides complementary information allowing us to ensure that the loss in information is sufficiently small in the case that we do coarsen the mesh.

- Limiting interacts with mesh adaptation. This is to be expected, since by both mesh coarsening and limiting we discard some information. Although it might be possible to remove the interaction between the two, since mesh adaptation and limiting perform two different functions, namely reducing the computational resources needed to carry out the simulations and suppressing artefacts, it would not be appropriate to do so and allowing the two to independently serve their function is preferred.

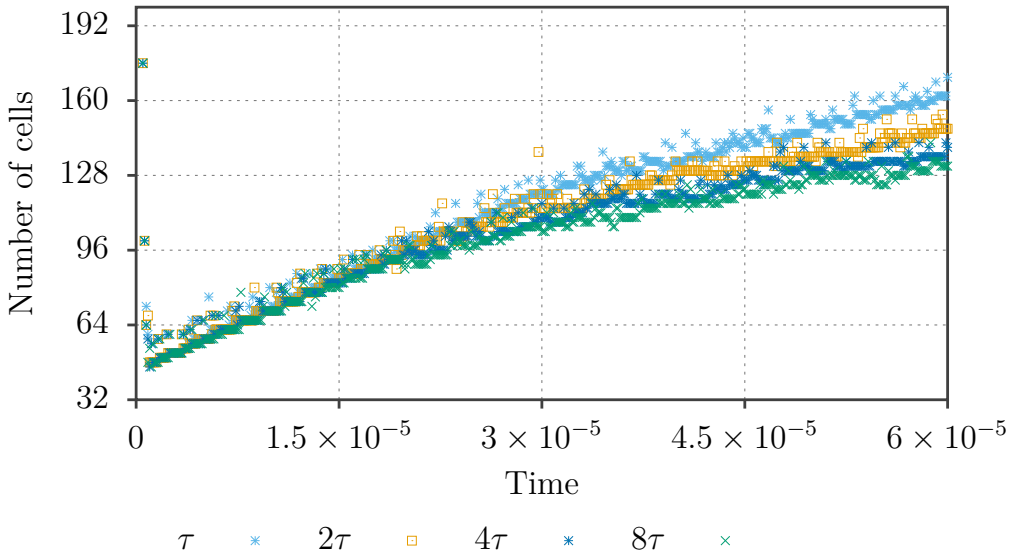

Figure 5.39: Total number of cells: Shock tube

5.5 Mesh and model adaptation

In this section, we present and discuss the numerical results of simultaneous mesh and model adaptive simulations. We consider the case of a shock tube.

5.5.1 Shock tube

Initial data and notation

The numerical setup is identical to that as employed in Section 5.3 for model adaptive simulations. The complex system and a uniform mesh is employed in the first time step, after which the mesh and model to be employed is determined according to the mesh and model adaptation strategy described in Section 3.3. The mesh and model adaptive simulations are conduced at four different tolerances, the simulation with the smallest tolerance being

$$\tau_\epsilon^s = 0.04, \tau_\kappa^s = 0.0004. \tag{5.18}$$

As done before, subsequent simulations are conducted at tolerances with values twice as that in the previous simulation. In addition, to enable the discussion about mesh and model adaptive simulations, the mesh and modelling error indicators $\mathtt{M}_s^{\mathtt{n}}, \mathtt{M}_c^{\mathtt{n}}, \mathtt{D}_s^{\mathtt{n}}, \mathtt{D}_c^{\mathtt{n}}$ and the mesh and model coarsening distances $\kappa_\varepsilon^{\mathtt{n}}, \kappa_\delta^{\mathtt{n}}$ are plotted where appropriate. The mesh and model adaptive simulations are denoted by $\mathbf{U}_h^\alpha$ and the reference simulation by $\mathbf{U}_h^r$ and the difference between the mesh adaptive simulations and the reference simulation in

some cell $\mathcal{V}_j$ is quantified by

$$\kappa_r^{\mathrm{n},j} := \left(\frac{1}{\Delta x_j} \cdot \int_{\mathcal{V}_j} H\left(\mathbf{U}_{h,j}^r | \mathbf{U}_{h,j}^\alpha \right) \mathrm{dx} \right)^{0.5}. \tag{5.19}$$

Observations

The following observations were made in the mesh and model adaptive simulations:

- Figures 5.40 and 5.41 show the plots of the density of molecular oxygen (ρ_{O_2}), the mesh refinement level and κ_r^{n} for mesh and model adaptive simulations with tolerances τ and 4τ. The regions where the simple system is employed is shown in a solid green and a hatched red background for the mesh and model adaptive simulation with tolerance 4τ and τ respectively. Observations similar to those made in simulations, where solely the mesh or model adaptation is performed, can be noted. The model and mesh is coarsened at the plateaus, i.e. between the shock and the contact discontinuity and between the contact discontinuity and the rarefaction. Naturally, the simple system and a coarser mesh is employed in larger regions in the simulation with tolerance 4τ than the simulation with tolerance τ.

