STABILIZED FINITE ELEMENT SIMULATIONS OF MULTISPECIES INVISCID
HYPERSONIC FLOWS IN THERMOCHEMICAL NONEQUILIBRIUM

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF APPLIED MATHEMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SÜLEYMAN CENGİZCİ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
SCIENTIFIC COMPUTING

MARCH 2022

Approval of the thesis:

STABILIZED FINITE ELEMENT SIMULATIONS OF MULTISPECIES INVISCID HYPERSONIC FLOWS IN THERMOCHEMICAL NONEQUILIBRIUM

submitted by **SÜLEYMAN CENGİZCİ** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Scientific Computing Department, Middle East Technical University** by,

Prof. Dr. A. Sevtap Kestel
Dean, Graduate School of **Applied Mathematics** _____

Assoc. Prof. Dr. Önder Türk
Head of Department, **Scientific Computing** _____

Prof. Dr. Ömür Uğur
Supervisor, **Scientific Computing, IAM, METU** _____

Prof. Dr. Tayfun E. Tezduyar
Co-supervisor, **Mechanical Engineering, Rice University**
Faculty of Science and Engineering, Waseda University _____

Examining Committee Members:

Assoc. Prof. Dr. Murat Uzunca
Department of Mathematics, Sinop University _____

Prof. Dr. Ömür Uğur
Scientific Computing, IAM, METU _____

Assoc. Prof. Dr. Hamdullah Yücel
Scientific Computing, IAM, METU _____

Assoc. Prof. Dr. Mehmet T. Atay
Engineering Sciences, Abdullah Gül University _____

Assoc. Prof. Dr. Önder Türk
Scientific Computing, IAM, METU _____

Date: _____

iv

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: SÜLEYMAN CENGİZCİ

Signature   :

ABSTRACT

STABILIZED FINITE ELEMENT SIMULATIONS OF MULTISPECIES INVISCID
HYPERSONIC FLOWS IN THERMOCHEMICAL NONEQUILIBRIUM

CENGİZCİ, SÜLEYMAN

Ph.D., Department of Scientific Computing

Supervisor : Prof. Dr. Ömür Uğur

Co-Supervisor : Prof. Dr. Tayfun E. Tezduyar

March 2022, 187 pages

For both military and civil aviation purposes, rockets, missiles, and spacecraft moving at hypersonic speeds are being utilized in recent years. While these vehicles move at speeds five times the speed of sound or more, they experience many extreme physical and chemical conditions during their flight. Because of molecular friction, such high velocities cause very high temperatures, and these high temperatures result in the excitation of the components of the gas mixture in which the vehicle moves. This situation causes various thermochemical interactions in the flow field and affects the dynamics of the flight. These interactions need to be examined accurately, for both the flight safety and the vehicle reaching the right target at the right time.

Wind tunnel experiments are both costly and insufficient to regenerate the high temperatures and shock interactions of hypersonic flights. These wind tunnel setups can also take a long time to design, test, and finally obtain the experimental data with. Therefore, computational fluid dynamics (CFD) tools are essential in analyzing the flight dynamics of hypersonic vehicles and designing them for such high speeds. Classical discretization methods need to be supplemented with stabilization and shock-capturing techniques since they suffer from spurious oscillations in simulating such high-speed flows.

In this thesis, hypersonic flows in thermochemical nonequilibrium are computationally

studied. To this end, hypersonic flows of a five-species (O, N, NO, O_2, N_2) gas mixture around a cylinder are examined with a 17-reaction chemical model. The gas particles may be in different energy modes in hypersonic regimes due to the high temperatures: translational, rotational, vibrational, and electron-electronic. Since they have the similar time scales to reach equilibrium, the translational and rotational energy modes can be represented by one temperature, and the vibrational and electron-electronic energy modes by another. Therefore, a two-temperature chemical kinetic model is adopted.

In the computations, the compressible-flow Streamline-Upwind/Petrov–Galerkin method is employed to stabilize the finite element formulation. The stabilized formulation is supplemented with the YZβ shock-capturing to obtain good solution profiles at shocks. The nonlinear system of equations resulting from the space and time discretizations is solved with the Newton–Raphson nonlinear iterative process and ILU-preconditioned generalized minimal residual (GMRES) iterative search technique. The solvers are developed in the FEniCS environment.

ÖZ

TERMOKİMYASAL DENGESİZLİKTEKİ ÇOK BİLEŞENLİ VİSKOZİTESİZ HİPERSONİK AKIŞLARIN STABİLİZE EDİLMİŞ SONLU ELEMAN BENZETİMLERİ

CENGİZCİ, SÜLEYMAN

Doktora, Bilimsel Hesaplama Bölümü

Tez Yöneticisi   : Prof. Dr. Ömür Uğur

Ortak Tez Yöneticisi   : Prof. Dr. Tayfun E. Tezduyar

Hipersonik hızlarda hareket eden roketler, füzeler ve uzay araçları son yıllarda hem askeri hem de sivil havacılık amaçlarıyla kullanılmaktadırlar. Bu araçlar, ses hızının beş katı veya üzerindeki hızlarda hareket ederlerken, birçok şiddetli fiziksel ve kimyasal etkileşim tecrübe ederler. Bu tür yüksek hızlar, moleküler sürtünmeler nedeniyle çok yüksek sıcaklıklara, bu yüksek sıcaklıklar ise aracın içerisinde hareket ettiği gaz karışımının bileşenlerinin uyarılmasına neden olurlar. Bu durum, akış ortamında termokimyasal etkileşimlere neden olur ve aracın uçuş dinamiklerini etkiler. Hem uçuşun güvenliği hem de aracın doğru zamanda doğru hedefe ulaşabilmesi için bu etkileşimlerin hassas bir şekilde incelenmesi gerekmektedir.

Rüzgar tüneli testleri hipersonik uçuşların yüksek sıcaklık ve şok etkileşimlerini oluşturmada hem yetersiz kalmaktadırlar hem de çok maliyetlidirler. Dahası, bu deneysel kurulumların tasarlanması, test edilmesi, ve nihayet deneysel verilerin elde edilmesi oldukça uzun zaman alabilmektedir. Bu nedenle, hipersonik araçların uçuş dinamiklerinin analizinde ve bu tür yüksek hızlara uygun araçların dizaynında hesaplamalı akışkanlar dinamiğinin (HAD) araçları büyük öneme sahiptirler. Standart ayrıklaştırma metotlarının, bu tür yüksek hızlı akışların simülasyonlarında sahte salınımlar ürettikleri için, stabilizasyon ve şok-yakalama teknikleri ile desteklenmeleri gerekmektedir.

Bu tezde, termokimyasal dengesizlikteki hipersonik akışlar hesaplamalı olarak incelenmektedir. Bu amaçla, 5-bileşenli (O, N, NO, O_2, N_2) bir gaz karışımının bir silindir etrafındaki hipersonik akışı 17-reaksiyonlu bir kimyasal modelle ele alınmaktadır. Gaz partikülleri hipersonik rejimlerdeki yüksek sıcaklıklar nedeni ile farklı enerji modlarında bulunabilirler: çizgisel, dönüşsel, titreşimsel, ve elektron-elektronik. Denge durumuna ulaşmada benzer zaman ölçeklerine sahip olduklarından, çizgisel ve dönüşsel enerji modları aynı sıcaklıkla, titreşimsel ve elektron-elektronik enerji modları ise bir diğer sıcaklıkla ifade edilebilirler. Bu nedenle, iki sıcaklıklı bir kimyasal kinetik model benimsenmektedir.

Hesaplamalarda, sonlu elemanlar formülasyonunu stabilize etmek için sıkıştırılabilir-akış Streamline-Upwind/Petrov–Galerkin metodu kullanılmaktadır. Stabilize edilmiş formülasyon, şoklarda daha iyi çözüm profilleri elde etme amacıyla, YZβ şok-yakalama tekniği ile desteklenmektedir. Uzay ve zaman ayrıklaştırmaları sonucu elde edilen doğrusal olmayan denklem sistemleri Newton–Raphson lineer olmayan yinelemeli çözüm metodu ve ILU yöntemi ile önkoşullandırılmış genelleştirilmiş minimal kalıntı (GMRES) arama tekniği kullanılarak çözülmektedir. Çözücü kodlar FEniCS ortamında geliştirilmektedir.

Anahtar Kelimeler: Viskozitesiz akış, Hipersonik akış, Sonlu elemanlar, Sıkıştırılabilir-akış SUPG, Şok-yakalama, Termokimyasal dengesizlik

There is a lot in this thesis: sadness, happiness, sleepiness, perseverance, despair, hope, sometimes loneliness and sometimes crowd, and sometimes tears and sometimes laughter...

I dedicate this thesis to my family,

ACKNOWLEDGMENTS

Dr. Hao was always very helpful and supportive.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AIAA American Institute of Aeronautics and Astronautics

ASME American Society of Mechanical Engineers

CAD computer-aided design

CFD computational fluid dynamics

CFL Courant–Friedrichs–Lewy

DSMC direct simulation Monte Carlo

FDM finite difference method

FEA finite element analysis

FEM finite element method

FVM finite volume method

GFEM Galerkin finite element method

GMRES generalized minimal residual

IGA isogeometric analysis

ILU incomplete lower-upper

LES large-eddy simulation

MINRES minimal residual

NASA National Aeronautics and Space Administration

N–R Newton–Raphson

N–S Navier–Stokes

PDE partial differential equation

SC shock-capturing

SUPG Streamline-Upwind/Petrov–Galerkin

SUPG-SC Streamline-Upwind/Petrov–Galerkin supplemented with shock-capturing

SUPG-YZβ Streamline-Upwind/Petrov–Galerkin supplemented with YZβ shock-capturing

TPS thermal protection system

CHAPTER 1

INTRODUCTION

Flows that are five times or above the speed of sound are called *hypersonic*. The term "*hypersonic*" first appeared in the literature in the mid-1940s and has been actively studied since the late 1950s. Today, advances in technology, world's growing population and the consequent depletion of raw materials have made the hypersonic research much more vital. For example, samples from other planets that scientists will analyze in the search for a sign of extraterrestrial life should arrive quickly enough for them to conduct their research effectively. If there will be space tourism, the capsule that will carry the passengers should be safe and fast enough so that the passengers can return to their lives on the Earth within a reasonable time. Besides, in today's world, many countries are developing their own hypersonic missiles and fighter spacecraft, making tremendous investments. These and many other factors have increased the interest in hypersonic research even more in recent years.