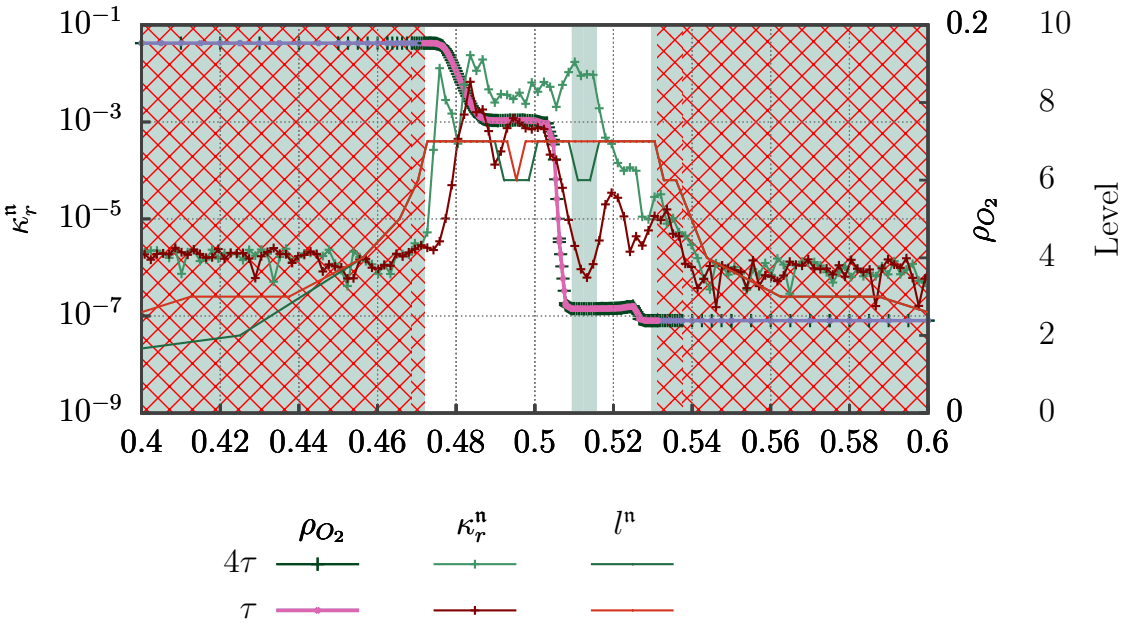

Figure 5.40: Shock tube: $t = 2.5 \cdot 10^{-4}$

- Mesh adaptation does not affect the model adaptation and vice versa, i.e. mesh coarsening does not lead to model refinement and conversely model coarsening does not lead to mesh refinement. Mesh and model adaptation for the most part function independently. Two exceptions noted are as follows:

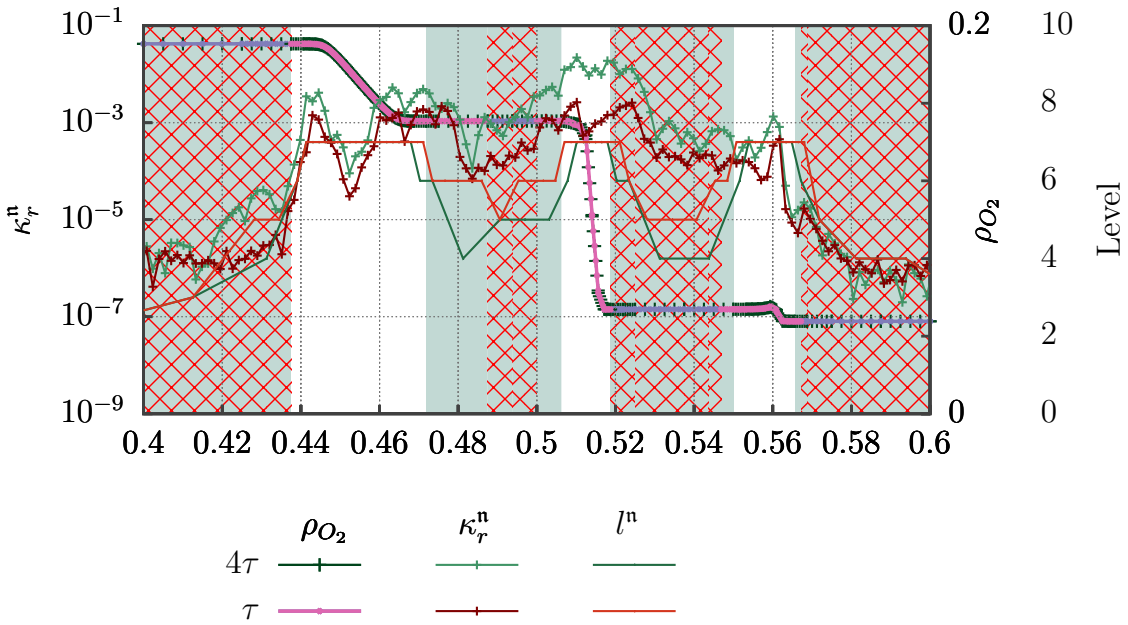

Figure 5.41: Shock tube: $t = 6 \cdot 10^{-4}$

– If coarsening the mesh reduces the number of cells in a patch where the simple system is being employed to just one cell, it triggers model refinement in the next time step. This is due to the fact that we do not allow employing the simple system in single cells. This can be observed in Figures 5.42 - 5.44, showing plots at three consecutive time steps of mesh and model adaptive simulation with tolerance 4τ. It can be observed that coarsening the mesh at $x \approx 0.51$ leads to a single cell where the simple system is employed, triggering model refinement.

– Recall that when the mesh at coupling interfaces is to be coarsened, i.e. when two fine cells are to be combined to give a coarser cell and the fine cells are such that the simple system is employed in one cell and the complex system in the other, then the mesh is coarsened and the complex system is employed in the coarsened cell. Hence, this leads to a shift in the position of the interface. This can be observed in Figure 5.45 and Figure 5.46, which show plots at two consecutive time steps of the mesh and model adaptive simulation with tolerance 4τ. We can observe that the mesh is coarsened at the interface at $x \approx 0.485$ in Figure 5.46, which leads to a shift in the interface to the right.

- Observations made in Sections 5.3 and 5.4 regarding the mechanics of model adaptation and mesh adaptation in pure mesh and model adaptive simulations also hold true in simultaneous mesh and model adaptive simulations.

- The mesh and model adaptive simulations qualitatively agree with the reference

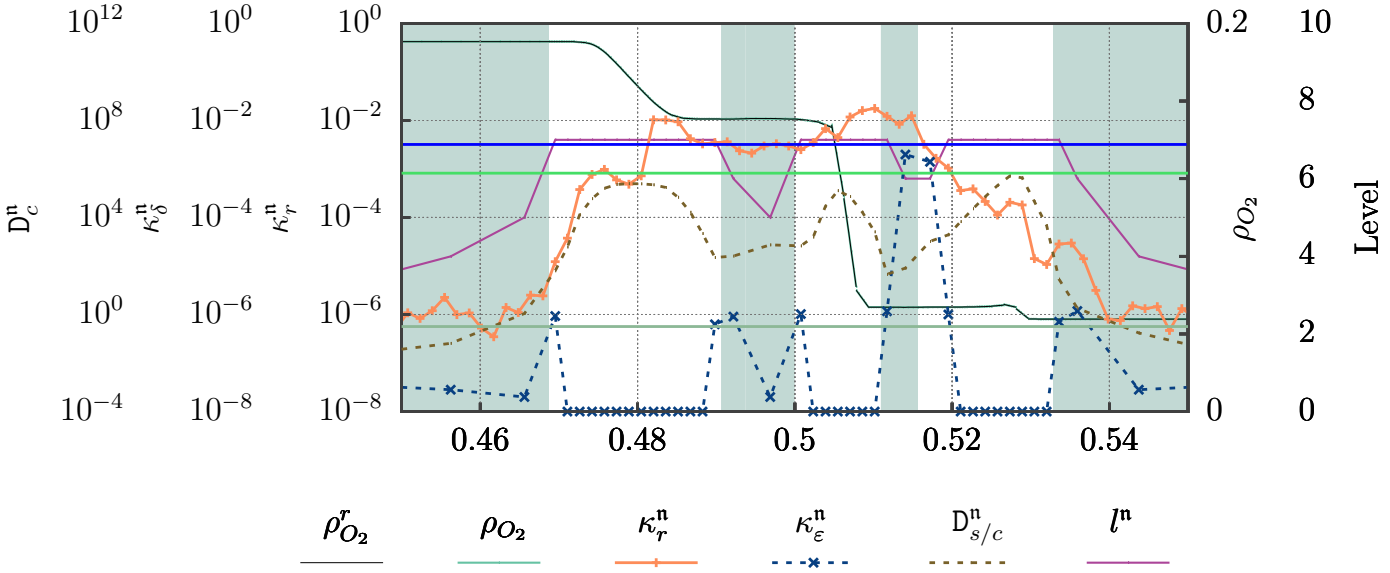

Figure 5.42: Shock tube: $t = 1.5875 \cdot 10^{-4}$

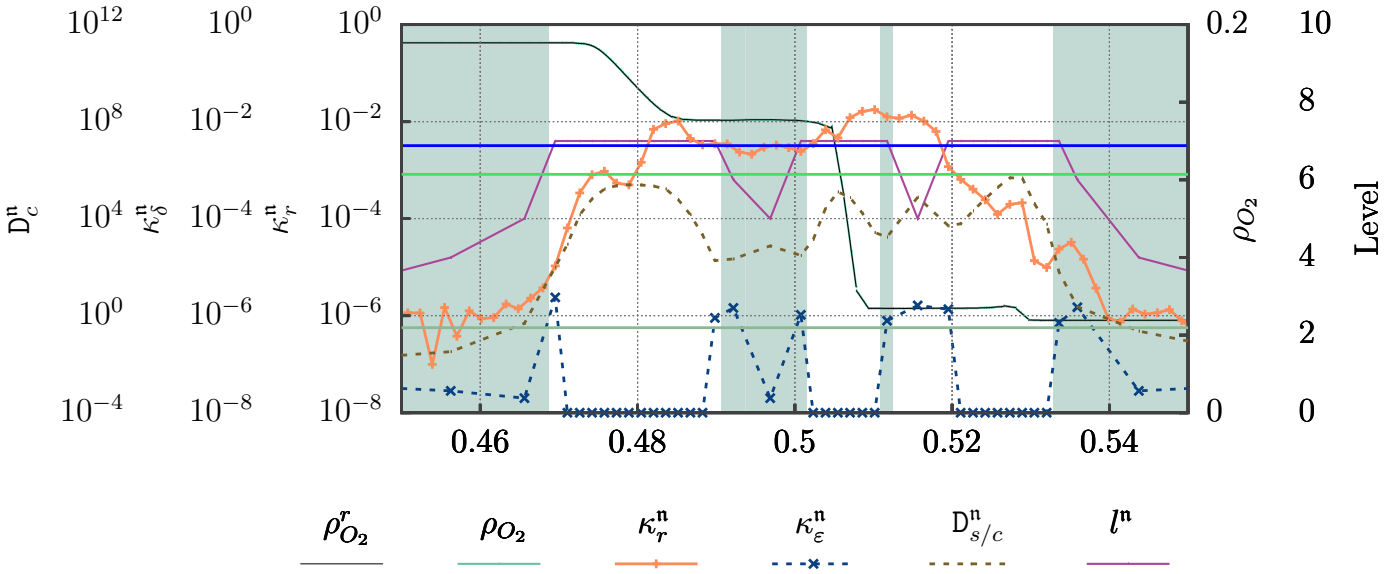

Figure 5.43: Shock tube: $t = 1.6 \cdot 10^{-4}$

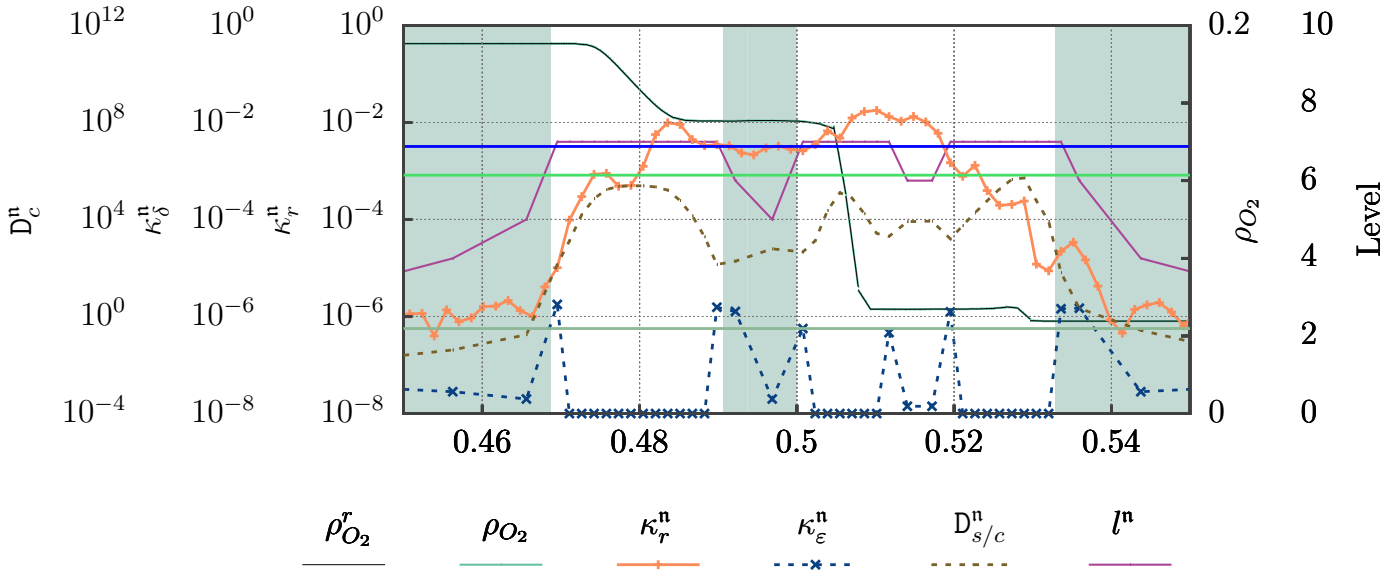

Figure 5.44: Shock tube: $t = 1.6125 \cdot 10^{-4}$