Hypersonic vehicles moving at such high speeds experience severe chemical and physical interactions during their missions. The detection, understanding of their causes, and accurate computation of these interactions are crucial both for the flight safety and for the vehicle to reach the right destination at the right time. Therefore, this chapter aims to provide fundamental knowledge about fluid dynamics with a special emphasis on hypersonic aerodynamics.

Section 1.1, provides a brief introduction to the relevant fluid dynamics concepts. Various flow regimes and the molecular structure of Earth's atmosphere at certain altitudes, real gas effects, catalytic and ablative effects, and finally, the shock wave and the blackout phenomenon are discussed in Section 1.2. Section 1.3 examines the

shapes of hypersonic configurations: the benefits they provide and the drawbacks they cause. In Section 1.4, the classical tools used in computational fluid dynamics are discussed briefly. Finally, the objectives and contributions of the study, as well as the thesis layout, are presented in Section 1.5 and in Section 1.6, respectively.

1.1 Background

In fluid dynamics, flows can be classified various ways (based on their variation based on space/time), such as viscous or inviscid, compressible or incompressible, laminar or turbulent, etc. These different types of flows may result from the fluid's thermodynamic properties, the conditions of the environment in which the fluid flows, or both.

The *viscosity* is a measure of the intermolecular frictions of a fluid. In general, the frequency of intermolecular collisions increases with temperature, and thus the viscosity of gases increases with temperature. Viscous flows are the flows in which the fluid's intermolecular forces are in action, while these forces are neglected for inviscid flows. Viscous flows are governed by the Navier–Stokes (N–S) equations, while inviscid flows by the Euler equations. Thus, the Euler equations can be obtained by ignoring the viscous and thermal conductivity terms in the N–S equations.

There are various viscosity models for different phases of substances that depend on temperature or pressure. For gases, the most commonly used model is *Sutherland's temperature-dynamic viscosity formula* [172]:

$$\mu_{gas} = \mu_0 \frac{T_0 + S}{T + S} \left(\frac{T}{T_0} \right)^{\frac{3}{2}}, \tag{1.1}$$

where T is the absolute temperature, μ_0 is the reference viscosity at a reference temperature T_0, and S is the Sutherland's constant. For moderate temperatures and pressures, the constants appearing in Eq. (1.1) are given approximately as $\mu_0 = 1.7894 \times 10^{-5}$ kg/(m·s), $T_0 = 273.11$ K, and $S = 110.56$ K. However, Sutherland's formula is only valid for temperatures between 100 K and $1,500$ K, which is not sufficient to handle extremely high temperatures arising in hypersonic flows. Therefore, more suitable models are investigated for high-temperature flows in Chapter 2.

2

If the fluid density remains nearly constant throughout the flow, then the flow is called *incompressible* or *divergence-free*; otherwise, the flow is *compressible*. Compressible flows are frequently called *variable-density flows*. Compressibility effects are critical in this thesis because the flow simulations performed are entirely based on gas flows that are compressible. Every fluid can be compressed to some extent in real life, so incompressibility is just an assumption. In practice, if the density change during the flow is not more than 5%, then the flow can be assumed to be incompressible [6].

Mach number, M, defined as the ratio of the vehicle's speed to the speed of sound (acoustic speed), is used for determining the magnitude of the level of compressibility effects [6]. Since the sound consists of vibrations propagating as pressure waves, the speed of sound can be used as an indicator of speed of transmission of small disturbances in the air.

The speed of sound depends on the temperature and the environment it travels through. For dry air, it is given as

$$c_{\text{sound}} = \sqrt{\gamma R T}, \tag{1.2}$$

where γ is the specific heat ratio, given as $\gamma = 1.4$, and R is the ideal-gas constant. Here, T is the absolute temperature, which is given in Kelvin (K) as

$$T = 273.15 + {}^\circ\text{C}. \tag{1.3}$$

In Eq. (1.3), the term ${}^\circ$C represents the temperature in Celsius with the ideal-gas constant for air, given as $R = 287.05$ J/(kg·K), the speed of sound through dry air at 20°C is approximately 343.23 m/s [205].

If the Mach number is lower than one ($M < 1.0$), then the compressibility effects can be ignored, and the flow is called *subsonic*. Around $M = 1.0$, the compressibility effects are quite important, and the flow is said to be *transonic*. For the values $1.0 < M < 5.0$, the flow is called *supersonic*. The heating rates are very high at these speeds, and the vehicle surface generates shock waves.

As mentioned before, flows with speeds of five times or higher than the speed of sound are classified as *hypersonic*. However, it would be wrong to say that a flow with a speed just below Mach 5.0, e.g., at $M = 4.99$, does not have a hypersonic character. Similarly, it is also not correct to say that a flow occurring at just above Mach 5.0,

for example, at $M = 5.01$, has a complete hypersonic character. Instead, it is more sensible to consider a flow as hypersonic if it experiences the effects explained in the following section. Table 1.1 summarizes flow characteristics based on various Mach numbers.

Table 1.1: Flow classification based on the Mach number.

Vehicle Speed	Classification	Flow Characteristic
$M < 1.0$	Subsonic	Compressibility effects can be ignored
$M \approx 1.0$	Transonic	Dangerous stress effects on vehicle's body
$1.0 < M < 5.0$	Supersonic	Compressibility effects must be considered
$M > 5.0$	Hypersonic	Thermochemical nonequilibrium effects
$M \approx 25.0$	Re-entry speeds	Ionized gas (plasma) effects

In Figure 1.1, the propagation of the pressure waves according to various Mach numbers is shown. The sound (pressure) waves propagate separately as the vehicle moves at speeds below the speed of sound. However, when the Mach number is greater than one, it is observed that the sound waves overlap and form shock waves.
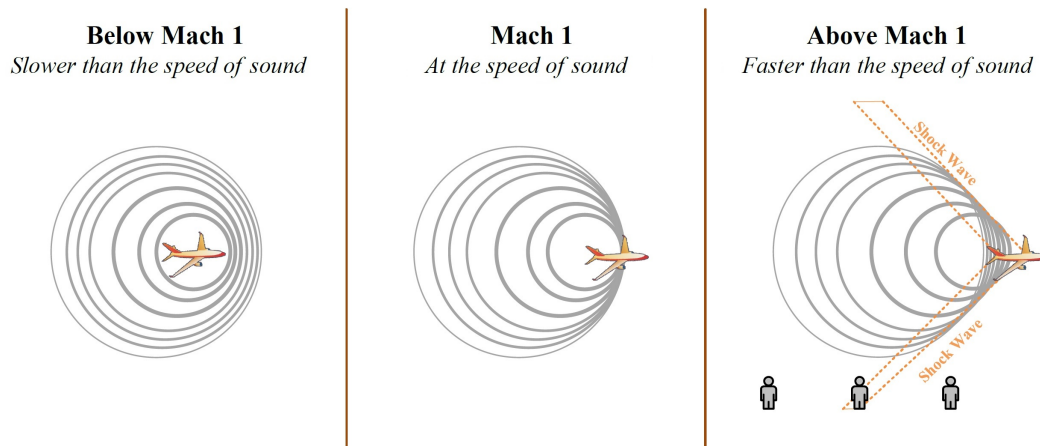
Figure 1.1: Sonic boom diagram based on the Mach number [40].

1.2 Characteristics of Hypersonic Flows

As a rocket launched from the Earth to the Moon rises above the atmosphere, the flow properties such as the density, temperature, pressure, etc., and consequently, the dynamics of the flight change. Besides that, the flow becomes chemically and thermally nonequilibrium due to the high temperatures caused by high speeds. The

surface material of the rocket can even be affected by the high temperatures and reacts with the air mixture constituting the flow field. Higher temperatures can cause the ionization process to begin and the surface of the rocket to be covered with a plasma layer, causing communication losses called *blackout*. These interactions are discussed in detail in the following sections.

1.2.1   Low-Density Effects

Vehicles moving at hypersonic speeds may encounter different flow regimes that correspond to *continuum flow*, *slip flow*, *transitional flow*, and *free-molecular flow*. The dimensionless *Knudsen number*, Kn, defined as the ratio of the mean free path, λ, to the characteristic length, L,

$$Kn = \frac{\lambda}{L},\tag{1.4}$$

is used to determine the characteristic of the flow regime [22]. If $Kn < 0.01$, the flow regime is classified as *continuum*, as *slip flow* if $0.01 < Kn < 0.1$, as *transitional flow* if $0.1 < Kn < 10.0$, and as *free-molecular flow* if $Kn > 10.0$.

At the altitudes above 120 km in Earth's atmosphere, the Knudsen number takes a value above $Kn = 10.0$, corresponding to free-molecular flows. The collisions of gas particles between the gas particles rebounding from the vehicle's surface and the incoming ones are so rare that they can be ignored. Therefore, it is not necessary to talk about any shock wave formation within this region [165].

At the altitudes between 90 km and 120 km, the Knudsen number ranges between 0.1 and 10.0. The collisions of gas particles in this regime can generate bow shocks, and the classical N–S equations suffer from capturing the shocks. Therefore, higher-order equations, such as the Boltzmann and Burnett equations, are required for simulating such flows [165].

Between the altitudes of 80 km and 90 km, the Knudsen number takes a value between 0.01 and 0.1, corresponding to a slip flow. Flows in this regime can be assumed to be a continuum except for the inner regions of shock layers. For simulating flows in this regime, the direct simulation Monte Carlo (DSMC) methods are suitable [53].

Below the altitudes of 80 km, which is the interest of this thesis, abrupt decreases are observed in the velocity of the vehicle due to the increasing density and consequent increases in the friction rates. Although the vehicles experience severe overheating rates in this region, we take advantage of the fact that a cubic centimeter of air mixture contains approximately 2.652×10^{19} molecules and behaves like a continuum under normal room conditions, ensuring the validity of differential models. In this regime, the vast majority of the kinetic energy is transformed into the thermal energy that may cause extremely high temperatures up to $50,000$ K [165]. During NASA's Apollo Project (1961–1975), plenty of data was collected that confirms this fact [143].

Earth's atmosphere consists mainly of nitrogen (N_2) and oxygen (O_2) with the approximate percentages of 78% and 21%, respectively. Therefore, a mathematical model considering the particles O, N, NO, O_2, and N_2, also known as the *five-species air mixture model*, can be used for simulating air flows [81, 135, 153]. For entries into various media or environments, different cases should be considered. This five-species model, for example, would not be appropriate for a Mars mission vehicle because the Martian atmosphere consists of approximately 97% carbon dioxide (CO_2) and 3% nitrogen (N_2) [206].

Aside from the studies on Earth's atmosphere, it's not surprising that the majority of the atmospheric-entry research has focused on the Martian atmosphere. In [124], Li and Jiang presented some comparisons of the atmospheres of Earth and Mars. In [26], Braun and Manning, and later in [171], Subrahmanyam and Rasky, reviewed past Mars landed missions presenting corresponding flight data. Wang et al. examined the heating predictions for Martian entry vehicles considering both laminar and turbulent flows in [201]. In [77], Hao et al. studied and compared several transport models for simulating Martian entries. In [31], Carandente et al. reviewed and compared the experimental data and numerical results regarding the entries into the atmosphere of the Titan, which is the Saturn's largest moon, and has a fluid cycle similar to that on Earth. Justus and Braun [98] provide detailed information on the atmospheric entries into the planets Venus, Earth, Mars, Saturn, and its moon Titan.

1.2.2 Real-Gas Effects

The gases consist of particles such as molecules, atoms, and ions in random motion called *translational motion*. These particles interact with each other due to their electronic structure. These interactions are called *intermolecular interactions*. The gases for which these interactions are ignored are called *perfect (ideal) gas*. If a gas mixture is not chemically reactive and the intermolecular forces are small enough to be neglected, then the gas is called *thermally perfect gas*. In addition, if the specific temperatures are constant, then the gas is called *calorically perfect gas*.

In Earth's atmosphere, the calorically perfect gas assumption can be made at temperatures below 800 K. However, it would be wrong to make this assumption for higher temperatures [56]. Since the vibrational excitements of oxygen (O_2) and nitrogen (N_2) cannot be ignored in the atmosphere above temperatures 800 K, the gas mixture turns into a thermally perfect gas. That is, the specific temperatures act as the functions of temperature. When the temperature rises above $2,000$ K, oxygen (O_2) molecules begin to decompose into their atoms, and chemical reactions should be taken into account. At temperatures above $4,000$ K, oxygen (O_2) molecules are completely decomposed into their atoms, and a similar decomposition process begins for nitrogen (N_2) molecules. And, when it reaches temperatures around $9,000$ K, nitrogen (N_2) molecules are completely decomposed into their atoms. As the temperature rises even more, i.e., after $10,000$ K, ionization processes become significant [34]. One can refer to [7] for further details.

Each particle's individual translational energy (e^{trans}) contributes to the *total energy* (e) of the gas mixture. If the particles under consideration are molecules, the *rotational energy* (e^{rot}) and *vibrational energy* (e^{vib}) should also be examined to calculate the total energy. Another source of (internal) energy, called the *electronic energy* (e^{elec}), stems from the movement of electrons. These all different energy modes of a gas mixture constitute the *internal energy* (e^{int}) [107]:

$$e^{int} = e^{trans} + e^{rot} + e^{vib} + e^{elec} + h_0, \tag{1.5}$$

where h_0 is called the *heat of formation* of the mixture, and stands for the energy

stored in chemical bonds. Then, the total energy is given as

$$e = e^{int} + \frac{1}{2}\|\mathbf{u}\|^2, \tag{1.6}$$

where the second term on the right hand side refers to the *kinetic energy* with the velocity vector $\mathbf{u}$. Here, the norm $\|\cdot\|$ stands for the standard Euclidean norm.

If the time scale that the chemical reactions take place is much shorter than the time scale that a fluid (gas) particle moves in the flow, then this situation is called *chemical equilibrium*. The dimensionless Damköhler number (Da) is used to determine whether the flow time scale or chemical reaction time scale is more significant. It is defined as follows [166]:

$$Da = \frac{\text{Fluid motion time scale}}{\text{Chemical reaction time scale}} = \frac{\text{Diffusion time}}{\text{Reaction time}}. \tag{1.7}$$

The gas reaches equilibrium as Da approaches infinity. As Da approaches zero, the gas is assumed to be *frozen*, with no chemical reactions taking place. When the fluid and flow properties result a state between these two extreme states, *chemical nonequilibrium* exists.

Thermal nonequilibrium can be examined concerning the particles' four different energy modes and corresponding temperatures, as illustrated on a diatomic molecule, called the *dumbbell model*, in Figure 1.2. These energy modes are:

1. **Translational energy** results from the Brownian motion of particles; and the corresponding temperature is denoted by T_t,

2. **Rotational energy** results from the rotations of the molecules around their center of mass; and the corresponding temperature is denoted by T_r,

3. **Vibrational energy** results from the periodic motion of atoms in molecules; and the corresponding temperature is denoted by T_v,

4. **Electron-electronic energy** results from the states (energy levels) of electrons; and the corresponding temperature is denoted by T_e.

When a molecule is in *thermal equilibrium*, a single temperature, T, can be used to model all energy modes. If the molecule is not in thermal equilibrium, then each
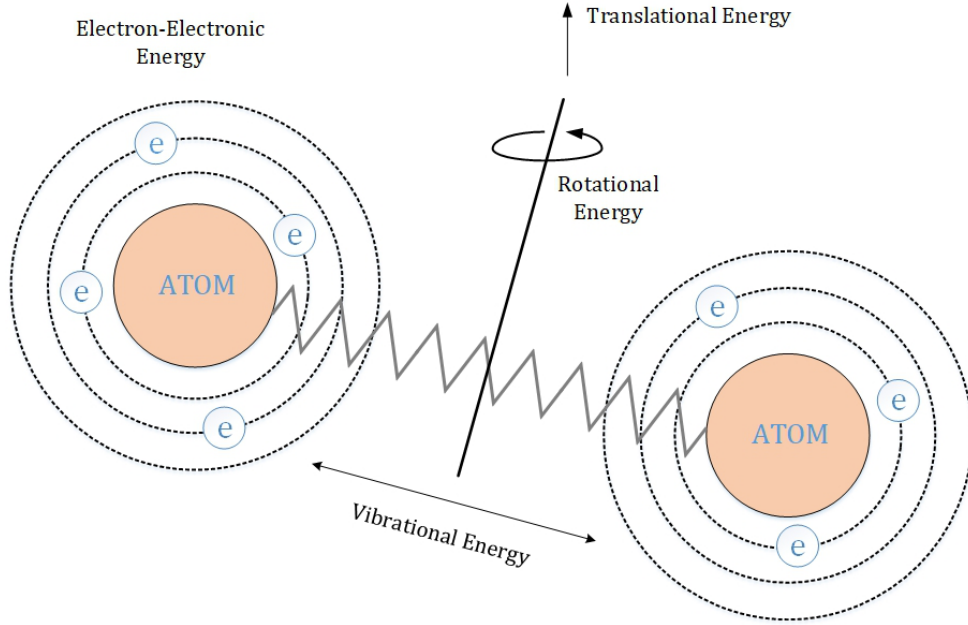
Figure 1.2: Different energy modes for a diatomic molecule [107].

mode should be represented by its own temperature, i.e., by T_t, T_r, T_v, and T_e. As reported in [129], the translational and rotational energy modes reach equilibrium very rapidly within 5–10 collisions, and therefore, they can be represented by the same temperature [106]:

$$T = T_t = T_r. \tag{1.8}$$

The vibrational mode requires many more collisions, on the order of $20,000$, to reach equilibrium [34]. One common approach given in [146,207] is to model the vibrational and electron-electronic temperatures by the same temperature as follows:

$$T_V = T_v = T_e. \tag{1.9}$$

As a result, in this thesis, a two-temperature model with the translational-rotational temperature, T, and vibrational-electron-electronic temperature, T_V, is used. The details on computing T_V are presented in Section 4.7.

1.2.3 Catalytic and Ablation Effects

Although we enforce neither *catalytic* nor *ablative* wall conditions in simulations performed in this study, catalytic boundary conditions and ablation effects should

be mentioned briefly because they are critical in hypersonic flows. As mentioned in Section 1.2.2, as a result of extremely high temperatures generated in the immediate vicinity of high-speed vehicles, some chemical reactions such as dissociation, recombination, etc. may occur. Besides the intermolecular reactions of gas particles, their interaction with the vehicle's surface material should also be considered. These reactions can be classified into *catalytic wall* and *ablative wall* interactions.

Catalytic wall interactions are those in which the vehicle's surface acts as a catalyst, increasing the rate of the reaction without changing or being consumed. The surface only helps the gas particles to have chemical reactions (generally recombination). On the other hand, the ablation process is pretty much complex than the catalytic one because it is the case in which the vehicle surface interacts with the free-stream gas particles to form new particles. The thermal protection systems (TPS) of spacecraft undergo shape changes and mass loss as a result of this process. One can find more on hypersonic TPSs in [38].

This section is actually a specialization of material science, so the details are beyond the scope of this study. The interested readers are referred to the sudies [100] by Kasen, and [194] by Tissera for more on wall chemistry.

1.2.4 Shock Wave Phenomena

In high-speed flows, shock waves are very common, and understanding the flow properties across the shock waves is critical for hypersonic computations. When an air vehicle travels faster than the speed of sound through the atmosphere, the sound waves superpose, causing a rapid increase in pressure, which is known as a *shock wave*. A *normal shock wave* occurs when the shock wave occurs perpendicular to the flow direction. As can be seen in Figure 1.3, the downstream velocity behind the shock decreases while the other flow properties such as the pressure (p), density (ρ), temperature (T), and entropy (s) increase.