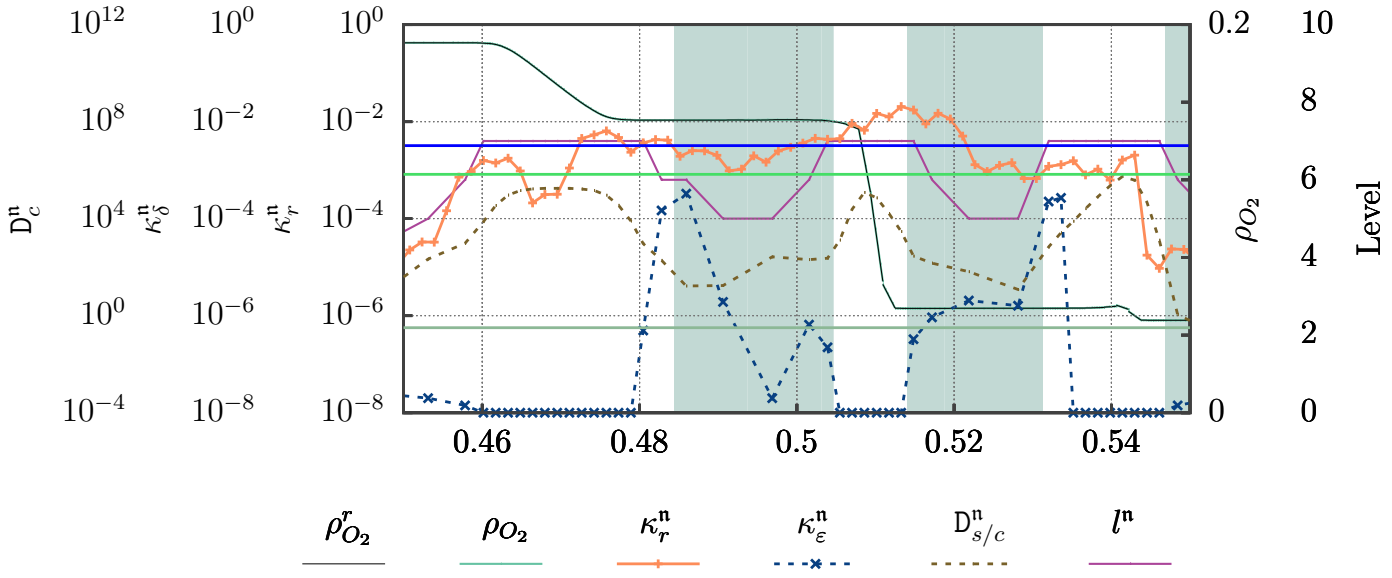

Figure 5.45: Shock tube: $t = 1.5875 \cdot 10^{-4}$

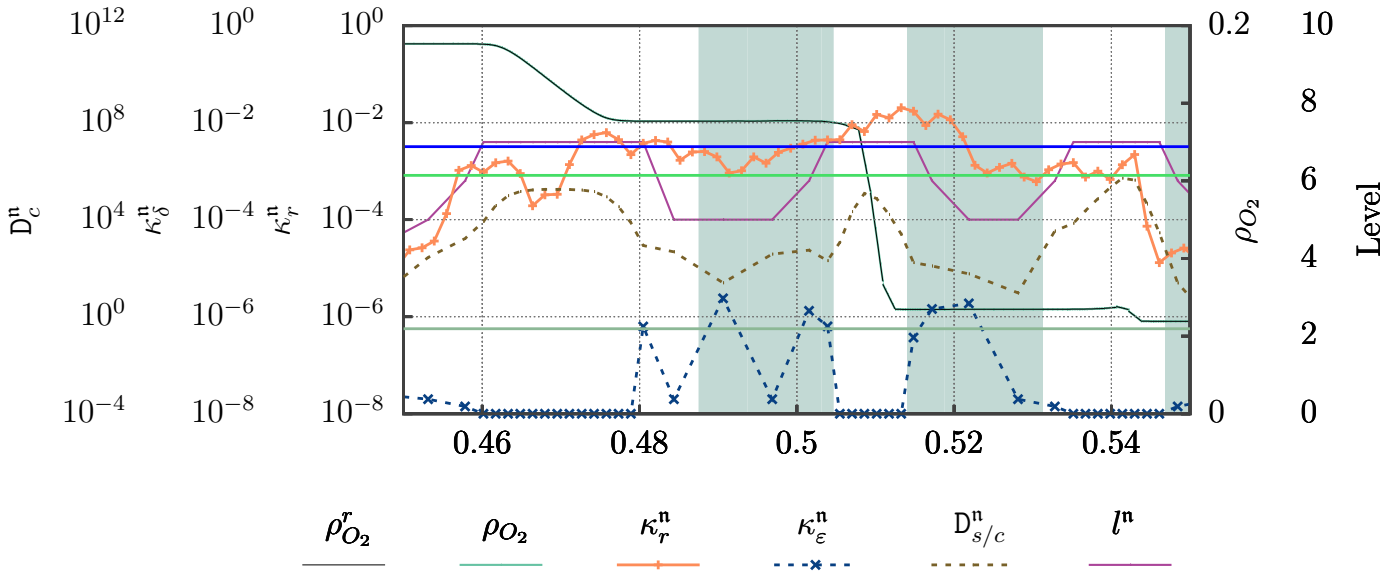

Figure 5.46: Shock tube: $t = 1.6 \cdot 10^{-4}$

simulation and the mesh and model adaptation strategy does not lead to decomposed domains which would lead to interfaces placed at positions which could adversely affect the accuracy of the adaptive simulations.