The shock waves that are inclined to the flow direction, depending on the shape and velocity of the vehicle, are known as *oblique shock waves*. Oblique shocks are generated by the nose and leading edges of the wings and tail. Normal shock waves
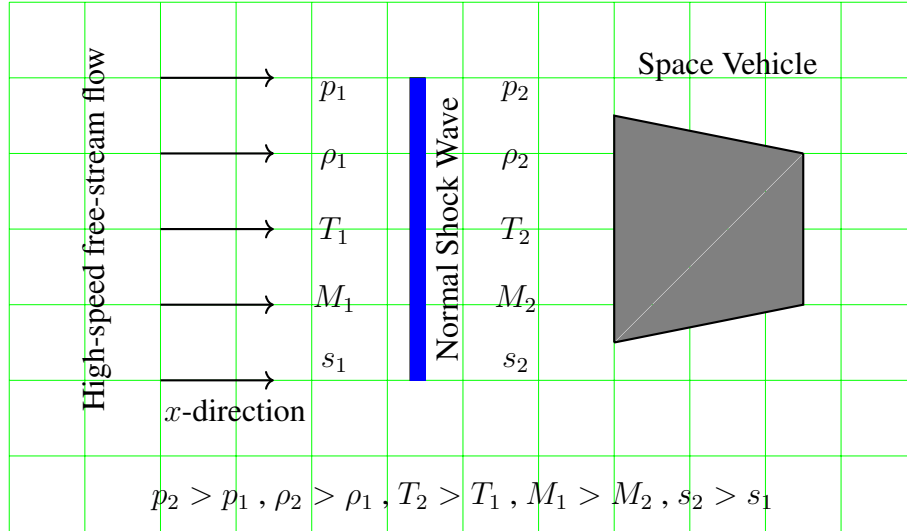
Figure 1.3: Normal shock wave and changes in the flow field.

can be considered as a special case of oblique shocks. An illustration of an oblique shock wave is given in Figure 1.4. The angles β and θ are called *shock angle* and *deflection angle*, respectively. The downstream Mach number, M_2, may be subsonic, sonic, and super or hypersonic depending on the deflection angle, θ, and the upstream (free-stream) Mach number, M_1. For calorically perfect gases, an important equation for calculating shock angles, called the θ–β–M equation, is given as follows [6]:

$$\tan(\theta) = 2\cot(\beta)\frac{M^2\sin^2(\beta) - 1}{M^2(\gamma + \cos(2\beta)) + 2}, \tag{1.10}$$

where γ is the ratio of specific heats which is typically set $\gamma = 1/4$ for standard conditions. For the derivation of Eq. (1.10), the interested reader refers to [6] by Anderson.

One can point out from Figure 1.5 that, as the free-stream Mach number, M_1, increases, the shock angle, β, decreases. The maximum value that the deflection angle, θ, can take in this range is denoted by $\theta_{\max}$. If the half wedge angle is greater than $\theta_{\max}$, i.e., $\delta > \theta_{\max}$, then a curved shock, called *detached oblique shock* or simply *bow shock*, is observed. The bow shock phenomena is illustrated in Figure 1.6.

1.2.5 Radio Blackout

Another important issue to mention caused by high temperatures is the radio communication blackout. High temperatures create a weakly ionized plasma layer around the

11
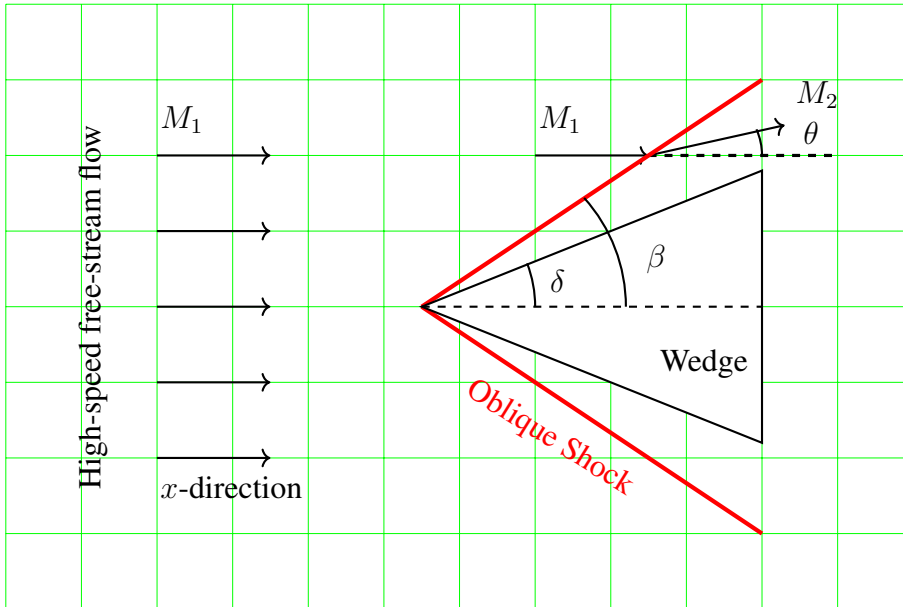
Figure 1.4: Oblique shock front a wedge.
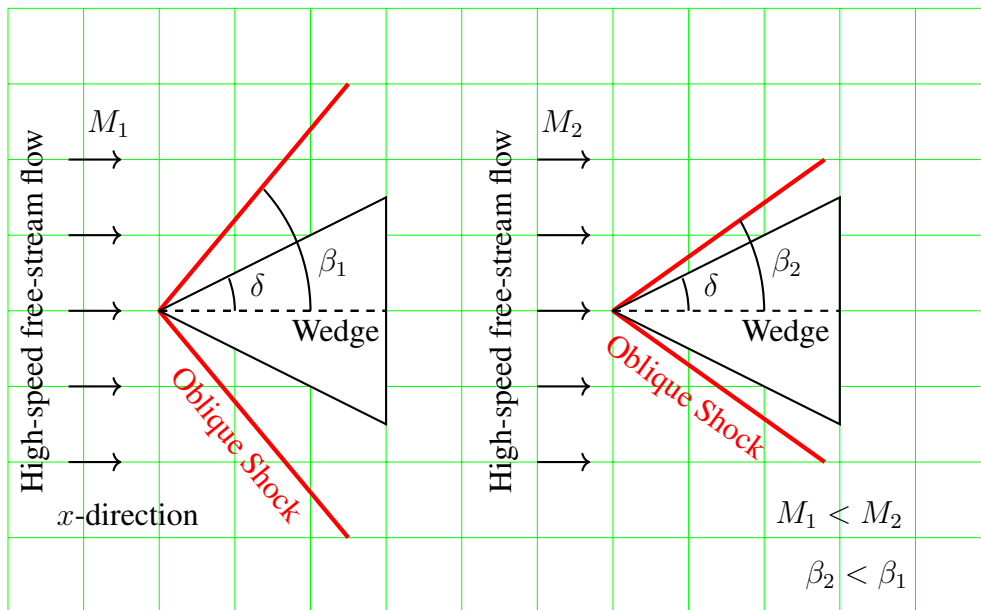
Figure 1.5: Effect of the free-stream Mach number on oblique shocks.

vehicle. The free electrons caused by the ionization occurring in the plasma layer may prevent the propagation of radio frequency electromagnetic waves. In this case, the plasma layer reflects radio waves, causing telemetry attenuation or non-transmission of radio waves. This phenomenon takes about 4–10 minutes in the process of entering Earth's atmosphere, while it takes about 30 minutes as entering the atmosphere of Jupiter [170]. Gilmann et al. studied the blackout phenomenon in detail in a NASA
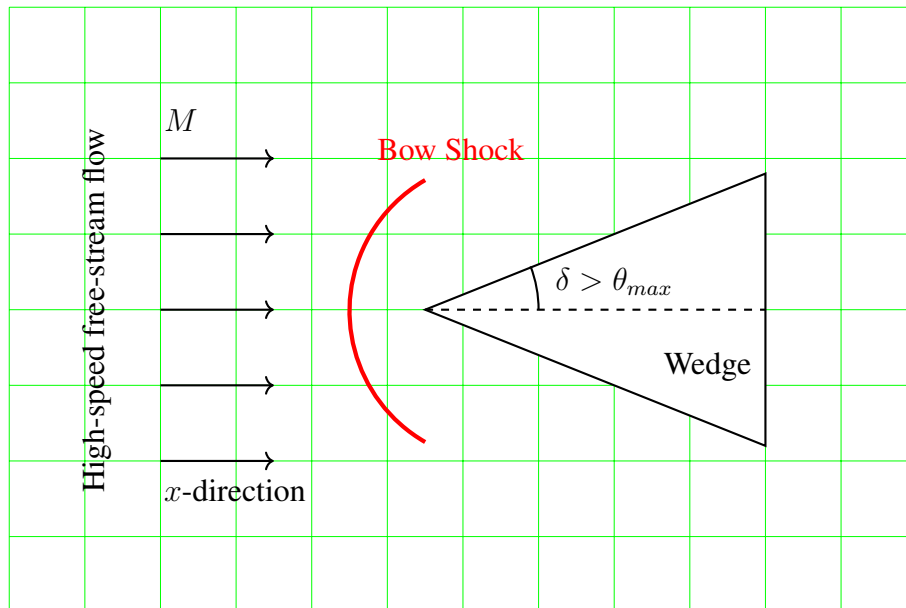
Figure 1.6: Bow shock front a wedge.

Report [69].

1.3 On the Shapes of Hypersonic Vehicles

One interesting point regarding the shapes of hypersonic vehicles is their blunt-shaped configurations. The question "*Wouldn't it be more reasonable for a high-speed vehicle to have sharp leading edges to reduce drag forces?*" may arise. If the only point of view is the reduction of drag forces, sharp edges would be indispensable. However, it should be noticed that the sharp edges generate severe high temperatures. This situation could not be fully analyzed until the early 1950s. The slender designs with sharp leading edges were used to minimize possible shocks at supersonic speeds. However, these designs would, at hypersonic speeds, cause sharp edges to melt due to overheating. After the studies of Hayes and Probstein [80] and Moretti and Abbett [141] in 1966, hypersonic research focused on blunt-body configurations.

Anderson explains the relationship between the energy and shapes of the vehicles in a very elegant way in [7]. Let us consider an aircraft moving at a very high altitude and speed. Both the kinetic and potential energies are very high due to the speed and altitude, and, when it lands on the ground, both energies will disappear. So, where

did this energy spend? The answer is: "*it was spent on heating the flow environment and the body of the vehicle flying.*" If the design of the vehicle has a slender shape with sharp leading edges, then the possible shock waves will be weak. Therefore, a large amount of the energy will be spent on heating the vehicle. As the radius of the blunt nose increases, possible shocks will be stronger. Thus, most of the energy will be spent on heating the flow environment, which will reduce the vehicle's heating. In [173], Sziroczak and Smith give a historical overview, and discuss the technical issues associated with the designs of hypersonic vehicles taking a multi-disciplinary approach. They furthermore present a summary of hypersonic vehicles by country of origin, first launch date, maximum speed, flight altitude, and payload.