- Figure 5.47 shows the plots of the square root of the integral of the relative entropy between the reference and the mesh and the model adaptive simulations. We can observe that the long term behaviour, as expected, scales approximately linearly with the tolerances being employed in the simulations.

- Figure 5.48 shows the plots of the number of cells employed with respect to time for the mesh and model adaptive simulations. In the simulations conducted, the number of cells employed at the end of the simulations were 146 and 118 in the simulations with the smallest and highest tolerance as compared to 1280 cells in the reference simulation

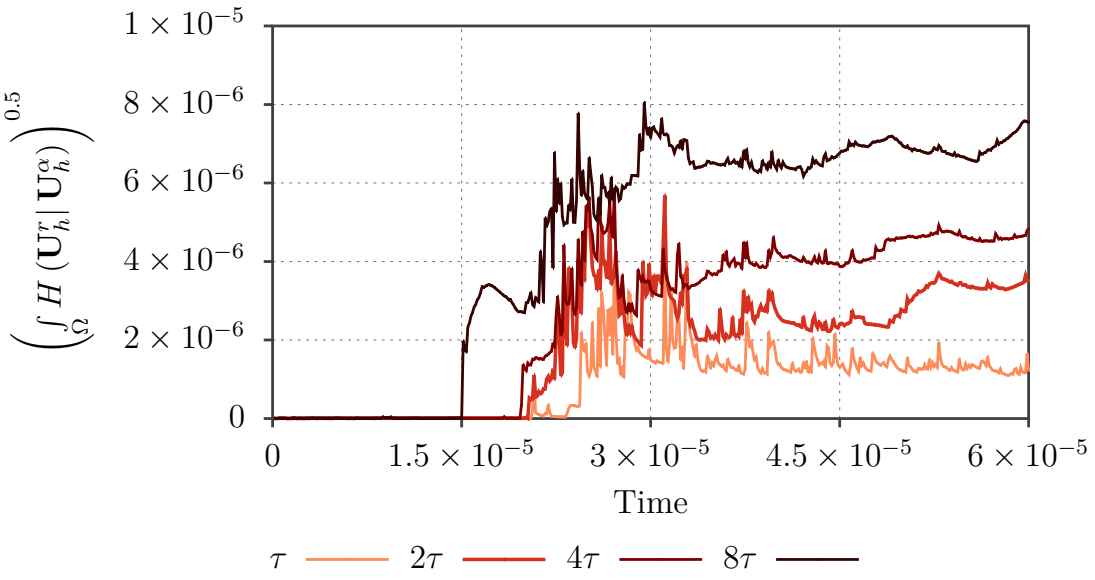

Figure 5.47: Relative entropy: Shock tube

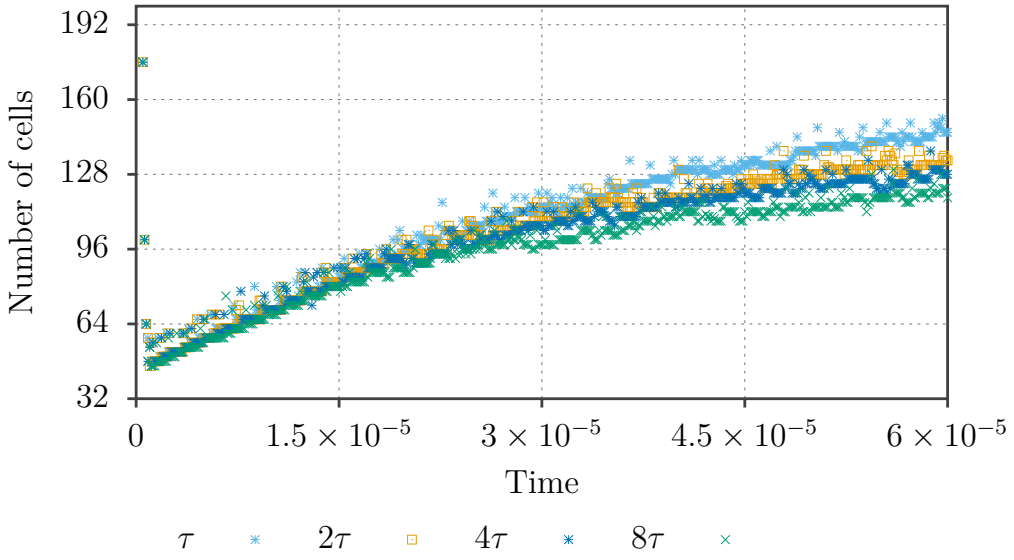

Figure 5.48: Total number of cells: Shock tube

5.5.2 Summary

Based on the numerical experiments conduced, we can conclude that the proposed mesh and model adaptation strategy works well. Features such as discontinuities and kinks are resolved well and the coarse model is employed in regions where it is qualitatively appropriate to do so. The mesh adaptation does not significantly interact with model adaptation. Mesh and model adaptation work independently and lead to suitable levels of mesh and model refinement.

Chapter 6

Outlook

The focus of this thesis was devising a mesh and model adaptation strategy for a model hierarchy consisting of two levels of model complexity where the fine model, referred to as the complex model, consists of a system of hyperbolic balance laws and the coarse model, referred to as the simple system, consists of a system of hyperbolic conservation laws. We demonstrated the efficacy of the proposed mesh and model adaptation strategy by conducting simulations. We give a brief outlook on further potential developments and prospective avenues of research.

1. In the cells where the simple system is employed, the Maxwellian needs to be frequently calculated. Program profiling, [75], showed that a single evaluation of the Maxwellian is computationally expensive and proves to be the costliest component of the implementation. This is due to the fact that calculating the Maxwellian consists of solving a non-linear system of equations. The problem is exacerbated by the fact that in each time step several Maxwellian evaluations are necessary for all the cells where the simple system is employed. The high cost of Maxwellian evaluation could be alleviated by replacing the Maxwellian by an approximate Maxwellian which consumes a smaller amount of computational resources. This could be done, as an example, by employing physics informed machine learning, cf. [41],[64],[71].