The blunt-nose design is vital for hypersonic aerodynamics as it reduces the aerodynamic heating of the vehicle. As reported in [142] by Grundmann, any blunt-nose shaped (re-)entry vehicle experiences extremely high velocities around 11.2 km/s called *orbital escape velocity*, and at these speeds, the temperature behind the bow shock is about $11,000$ K, while the temperature at the surface of the Sun is about $5,000$ K [7]. The simplified models are extremely insufficient to simulate the high-temperature effects accurately. Moreover, the calorically perfect gas assumption is no longer valid at these temperatures. Therefore, there is a need for models that can provide more realistic and accurate results. Further details are investigated in Chapter 2.

In high-speed flow computations conducted in this thesis, 2D configurations are adopted as a space vehicle's blunt-nose. Although the 2D simulations are not always possible for many 3D models (they may not be good representations of the real problem), the 2D reduction might be quite advantageous in many aspects in modeling plane stress, plane strain, and axisymmetric problems. Some advantages of reducing suitable 3D models to 2D under appropriate conditions can be summarized as follows:

- They provide drastic savings in terms of memory usage, computational time, and power consumption.

- They provide insight and guidance on how the 3D model (real-life phenomenon) works.

- If commercial software is not used, it is quite possible to get lost in 3D models. Furthermore, diagnosing and resolving a problem encountered in a 3D model can be much more difficult than in a 2D model.

For more discussion on the representativeness of 2D models used in industrial and academic simulations, the interested reader can refer to [51, 110, 155].

1.4 Notes on the Tools of CFD

Computational fluid dynamics (CFD) is a sub-discipline of fluid mechanics that uses numerical methods to simulate and analyze fluid flows. The main tools of CFD are based on three discretization methods in general: the finite difference method (FDM), the finite volume method (FVM), and the finite element method (FEM). Each of these methods has its own set of benefits and drawbacks discussed in the following paragraphs briefly. In comparison to the other two methods and their variants, the finite volume methods are much more widely used for solving CFD problems.

The FDM is the oldest method for the numerical solution of differential equations and is based on replacing the derivatives appearing in differential equations with corresponding finite difference approximations [61]. This procedure usually comes with a large but finite system of algebraic equations. Compared to other methods, the FDM is relatively simpler to employ in terms of both computation and programming. Besides that, the method has a very robust theory. Unfortunately, the FDM suffers from the inability to handle complex geometries. One can refer to [123] by LeVeque for more on finite difference methods for solving ordinary and partial differential equations.

The FVM was first proposed by McDonald [137] and MacCormac and Paullay [130] for solving 2D Euler equations in the early 1970s and is based on balancing fluxes through control volumes from a physical point of view. Conservation equations (laws) are discretized directly in space in their integral formulation, and the method ensures that quantities (mass, momentum, and energy) are conserved at a discretized level, that is, at a local scale. The fluxes between adjacent control volumes are directly balanced. The method is able to handle discontinuities accurately. Although not as much as the

15

FEM and FDM, it has an strong theoretical background. Its deficiency is the inability to achieve high-order accuracy on general grids because of the extended stencils. One can find more details and applications concerning the FVM in [61, 199].

Although Clough [41] introduced the phrase "finite element method" into the literature in a paper on solving plane elasticity problems in 1960, the concepts of finite elements date back much further. The FEM originates from Courant's work [46] conducted in 1943; the first attempt to use piecewise continuous functions defined over triangular domains appears in the applied mathematics literature. After being implemented to various linear and nonlinear structural mechanics problems, the FEM was subsequently employed for solving continuous field problems in 1965 by Cheung and Zienkiewicz [39]. The FEM is very suitable for multi-physics problems defined on complex geometries. It has an extensively developed mathematical theory in terms of convergence, stability, and accuracy analyses. The method is still employed for solving a wide range of problems arising in industry and for academic research, increasingly. FEM's historical development can be found in [60] by Felippa.

Many CFD problems are governed by first-order systems such as the Euler equations, and their solutions usually contain discontinuities, especially near shocks. The FEM approximations suffer from containing nonphysical oscillations for simulating high-speed flows and in solving convection-dominated problems (see Section 3.1). Even though these instability issues can be overcome with stabilized formulations and shock-capturing techniques in the context of finite element methods, the FVM has become a favorite against the FEM over time. However, CFD's history shows that once a suitable FEM formulation is determined, the method is used exclusively because of its advantages in handling complex geometries and achieving higher-order accuracy [52]. For more on the tools of CFD from various perspectives, one can refer to [63, 111, 154, 195].

1.5   Objective and Contribution of the Thesis

The determination of this topic is immensely affected by the fact that the classical discretization methods (e.g., the Galerkin finite element method, central finite differ-

ences scheme) experience drastic instability issues in simulating high-speed flows. In performing such simulations, the Galerkin finite element method (GFEM) can be stabilized using the compressible-flow Streamline-Upwind/Petrov–Galerkin (SUPG) formulation. However, even if a stabilized formulation is used, it is a well-known fact that there is still a need for special techniques to resolve strong gradients accurately at shocks and near discontinuities. Therefore, the stabilized formulations are needed to be supplemented with shock-capturing (also called discontinuity-capturing) techniques as well. To this end, the objectives of this thesis can be given as follows:

- As we approach the middle of the 21st century, humanity's curiosity and search for extraterrestrial life are increasing day by day for many reasons. The rockets and capsules to perform such missions are exposed to tremendous high speeds and resulting high-temperature effects that must be carefully examined and evaluated for these tasks to be carried out successfully. The main objective of this study is to contribute to a better understanding of high-speed flow research by simulating flows ranging from thermochemical equilibrium to nonequilibrium states and supersonic to hypersonic regimes.

- As discussed in Section 1.4, although the FEM achieves successful (nonoscillatory) results in solving the problems arising in CFD when supplemented with appropriate stabilization methods and shock-capturing techniques, the FVM has become a favorite tool in CFD over time. Therefore, this study aims to contribute to breaking down the hesitations about the employment of the finite element methods for simulating high-speed flows by using appropriate stabilization and shock-capturing techniques.

- Since the vast majority of research in the literature to simulate high-speed flows is based on commercial software, and almost all commercial flow solvers rely on finite volume methods, it is not surprising that high-speed flow research is overwhelmingly based on finite volume methods. Therefore, one of the objectives of this thesis is to develop compressible flow solvers in the FEniCS environment (an open-source FEM solver) as well for simulating hypersonic flows in thermochemical equilibrium/nonequilibrium. Some of the reasons for choosing FEniCS are that it is open-source, available free of charge, being developed

continuously collaboratively, has a growing experience sharing platform, and has a language for mathematical abstractions and definitions very similar to pen-and-paper notation that the reader can easily understand. Further details on the FEniCS project are provided in Section 4.8.

The principal contributions of the thesis can be summarized as follows:

- A comprehensive set of convection-dominated test problems, including the Burgers'-type equations at high Reynolds numbers, are studied by employing the SUPG formulation. The SUPG-stabilized formulations are also augmented with YZβ shock-capturing terms in order to obtain better solution profiles around sharp gradients. We modify the SUPG-YZβ formulation, which has been almost always used for computing compressible flows, to solve 2D convection-dominated equations, including the Burger's-type equations.

- High-speed (supersonic and hypersonic) flows in thermochemical equilibrium/nonequilibrium are investigated. Towards this end, a compressible-flow SUPG formulation supplemented with YZβ shock-capturing is proposed. For the equilibrium case, nitrogen (N_2) flow around a cylinder is studied. For the nonequilibrium case, flow of a five-species (O, N, NO, O_2, N_2) air mixture around a cylinder is considered with 17 chemical reactions. In nonequilibrium simulations, Park's two-temperature ($T - T_V$) model is adopted. We modify the compressible-flow SUPG-YZβ formulation for computing nonequilibrium flows for the first time in this thesis. The solvers are developed in the FEniCS environment.

- Enforcing the slip (zero-normal-velocity, no-penetration, impermeable) boundary conditions on solid surfaces (walls) is not a trivial task in simulating inviscid flows compared to the well-known no-slip boundary conditions of viscous flows. Two different approaches, i.e., weak implementation and a penalty technique, are employed for imposing the slip boundary condition on the cylinder surface. The performances of these methods are compared in simulating high-speed flows for the first time.

1.6 Thesis Layout

This thesis consists of six chapters. There is also an Appendices section at the end of the study that contains any additional information that the reader may require.

Chapter 2 presents the equations governing inviscid flows of five-species (O, N, NO, O_2, N_2) air mixture. The formulations are given for both the N–S equations that govern viscous flows and the Euler equations that govern inviscid flows.

In Chapter 3, the SUPG formulation and $YZ\beta$ shock-capturing technique are introduced for solving convection-dominated problems. The proposed formulations are tested on a wide range of problems, e.g., steady-state/ time-dependent, linear/nonlinear, scalar/coupled.

The computational details of the simulations performed in the thesis are the focus of Chapter 4: the computational domain, mesh, and initial and boundary conditions are addressed, the implementation of the zero-normal-velocity boundary condition on the cylinder surface is introduced. The chapter ends with an introduction of temporal discretization of the governing equations, as well as a discussion of iterative techniques for solving linear/nonlinear equation systems.

In Chapter 5, the simulations of supersonic and hypersonic flows around a cylinder are carried out. The relevant simulations are performed for both thermochemical equilibrium and nonequilibrium flows.

Finally, Chapter 6 summarizes the methods and techniques used in the thesis and discusses the results obtained. In addition, a comprehensive list of suggestions for future research directions is presented.