2. A Lipschitz continuous reconstruction of the numerical solution needs to be computed in order to enable us to employ the derived error estimates. The reconstruction technique can be extended to Cartesian meshes in two space dimension, cf. [44], but the extension to general meshes in two and three space dimension remains an open problem.

3. The proposed coupling is expected to work well when the numerical solution in the cells in the vicinity of the interface is close to the equilibrium manifold. However, coupling interfaces are nevertheless sources of errors. A further topic of research could be quantifying the errors created due to the presence of interfaces and to propose a modified adaptation strategy.

4. The real world cost savings when employing the proposed mesh and model adaptation strategy could be investigated by conducting tests for more complex problems

by employing a sufficiently well optimized code. An example could be simulating air flows in Earth's atmosphere by simulating 21 reactions (Model of Park, cf. [46]) or 26 reactions (Model of Dunn and Kang, cf. [78]) using the 11 species air model.

Bibliography

[1] A. Ambroso, C. Chalons, F. Coquel, E. Godlewski, F. Lagoutière, P.-A. Raviart, and N. Seguin. Homogeneous models with phase transition: coupling by finite volume methods. *FVCA*, 4:483–492, 2005.

[2] A. Ambroso, C. Chalons, F. Coquel, E. Godlewski, F. Lagoutière, P.-A. Raviart, and N. Seguin. The coupling of homogeneous models for two-phase flows. *Int. J. Finite Vol.*, 4, 2007.

[3] C. Arvanitis, C. Makridakis, and N. Sfakianakis. Entropy conservative schemes and adaptive mesh selection for hyperbolic conservation laws. *J. Hyperbol. Difer. Eq.*, 7:383–404, 2010.

[4] D. Baulch, C. Cobbs, R. Cox, C. Esser, P. Frank, T. Just, J. Kerr, M. Pelling, J. Trow, R. Walked, and J. Warnatz. Evaluated Kinetic data for combustion modeling. *J. Phys.Chem.*, 21:411–734, 1992.

[5] R. Becker, M. Braack, R. Rannacher, and T. Richter. Mesh and model adaptivity for flow problems. *Reactive flows, diffusion and transport*, page 47–75, 2007.

[6] F. Berthelin and A. Vasseur. From kinetic equations to multidimensional isentropic gas dynamics before shocks. *SIAM J.Math.Anal*, 36:1807–1835, 2005.

[7] A. Birkefeld. *Computational Aeroacoustics with a High Order Discontinuous Galerkin Scheme*. PhD thesis, Universität Stuttgart, 2012.

[8] M. Borrel, L. Halpern, and J. Ryan. Euler–Navier–Stokes coupling for aeroacoustics problems. *Computational Fluid Dynamics*, 180:427–433, 2011.

[9] S. Boscarino, J. Qiu, G. Russo, and T. Xiong. High order semi-implicit WENO schemes for all-mach full Euler system of gas dynamics. *SIAM J. Sci. Comput.*, pages 368–394, 2022.

[10] D. Bothe and W. Dreyer. Continuum thermodynamics of chemically reacting fluid mixtures. *Acta Mech.*, 226:1757–1805, 2015.

[11] M. Braack and A. Ern. A posteriori control of modeling errors and discretization errors. *Multiscale Model. Simul.*, 1:221–238, 2003.

[12] A. Bressan, S. Čanić, M. garavello, M. Herty, and B. Piccoli. Flows on networks: recent results and perspectives. *EMS Surv. Math. S.*, 1:47–111, 2014.

[13] C. Cancès, F. Coquel, E. Godlewski, H. Mathis, and N. Seguin. Error analysis of a dynamic model adaptation procedure for nonlinear hyperbolic equations. *Commun. Math. Sci.*, 14:1–30, 2016.

[14] N. Chalmers and L. Krivodonova. A robust CFL condition for the discontinuous Galerkin method on triangular meshes. *J. Comput. Phys.*, 403:109095, 2020.

[15] F. Chalot, T. Hughes, and F. Shakib. Symmetrization of conservation laws with entropy for high-temperature hypersonic computations. *Comp. Sys. Eng.*, 1:495–521, 1990.

[16] M. Chase. *NIST-JANAF Thermochemical Tables*. American Institute of Physics, 1998.

[17] G.-Q. Chen, David-Hoff, and K. Trivisa. Global solutions to a model for exothermically reacting, compressible flows with large discontinuous initial data. *Arch. Ration. Mech. Anal.*, 166:321–358, 2003.

[18] H. Z. Y. Cheng and J. Qiu. A comparison of the performance of limiters for Runge-Kutta discontinuous Galerkin methods. *Adv. Appl. Math. Mech.*, 5:365–390, 2013.

[19] B. Cockburn, S. Hou, and C.-W. Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. the multidimensional case. *Math. Comp.*, 54:545–581, 1990.

[20] B. Cockburn, S. Y. Lin, and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. III. One-dimensional systems. *J. Comput. Phys.*, 84:90–113, 1989.

[21] B. Cockburn and C.-W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework. *Math. Comp.*, 52:411–435, 1989.

[22] B. Cockburn and C.-W. Shu. The Runge Kutta Discontinuous Galerkin method for conservation laws V: Multidimensional systems. *J. Comput. Phys.*, 141:199–224, 1998.

[23] C. Coclici and W. Wendland. Analysis of a heterogeneous domain decomposition for compressible viscous flow. *Math. Models Methods Appl. Sci.*, 11:565–599, 2001.

[24] C. Dafermos. Stability of motions of thermoelastic fluids. *J. Thermal Stresses*, 2:127–134, 1978.

[25] C. Dafermos. The second law of thermodynamics and stability. *Arch. Rational Mech. Anal.*, 70:167–179, 1979.

[26] C. Dafermos. *Hyperbolic conservation laws in continuum Physics*. Springer, 2010.

[27] S. R. de Groot and P. Mazur. *Non-equillibrium Thermodynamics*. Dover, 1962.

[28] C. de Lellis and L. Székelyhidi. On admissibility criteria for weak solutions of the Euler equations. *Arch. Ration. Mech. Anal.*, 195:225–260, 2010.