CHAPTER 2

MATHEMATICAL MODELING AND THERMOCHEMISTRY BEHIND HYPERSONIC FLOW

The Navier–Stokes (N–S) equations are at the heart of computational fluid dynamics (CFD). In physics, mathematics, and engineering sciences, as well as in industry, the N–S equations have numerous applications. They are reduced to the Euler equations when the viscous forces are completely ignored, i.e., the stress tensor is controlled only by the pressure contribution. Since the Reynolds number is high, i.e., viscosity is low, at high velocities, viscosity and turbulence effects are negligible except for a small region near the walls [61]. Therefore, the Euler equations of compressible flows are very useful descriptive tools for high-speed flow simulations.

Although the presence of viscous stress terms, mathematically expressed by second-order derivatives, complicates the N–S equations, the situation is quite different from a computational viewpoint; indeed, the presence of these terms plays a crucial role in viscous regularization (stabilization) of (numerical) solutions [75]. The Euler equations, on the other hand, lack viscous terms and allow for shock waves and (contact) discontinuities in solutions, necessitating the use of advanced computational techniques to capture these phenomena accurately. Besides that, because the flow is considered inviscid, the fluid cannot adhere to walls and slips [61]. As a result, problems involving the Euler equations require the enforcement of slip (zero-normal-velocity) boundary conditions (as opposed to the well-known no-slip condition of N–S equations), i.e., the normal velocity vanishes at walls. Imposing the slip condition for complex geometries (e.g., on a nonrectangular obstacle) is a challenging task that involves the use of specialized techniques [145]. The details are discussed in

Section 4.3.4. For characteristic properties of the Euler equations, one can also refer to Appendix B. Besides, Appendix C presents a brief survey on the existence and uniqueness of the Euler equations of compressible flows in the continuous setting.

In this chapter, multispecies N–S equations of compressible flows are studied for modeling hypersonic flows in thermochemical nonequilibrium. Thermodynamic relations, transport properties, and chemical reaction mechanisms are examined in detail. First, the N–S equations are given more compactly in a quasi-linear form. Then, the compressible Euler equations are presented for single-species gas flows. Finally, single-species Euler equations are extended to include five-species. Although only the Euler equations are used in this thesis for computing hypersonic flows, we start with the N–S equations in order to better understand the physics behind the formulations.

2.1 Governing Equations of Five-Species Viscous Hypersonic Flow

In this section, the classical N–S equations are modified for thermochemical nonequilibrium flow computations to include multispecies as done in the NASA technical report by Gnoffo et al. [72]. For a simpler chemistry model with a special emphasis on the symmetrization of the conservation equations in terms of the entropy variables, one can refer to the study [34] by Chalot et al.

2.1.1 Conservation Equations

The 2D five-species mass conservation equations of compressible flows can be given as follows [72, 107]:

$$\frac{\partial \rho_O}{\partial t} + \frac{\partial \rho_O u_1}{\partial x_1} + \frac{\partial \rho_O u_2}{\partial x_2} = \frac{\partial \left(\rho D_O \frac{\partial Y_O}{\partial x_1} \right)}{\partial x_1} + \frac{\partial \left(\rho D_O \frac{\partial Y_O}{\partial x_2} \right)}{\partial x_2} + \omega_O, \qquad (2.1)$$

$$\frac{\partial \rho_N}{\partial t} + \frac{\partial \rho_N u_1}{\partial x_1} + \frac{\partial \rho_N u_2}{\partial x_2} = \frac{\partial \left(\rho D_N \frac{\partial Y_N}{\partial x_1} \right)}{\partial x_1} + \frac{\partial \left(\rho D_N \frac{\partial Y_N}{\partial x_2} \right)}{\partial x_2} + \omega_N, \qquad (2.2)$$

$$\frac{\partial \rho_{NO}}{\partial t} + \frac{\partial \rho_{NO} u_1}{\partial x_1} + \frac{\partial \rho_{NO} u_2}{\partial x_2} = \frac{\partial \left(\rho D_{NO} \frac{\partial Y_{NO}}{\partial x_1} \right)}{\partial x_1} + \frac{\partial \left(\rho D_{NO} \frac{\partial Y_{NO}}{\partial x_2} \right)}{\partial x_2} + \omega_{NO}, \qquad (2.3)$$

22

$$\frac{\partial \rho_{O_2}}{\partial t} + \frac{\partial \rho_{O_2} u_1}{\partial x_1} + \frac{\partial \rho_{O_2} u_2}{\partial x_2} = \frac{\partial \left(\rho D_{O_2} \frac{\partial Y_{O_2}}{\partial x_1} \right)}{\partial x_1} + \frac{\partial \left(\rho D_{O_2} \frac{\partial Y_{O_2}}{\partial x_2} \right)}{\partial x_2} + \omega_{O_2}, \qquad (2.4)$$

$$\frac{\partial \rho_{N_2}}{\partial t} + \frac{\partial \rho_{N_2} u_1}{\partial x_1} + \frac{\partial \rho_{N_2} u_2}{\partial x_2} = \frac{\partial \left(\rho D_{N_2} \frac{\partial Y_{N_2}}{\partial x_1} \right)}{\partial x_1} + \frac{\partial \left(\rho D_{N_2} \frac{\partial Y_{N_2}}{\partial x_1} \right)}{\partial x_2} + \omega_{N_2}, \qquad (2.5)$$

where the subscripts O, N, NO, O_2, and N_2 represent the species. Here, u_1 and u_2 are the components of the velocity field $\mathbf{u}$ in spatial directions x_1 and x_2, respectively. The term ρ_s denotes the density of species s, and the D_s are the effective diffusion coefficients of the species. The terms ω_s stand for the species mass rates of production, as well as any other components that may enter the equations, such as external forces. The mass fraction of species s is given as

$$Y_s = \frac{\rho_s}{\rho}, \qquad (2.6)$$

where the mixture density, ρ, is given by Eq. (2.12). One should notice that the sum of the species mass fractions is one:

$$\sum_s Y_s = \sum_s \frac{\rho_s}{\rho} = 1. \qquad (2.7)$$

The molar mass (molecular weight), $\overline{M}$, of the gas mixture is given as

$$\overline{M} = \left(\sum_s \frac{Y_s}{M_s} \right)^{-1} = \left(\sum_s c_s \right)^{-1}, \qquad (2.8)$$

where the molecular concentration of species s is defined as

$$c_s = \frac{\rho_s}{\rho M_s}. \qquad (2.9)$$

Unless otherwise stated, throughout this thesis, the notations $\sum_s$ and $\sum_{mol}$ refer to the summations over species $s \in \{O, N, NO, O_2, N_2\}$ and molecules mol $\in \{NO, O_2, N_2\}$.

The 2D global (mixture) momentum conservation equations are given as [107]

$$\frac{\partial \rho u_1}{\partial t} + \frac{\partial \left(\rho u_1^2 + p \right)}{\partial x_1} + \frac{\partial \left(\rho u_1 u_2 \right)}{\partial x_2} = \frac{\partial}{\partial x_1} \sigma_{11} + \frac{\partial}{\partial x_2} \sigma_{12}, \qquad (2.10)$$

$$\frac{\partial \rho u_2}{\partial t} + \frac{\partial \left(\rho u_2 u_1 \right)}{\partial x_1} + \frac{\partial \left(\rho u_2^2 + p \right)}{\partial x_2} = \frac{\partial}{\partial x_1} \sigma_{21} + \frac{\partial}{\partial x_2} \sigma_{22}, \qquad (2.11)$$

where the ρ is the density of the gas mixture given as [72]

$$\rho = \sum_s \rho_s. \tag{2.12}$$

The σ_{ij}'s are the viscous stress tensors defined as

$$\sigma_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \mu_B \frac{\partial u_k}{\partial x_k} \delta_{ij}, \tag{2.13}$$

where the term μ denotes the shear viscosity. The term μ_B represents the bulk viscosity and is defined as

$$\mu_B = -\frac{2}{3}\mu. \tag{2.14}$$

The term δ_{ij} represents the *Kronecker delta*:

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{2.15}$$

To this end, based on the *Stokes' hypothesis* for Newtonian fluids, the viscous stresses are explicitly written as follows:

$$\sigma_{11} = \frac{4}{3}\mu \frac{\partial u_1}{\partial x_1} - \frac{2}{3}\mu \frac{\partial u_2}{\partial x_2}, \tag{2.16}$$

$$\sigma_{12} = \sigma_{21} = \mu \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right), \tag{2.17}$$

$$\sigma_{22} = \frac{4}{3}\mu \frac{\partial u_2}{\partial x_2} - \frac{2}{3}\mu \frac{\partial u_1}{\partial x_1}. \tag{2.18}$$

The mixture pressure, p, is obtained by using *Dalton's partial pressures law* [72]:

$$p = \sum_s p_s = \sum_s \rho_s R_s T = \sum_s \rho_s \frac{R_u}{M_s} T, \tag{2.19}$$

where the p_s are the species individual pressures, and the R_s are the species specific gas constants defined as

$$R_s = \frac{R_u}{M_s}. \tag{2.20}$$

Here, $R_u = 8,314.4598$ J/(kmol·K) is the universal gas constant, and the M_s are the species molar masses (see Appendix A).

The 2D total energy conservation equation reads [107]:

$$\frac{\partial \rho e}{\partial t} + \frac{\partial \left(\rho h u_1\right)}{\partial x_1} + \frac{\partial \left(\rho h u_2\right)}{\partial x_2} = \frac{\partial \left(u_1 \sigma_{11} + u_2 \sigma_{12}\right)}{\partial x_1} + \frac{\partial \left(u_1 \sigma_{21} + u_2 \sigma_{22}\right)}{\partial x_2}$$
$$+ \frac{\partial \left(k_{tr}\frac{\partial T}{\partial x_1} + k_v \frac{\partial T_V}{\partial x_1}\right)}{\partial x_1} + \frac{\partial \left(k_{tr}\frac{\partial T}{\partial x_2} + k_v \frac{\partial T_V}{\partial x_2}\right)}{\partial x_2}$$
$$+ \frac{\partial \left(\sum_s \rho D_s h_s \frac{\partial Y_s}{\partial x_1}\right)}{\partial x_1} + \frac{\partial \left(\sum_s \rho D_s h_s \frac{\partial Y_s}{\partial x_2}\right)}{\partial x_2}, \quad (2.21)$$

where the total enthalpy, h, is given as [72]

$$h = e + \frac{p}{\rho}, \quad (2.22)$$

and the total energy, e, is defined by Eq. (1.6). The terms h_s denote the species enthalpy, and T is the translational-rotational temperature.