[29] R. Diperna. Uniqueness to solutions to hyperbolic conservation laws. *Indiana University Mathematics Journal*, 28:137–188, 1979.

[30] W. Dreyer, C. Guhlke, and R. Müller. Rational modeling of electrochemical double-layers and derivation of Butler-Volmer equations. *WIAS pre-print*, 1860, 2013.

[31] W. Dreyer, C. Guhlke, and R. Müller. Modeling of electrochemical double layers in thermodynamic non-equilibrium. *Phys. Chem. Chem. Phys.*, 40:27176–27194, 2015.

[32] W. Dreyer, C. Guhlke, and R. Müller. A new perspective on the electron transfer: recovering the Butler–Volmer equation in non-equilibrium thermodynamics. *Phys. Chem. Chem. Phys.*, 18:24966–24983–1189, 2016.

[33] E. Feireisl, B. J. Jin, and A. Novotný. Relative entropies, suitable weak solutions, and weak-strong uniqueness for the compressible Navier-Stokes system. *J.Math.Fluid Mech.*, 14:717–730, 2012.

[34] E. Feireisl, A. Novotný, and Y. Sun. Suitable weak solutions to the Navier-Stokes equations of compressible viscous fluids. *Indiana Univ. Math. J.*, 20:611–631, 2011.

[35] M. Galassi. GNU scientific library reference manual. http://www.gnu.org/software/gsl/, 2018.

[36] N. Gerhard. *An adaptive multiresolution discontinuous Galerkin Scheme for conservation laws*. PhD thesis, RWTH Aachen University, 2017.

[37] N. Gerhard, F. Iacono, G. May, S. Müller, and R. Schäfer. A high-order discontinuous Galerkin discretization with multigrid-based grid adaptation for compressible flows. *J. Sci. Comput.*, 62:25–52, 2015.

[38] N. Gerhard, S. Müller, and A. Sikstel. A wavelet-free approach for multiresolution-based grid adaptation for conservation laws. *Commun. Appl. Math. Comput.*, 4:108–142, 2022.

[39] N. Gerhard and S. Müller. Adaptive multiresolution discontinuous Galerkin schemes for conservation laws: multi-dimensional case. *Comput. Appl. Math.*, 35:321–349, 2016.

[40] J. Giesselmann and A. Dedner. A posteriori analysis of fully discrete method of lines discontinuous Galerkin schemes for systems of conservation laws. *SIAM J. Numer. Anal.*, 54:3523–3549, 2016.

[41] J. Giesselmann and H. Joshi. Model adaptation of balance laws based on a posteriori error estimates and surrogate fluxes. In *Finite volumes for complex applications IX – methods, theoretical aspects, examples. FVCA 9*, pages 375–383. Cham: Springer, 2020.

[42] J. Giesselmann, C. Makridakis, and T. Pryer. A posteriori analysis of discontinuous Galerkin schemes for systems of hyberbolic conservation laws. *SIAM J. Numer. Anal.*, 53:1280–1303, 2015.

[43] J. Giesselmann, F. Meyer, and C. Rohde. Error control for statistical solutions of hyperbolic systems of conservation laws. *Calcolo*, 58:23–29, 2021.

[44] J. Giesselmann and T. Pryer. A posteriori analysis for dynamic model adaptation problems in convection dominated problems. *Math. Models Methods Appl. Sci.*, 27:2381–2432, 2017.

[45] V. Giovangigli. *Multicomponent flow modeling*. Springer, 1999.

[46] P. Gnoffo, R. Gupta, and J. Shinn. Conservation equations and physical models for hypersonic air flows in thermal and chemical nonequilibrium. *NASA TP*, 1989.

[47] E. Godlewski and P. A. Raviart. *Numerical Approximation of Hyperbolic Systems of Conservation Laws*. Springer, 1995.

[48] E. Godlewski and P.-A. Raviart. The numerical interface coupling of nonlinear hyperbolic systems of conservation laws. I. the scalar case. *Numer. Math.*, 97:81–130, 2004.

[49] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Rev.*, 1:89–112, 2001.

[50] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability preserving high-order time discretization methods. *SIAM Rev.*, 43:89–112, 2001.

[51] A. Gouasmi, K. Duriaisamy, S. Murman, and E. Tadmor. A minimum entropy principle in the compressible multicomponent Euler equations. *ESAIM Math. Model. Numer. Anal.*, 54:373–389, 2020.

[52] M. Hantke and S. Müller. Analysis and simulation of a new multi-component two-phase flow model with phase transitions and chemical reactions. *Quart. Appl. Math.*, 76:253–287, 2018.

[53] A. Harten, P. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Rev.*, 25:35–61, 1983.

[54] R. Hartmann and P. Houston. Adaptive Discontinuous Galerkin finite element methods for non-linear hyperbolic conservation laws. *SIAM J. Sci.Comput.*, 24:79–1004, 2002.

[55] M. Herty, S. Müller, and A. Sikstel. Coupling of two hyperbolic systems by solving half-Riemann problems. *Math. Ind.*, 34:285–302, 2021.

[56] J. Hesthaven. *Nodal Discontinuous Galerkin Methods*. Springer, 2008.

[57] N. Hovhannisyan, S. Müller, and R. Schäfer. Adaptive multiresolution discontinuous Galerkin schemes for conservation laws. *Math. Comp.*, 285:113–151, 2014.

[58] E. Isaacson and H. B. Keller. *Analysis of Numerical methods*. Dover, 1966.

[59] K. Kuo. *Principles of Combustion*. John Wiley, 1986.

[60] C. Lattanzio and A. Tzavaras. Structural properties of stress relaxation and convergence from viscoelasticity to polyconvex elastodynamics. *Arch. Ration. Mech. Anal.*, 180:449–492, 2006.

[61] C. Lattanzio and A. Tzavaras. Relative entropy in diffusive relaxation. *SIAM J.Math.Anal*, 45:1563–1584, 2013.

[62] R. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser Verlag, 1999.

[63] D. Lide. *Handbook of Chemisty and Physics*. CRC, 2010.

[64] J. Magiera, D. Ray, J. Hesthaven, and C. Rohde. Constraint-aware neural networks for Riemann problems. *J. Comput. Phys.*, 409, 2020.