The 2D total vibrational-electron-electronic energy is governed by [107]

$$\frac{\partial \rho e_V}{\partial t} + \frac{\partial \left(\rho e_V u_1\right)}{\partial x_1} + \frac{\partial \left(\rho e_V u_2\right)}{\partial x_2} = \frac{\partial \left(k_v \frac{\partial T_V}{\partial x_1}\right)}{\partial x_1} + \frac{\partial \left(k_v \frac{\partial T_V}{\partial x_2}\right)}{\partial x_2}$$
$$+ \frac{\partial \left(\sum_s \rho D_s e_{V,s} \frac{\partial Y_s}{\partial x_1}\right)}{\partial x_1} + \frac{\partial \left(\sum_s \rho D_s e_{V,s} \frac{\partial Y_s}{\partial x_2}\right)}{\partial x_2} + \omega_V, \quad (2.23)$$

where T_V represents the vibrational-electron-electronic temperature. Besides, k_{tr} and k_v represent the thermal conductivities for the translational-rotational and vibrational energy modes, respectively. The term k_{tr} is calculated as the sum of the thermal conductivities of the translational and rotational energy modes as follows [107]:

$$k_{tr} = k_t + k_r. \quad (2.24)$$

The terms $e_{V,s}$ represent the species vibrational-electron-electronic energies, and ω_V is the mixture vibrational-electronic energy source term. If the vibrational-electron-electronic energy equation (2.23) is ignored in the coupled system of governing equations, then thermal equilibrium is assumed.

All the terms appearing in governing equations are discussed in detail in the following sections.

2.1.2 Transport Properties

The *Wilke–Blottner–Eucken rule* can be used for determining the flow field's transport properties. It is reported in [150] that this approach is reliable up to temperatures of $10,000$ K. For higher temperatures, up to $30,000$ K, it is reported in [76] that *Gupta's collision cross-section rule* is more reliable.

The Lewis number, (Le), plays a crucial role in transport phenomena and is defined as

$$Le = \frac{\text{Schmidt number } (Sc)}{\text{Prandtl number } (Pr)}. \tag{2.25}$$

As given in [107], a constant Lewis number approach can be adopted for calculating the species diffusion coefficients:

$$D_s = D = Le \frac{k_{tr}}{\rho C_{tr,p}}, \tag{2.26}$$

where $C_{tr,p}$ is the mixture translational-rotational specific heat at constant pressure.

Another approach proposed in [208] reads:

$$D_s = \begin{cases} \frac{(1-Y_s)\mu}{(1-\chi_s)\rho Sc}, & \text{if } s \text{ is not ion,} \\ M_e \frac{\sum\limits_{i=\text{ion}} D_i(Y_i/M_i)}{\sum\limits_{i=\text{ion}} Y_i}, & \text{if } s \text{ is ion.} \end{cases} \tag{2.27}$$

For atomic and molecular particles, the Schmidt number is $Sc = 0.5$, while for ions, it is set $Sc = 0.25$. The Prandtl number is defined as the ratio of momentum diffusivity to thermal diffusivity. For specific values of the Prandtl number, one can refer to [200]. The term M_e in Eq. (2.27) represents the molecular weight of an electron.

The idea behind expressing the species diffusion coefficients, D_s, with a single coefficient, D, stems from the fact that the species used to model the air mixture have similar molecular weights. This assumption would not be appropriate in a model involving the particles hydrogen and oxygen, for example, due to the considerable molecular weight difference between these particles. Besides, it is reported in [74] that the models with a single diffusion coefficient do not give accurate results for speeds over 10 km/s.

Another constant Lewis number diffusion model used in [74] and the mixing rule

of [72] can be combined as in [78]:

$$\rho D_s = \begin{cases} Le \frac{(1-Y_s)}{(1-\chi_s)} \frac{k_{tr}}{C_{tr,p}}, & \text{if } s \text{ is not ion,} \\ \\ 2Le \frac{(1-Y_s)}{(1-\chi_s)} \frac{k_{tr}}{C_{tr,p,s}}, & \text{if } s \text{ is ion,} \end{cases} \qquad (2.28)$$

where χ_s denotes the molar fraction of species s, and is given as [72]

$$\chi_s = \frac{\frac{\rho_s}{M_s}}{\sum\limits_{i=1}^{N_s} \frac{\rho_i}{M_i}}. \qquad (2.29)$$

Here, the term N_s represents the number of species, and $C_{tr,v,s}$ is the species specific heat at constant volume. The Lewis number is generally set $Le = 1.4$ for a reacting air mixture. In this model, the diffusion transportation for electrons is

$$\rho D_e = M_e \frac{\sum\limits_{s=\text{ion}} \rho_s D_s (Y_s/M_s)}{\sum\limits_{s=\text{ion}} (Y_s/M_s)}. \qquad (2.30)$$

The translational-rotational specific heat for species s is given as the sum of the translational and rotational specific heats at constant volume as [107]

$$C_{tr,v,s} = C_{trans,v,s} + C_{rot,v,s}. \qquad (2.31)$$

Note that the rotational energy is only defined for molecular particles since monoatomic particles do not have rotational energies. Therefore, the translational-rotational specific heat of species s can be given as follows [107]:

$$C_{tr,v,s} = \begin{cases} \frac{5}{2} \frac{R_u}{M_s}, & \text{if } s \text{ is molecule,} \\ \\ \frac{3}{2} \frac{R_u}{M_s}, & \text{if } s \text{ is atom.} \end{cases} \qquad (2.32)$$

The specific heat of species s for vibrational energy mode at constant volume is given as (see Appendix A)

$$C_{vib,v,s} = \begin{cases} \frac{R_u}{M_s} \frac{\left(\frac{\theta_{v,s}}{T_v}\right)^2 \exp\left(\frac{\theta_{v,s}}{T_v}\right)}{\left[\exp\left(\frac{\theta_{v,s}}{T_v}\right)-1\right]^2}, & \text{if } s \text{ is molecule,} \\ \\ 0, & \text{if } s \text{ is atom/electron.} \end{cases} \qquad (2.33)$$

The viscosity and thermal conductivity of the gas mixture are determined using *Wilke's mixing rule* [204]. The mixture viscosity is defined as [107]

$$\mu = \sum_s \frac{\chi_s \mu_s}{\phi_s}, \qquad (2.34)$$

and the thermal conductivity is given as

$$\kappa = \sum_s \frac{\chi_s k_s}{\phi_s},$$ (2.35)

where the scaling factor is as follows:

$$\phi_s = \sum_{r=1}^{N_s} \frac{\chi_r \left[1 + \sqrt{\frac{\mu_s}{\mu_r} \left(\frac{M_r}{M_s} \right)^{\frac{1}{4}}} \right]^2}{\sqrt{8 \left(1 + \frac{M_s}{M_r} \right)}}.$$ (2.36)

The terms μ_s and k_s represent the species viscosity and the species thermal conductivity, respectively.

The species viscosity coefficients, μ_s, are determined by using *Blottner's curve-fitting approach* [24]:

$$\mu_s = 0.1 \exp \left[\left(A_s \ln T + B_s \right) \ln T + C_s \right],$$ (2.37)

where A_s, B_s, and C_s are the Blottner's curve-fitting coefficients given in Appendix A. Then, the mixture viscosity can be computed as given in Eq. (2.34).

The thermal conductivity terms of species s for the translational, rotational, and vibrational energy modes are determined by using the *Eucken's relation* [57] as follows:

$$k_{trans,s} = \frac{5}{2} \mu_s C_{trans,v,s},$$ (2.38)

$$k_{rot,s} = \mu_s C_{rot,v,s},$$ (2.39)

$$k_{vib,s} = \mu_s C_{vib,v,s}.$$ (2.40)

As reported in [72], the thermal conductivity associated with the vibrational energy mode can be assumed to be equal to that of the rotational mode,

$$k_{vib,s} = k_{rot,s}.$$ (2.41)

Then, the thermal conductivity terms associated with different energy modes can be calculated using Eq. (2.35).

2.1.3 Energy Equations

Based on the discussion conducted in Section 1.2.2, the translational-rotational energy, e_s^{tr}, of species s now can be given as follows [107]:

$$e_s^{tr} = e_s^{trans} + e_s^{rot} = C_{tr,v,s}T,$$ (2.42)

where e_s^{trans} and e_s^{rot} are species translational and rotational energies. Therefore, the mixture total translational-rotational energy reads

$$e^{tr} = \sum_s \frac{\rho_s}{\rho} e_s^{tr}.$$ (2.43)

The vibrational energy for diatomic molecules O_2, N_2, and NO, based on the harmonic oscillator model, is given as [200]

$$e_s^{vib} = \begin{cases} \frac{R_s \theta_{v,s}}{\exp\left(\frac{\theta_{v,s}}{T_v}\right)-1}, & \text{if } s \text{ is molecule,} \\ \\ 0, & \text{if } s \text{ is atom.} \end{cases}$$ (2.44)

The electron-electronic energy of species s can be written as

$$e_s^{elec} = R_s \frac{\sum\limits_{i=1}^{\infty} \theta_{i,s}^{elec} g_{i,s} \exp\left(-\frac{\theta_{i,s}^{elec}}{T_e}\right)}{g_{0,s} + \sum\limits_{i=1}^{\infty} g_{i,s} \exp\left(-\frac{\theta_{i,s}^{elec}}{T_e}\right)},$$ (2.45)

where $\theta_{v,s}$ is the characteristic vibrational temperature, $\theta_{i,s}^{elec}$ is the characteristic electronic temperature, and T_e is the electron-electronic excitation temperature (see Appendix A). The terms $g_{0,s}$ and $g_{i,s}$ are the degeneracy of the ground and the ith electronic level, respectively. As discussed in Section 1.2.2, we assume that the electron-electronic and vibrational temperatures are equal:

$$T_V = T_v = T_e.$$ (2.46)

The mixture vibrational-electron-electronic energy, e_V, can be given by the relation

$$\rho e_V = \sum_s \left(\rho_s e_s^{vib} + \rho_s e_s^{elec} \right).$$ (2.47)

If the electron-electronic energies are neglected, the mixture total vibrational energy reads

$$e_V = \sum_s \frac{\rho_s}{\rho} e_s^{vib}.$$ (2.48)

Then, the internal energy can be defined as [208]

$$e^{int} = e^{trans} + e^{rot} + e^{vib} + e^{elec} + h_0$$
$$= e^{tr} + e_V + \sum_s \frac{\rho_s}{\rho} h_{0,s}, \tag{2.49}$$

where the species heat of formation energies, $h_{0,s}$, are given in Appendix A.