[65] H. Mathis, C. Cancès, E. Godlewski, and N. Seguin. Dynamic model adaptation for multiscale simulation of hyperbolic systems with relaxation. *J. Sci. Comput.*, 63:820–861, 2015.

[66] H. Mathis and N. Seguin. Model adaptation for hyperbolic systems with relaxation. *Finite Volumes for Complex Applications*, VI:673–682, 2011.

[67] A. D. McNaught and A. Wilkinson. *IUPAC compendium of chemical terminology*. RSC, 1997.

[68] I. Mills, T. Cvitas, K. Homann, N. Kallay, and K. Kuchitsu. *Quantities, Units and Symbols in Physical Chemistry*. RSC, 1993.

[69] A. Miroshnikov and Konstantina Trivisa. Relative entropy in hyperbolic relaxation for balance laws. *Commun. Math. Sci*, 12:1017–1043, 2014.

[70] S. Müller and A. Sikstel. Multiwave. https://www.igpm.rwth-aachen.de/forschung/multiwave, Institut für Geomtrie und Praktische Mathematik, RWTH Aachen, 2018.

[71] P. Márquez-Neila, M. Salzmann, and P. Fua. Imposing hard constraints on deep networks: Promises and limitations. *arXiv:1706.02025*, 2017.

[72] I. Müller and W. H. Müller. *Fundamentals of Thermodynamics and applications*. Springer, 2009.

[73] S. Müller. *Erweiterung von ENO-Verfahren auf zwei Raumdimensionen und Anwendung auf hypersonische Staupunktprobleme*. PhD thesis, RWTH Aachen University, 1993.

[74] S. Müller, M. Hantke, and P. Richter. Closure conditions for non-equilibrium multi-component models. *Contin. Mech. Thermodyn.*, 4:1157–1189, 2016.

[75] R. W. Nicholas Nethercote and J. Fitzhardinge. Building workload characterization tools with Valgrind. *IEEE*, 2006.

[76] L. Pareschi and G. Russo. Implicit-explicit Runge-Kutta schemes for stiff systems of differential equations. *Adv. Theory Comput. Math.*, 3, 2001.

[77] L. Pareschi and G. Russo. Implicit-Explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation. *J. Sci. Comput.*, 25:129–155, 2007.

[78] C. Park. Convergence of computation of chemical reacting flows. *Progress in Astronautics and Aeronautics*, pages 478–513, 2016.

[79] D. A. D. Pietro and A. Ern. *Mathematical aspects of Discontinuous Galerkin Methods*. Springer, 2012.

[80] J. Piraux and B. Lombard. A new interface method for hyperbolic problems with discontinuous coefficients: one-dimensional acoustic example. *J. Comput. Phys.*, 168:227–248, 2001.

[81] J. Qiu and C.-W. Shu. A comparison of troubled-cell indicators for Runge-Kutta discontinuous Galerkin methods using weighted essentially nonoscillatory limiters. *SIAM J. Sci. Comput.*, 27:955–1013, 2005.

[82] S. Raoux and M. Wuttig. *Phase Change Materials*. Springer, 2009.

[83] J. Remacle, J. Flaherty, and M. Shephard. An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible fow problems. *SIAM Rev.*, 45:53–72, 2003.

[84] P. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.*, 135:249–258, 1997.

[85] S. Ruuth. Global optimization of explicit strong-stability-preserving Runge-Kutta methods. *Math. Comp.*, 75:183–207, 2006.

[86] J. Seinfeld and A. Pandnis. *Atmospheric Chemistry and Physics*. Wiley, 1998.

[87] D. Serre. Multidimensional shock interaction for a chaplygin gas. *Arch. Ration. Mech. Anal.*, 191:539–577, 2009.

[88] A. Sikstel. *Analysis and Numerical Methods for Coupled Hyperbolic Conservation Laws*. PhD thesis, RWTH Aachen University, 2021.

[89] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer, 1975.

[90] F. Thein. *Results for Two Phase Flows with Phase Transition*. PhD thesis, Universität Magdeburg, 2019.

[91] E. Toro. *Riemann solvers and numerical methods for fluid dynamics*. Springer, 2009.

[92] M. Torrilhon and N. Sarna. Hierarchical Boltzmann simulations and model error estimation. *J. Comput. Phys.*, 342:66–84, 2017.

[93] A. Tzavaras. Materials with internal variables and relaxation to conservation laws. *Arch. Ration. Mech. Anal.*, 146:129–155, 2000.

[94] A. Tzavaras. Relative entropy in hyperbolic relaxation. *Commun. Math. Sci*, 3:119–132, 2005.

[95] J. Utzmann, T. Schwartzkopff, M. Dumbser, and C. D. Munz. Heterogeneous domain decomposition for computational aeroacoustics. *AIAA J.*, 44:2231–2250, 2006.

[96] C. Westbrook. Chemical kinetics on hydrocarbon oxidation in gaseous detonations. *Combust. Flame*, 46:191–210, 1982.

[97] W.-A. Yong. Entropy and global existence for hyperbolic balance laws. *Arch. Ration. Mech. Anal.*, 172:247–266, 2004.

[98] Y. Zeng. Gas dynamics in thermal nonequilibrium and general hyperbolic systems with relaxation. *Arch. Rational Mech. Anal.*, 150:225–279, 1999.

[99] Q. Zhang and C.-W. Shu. Error estimates to smooth solutions of Runge-Kutta discontinuous Galerkin methods for scalar conservation laws. *SIAM J. Numer. Anal.*, 42:641–666, 2004.

[100] Q. Zhang and C.-W. Shu. Error estimates to smooth solutions of Runge-Kutta discontinuous Galerkin method for symmetrizable systems of conservation laws. *SIAM J. Numer. Anal.*, 44:1703–1720, 2006.

Wissenschaftlicher Werdegang

Hrishikesh Joshi

2022	Promotion Mathematik
2018-2022	Wissenschaftlicher Mitarbeiter an der TU Darmstadt
2017-2018	Wissenschaftlicher Mitarbeiter an der Universitität Stuttgart
2015-2017	Studium der Maschinenbau an der TU Delft, Abschluss M.Sc.
2011-2015	Studium der Maschinenbau an der University of Pune, Abschluss B.E.