The species enthalpies are defined as [66]

$$h_s = \begin{cases} R_s T + e_s^{int}, & \text{if } s \text{ is molecule/atom,} \\ R_e T_V + e_e^{int}, & \text{if } s \text{ is electron,} \end{cases} \tag{2.50}$$

where the species internal energies are given as

$$e_s^{int} = \begin{cases} e_s^{tr} + h_{0,s} + e_{V,s}, & \text{if } s \text{ is molecule/atom,} \\ e_{V,e}, & \text{if } s \text{ is electron.} \end{cases} \tag{2.51}$$

The vibrational-electron-electronic energy, $e_{V,s}$, for species s is

$$e_{V,s} = \begin{cases} e_s^{vib} + e_s^{elec}, & \text{if } s \text{ is molecule/atom,} \\ C_{trans,v,e} T_V, & \text{if } s \text{ is electron,} \end{cases} \tag{2.52}$$

with

$$C_{trans,v,e} = \frac{3}{2} \frac{R_u}{M_e}. \tag{2.53}$$

Now, the total energy addressed by Eq. (1.6) in Section 1.2.2 can be given as follows:

$$e = e^{tr} + e_V + \sum_s \frac{\rho_s}{\rho} h_{0,s} + \frac{1}{2} \|\mathbf{u}\|^2. \tag{2.54}$$

The vibrational-electron-electronic energy given in Eq. (2.48) is obtained by solving the governing equations. The computational details for obtaining the translational-rotational temperature, T, and the vibrational-electron-electronic temperature, T_V, of the two-temperature model are discussed in Section 4.7. The norm $\|\cdot\|$ represents the standard Euclidean norm.

2.1.4   Chemical Reactions Source Terms

Since a five-species air mixture model is adopted in simulations performed in this thesis, the most important chemical reactions between the species O, N, NO, O_2, and N_2 can

be given as follows [152]:

$$N_2 + \Pi \rightleftharpoons 2N + \Pi,$$

$$O_2 + \Pi \rightleftharpoons 2O + \Pi,$$

$$NO + \Pi \rightleftharpoons N + O + \Pi,$$

$$N_2 + O \rightleftharpoons NO + N,$$

$$NO + O \rightleftharpoons O_2 + N,$$
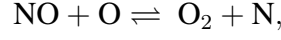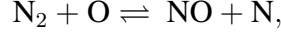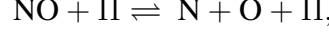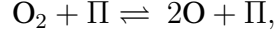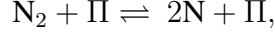
where $\Pi \in \{O, N, NO, O_2, N_2\}$ is one of the possible collision partners. The first three reactions are called *dissociation process*, and the last two are called *exchange reactions* or *Zel'dovich reactions*. These both types of reactions may take place in the forward and backward directions, as shown by bidirectional arrows.

Let k_{fr} and k_{br} denote the reaction rate coefficients in the forward and the backward directions, respectively, then the corresponding reaction rates become [72]

$$\mathcal{R}_1 = \sum_s \left[-k_{fr1s} \frac{\rho_{N_2}}{M_{N_2}} \frac{\rho_s}{M_s} + k_{br1s} \frac{\rho_N}{M_N} \frac{\rho_N}{M_N} \frac{\rho_s}{M_s} \right], \tag{2.55}$$

$$\mathcal{R}_2 = \sum_s \left[-k_{fr2s} \frac{\rho_{O_2}}{M_{O_2}} \frac{\rho_s}{M_s} + k_{br2s} \frac{\rho_O}{M_O} \frac{\rho_O}{M_O} \frac{\rho_s}{M_s} \right], \tag{2.56}$$

$$\mathcal{R}_3 = \sum_s \left[-k_{fr3s} \frac{\rho_{NO}}{M_{NO}} \frac{\rho_s}{M_s} + k_{br3s} \frac{\rho_N}{M_N} \frac{\rho_O}{M_O} \frac{\rho_s}{M_s} \right], \tag{2.57}$$

$$\mathcal{R}_4 = -k_{fr4s} \frac{\rho_{N_2}}{M_{N_2}} \frac{\rho_O}{M_O} + k_{br4s} \frac{\rho_{NO}}{M_{NO}} \frac{\rho_N}{M_N}, \tag{2.58}$$

$$\mathcal{R}_5 = -k_{fr5s} \frac{\rho_{NO}}{M_{NO}} \frac{\rho_O}{M_O} + k_{br5s} \frac{\rho_{O_2}}{M_{O_2}} \frac{\rho_N}{M_N}. \tag{2.59}$$

As one can point out from Eqs. (2.55)–(2.59), we assume that seventeen different chemical reactions can take place in the flow field.

The effective (controlling) temperature is given based on *Park's two-temperature model* as [152]

$$\overline{T} = T^a T_V^b, \tag{2.60}$$

where $a + b = 1$. The vast majority of the studies in the literature concentrate on the two scenarios:

$$\overline{T} = \sqrt{TT_V} \tag{2.61}$$

and

$$\overline{T} = T^{0.7} T_V^{0.3}. \tag{2.62}$$

It is reported by Sarma [164] and Candler and Nompelis in [30] that Eq. (2.62) gives better results and is adopted in this thesis.

It is clear that in the case of thermal equilibrium, the translational-rotational and vibrational temperatures are equal, i.e., $\overline{T} = T$. The effective temperature-dependent forward reaction rate coefficients are calculated with a modified form of *Arrhenius formula* [72]:

$$k_{fr}\left(\overline{T}\right) = C_{fr}\overline{T}^{\Upsilon_r} \exp\left(-\frac{E_{ar}}{R_u\overline{T}}\right),\qquad(2.63)$$

where C_{fr} is the reaction rate constant, Υ_r is the pre-exponential factor, and E_{ar} is the activation energy for reaction r. Then, the corresponding backward rate coefficients are given by the relation [200]

$$K_{eq,r} = \frac{k_{fr}}{k_{br}}.\qquad(2.64)$$

Here, the term $K_{eq,r}$ represents the equilibrium constant for reaction r and is expressed in the same fashion as given in [152]:

$$K_{eq,r}\left(T\right) = \exp\left[A_1\left(1/Z\right) + A_2 + A_3\ln\left(Z\right) + A_4\left(Z\right) + A_5\left(Z\right)^2\right],\qquad(2.65)$$

where

$$Z = \frac{10,000}{T}.\qquad(2.66)$$

Another approach for determining the equilibrium constant, $K_{eq,r}$, is the Gibbs free energy formula [12], but this will not be addressed in this study.

The species source terms can be given in a canonical form [72]:

$$\omega_s = M_s \sum_{r=1}^{nr} \left(\alpha_{sr} - \beta_{sr}\right)\left(\mathcal{R}_{br} - \mathcal{R}_{fr}\right),\qquad(2.67)$$

where nr denotes the total number of reactions, α_{sr} and β_{sr} are the stoichiometric coefficients of the reactants and products in reaction r, respectively. Eventually, the species source terms emerging from the chemical reactions are given as follows [72]:

$$\omega_{\text{O}} = M_{\text{O}}\left(-2\mathcal{R}_2 - \mathcal{R}_3 + \mathcal{R}_4 + \mathcal{R}_5\right),\qquad(2.68)$$

$$\omega_{\text{N}} = M_{\text{N}}\left(-2\mathcal{R}_1 - \mathcal{R}_3 - \mathcal{R}_4 - \mathcal{R}_5\right),\qquad(2.69)$$

$$\omega_{\text{NO}} = M_{\text{NO}}\left(\mathcal{R}_3 - \mathcal{R}_4 + \mathcal{R}_5\right),\qquad(2.70)$$

$$\omega_{\text{O}_2} = M_{\text{O}_2}\left(\mathcal{R}_2 - \mathcal{R}_5\right),\qquad(2.71)$$

$$\omega_{\text{N}_2} = M_{\text{N}_2}\left(\mathcal{R}_1 + \mathcal{R}_4\right).\qquad(2.72)$$

One should notice that the sum of the species' source terms is zero being compatible with the conservation of the total mass.

2.1.5   Vibrational Energy Source and Relaxation Model

This section examines the energy exchange mechanism caused by elastic/inelastic collisions between the gas particles. Although various energy exchange processes, such as the translational-electron, translational-vibrational, and vibrational-vibrational energy exchanges are likely in the flow field, only the translational-vibrational energy exchanges are considered in this thesis, as in many studies in the literature (see, e.g., [103, 106, 107]). For further details, e.g., the energy contribution of collisions between electrons and neutral atoms, energy loss due to the ionization reactions, and energy transfer between the translational and electron modes, the interested reader can refer to [78, 112].

The vibrational-electron-electronic energy source term, ω_V, appearing in Eq. (2.23) accounts for the energy exchanges between the translational-rotational and the vibrational energy modes due to collisions, and for the vibrational energy exchanges due to the dissociation and recombination processes. This term can be given as follows [107]:

$$\omega_V = Q_v^{\text{chem}} + Q_{\text{transfer}}^{t-v}, \tag{2.73}$$

where the species average vibrational energy production rate due to chemical reactions and electrons is defined as [107]

$$Q_{v,s}^{\text{chem}} = \omega_s \left(e_s^{vib} + e_s^{elec} \right). \tag{2.74}$$

Here, the term ω_s represents the species chemical source term (see Section 2.1.4). Then, the net vibrational energy production rate arising from chemical reactions and electrons is as follows [107]:

$$Q_v^{\text{chem}} = \sum_s Q_{v,s}^{\text{chem}}. \tag{2.75}$$

The energy transport between the translational and vibrational energy modes can be given based on the *Landau–Teller model* [107]:

$$Q_{s,\text{transfer}}^{t-v} = \rho_s \frac{\hat{e}_s^{vib} - e_s^{vib}}{\tau_s^{vib}}, \tag{2.76}$$

where the term $\hat{e}_s^{vib}$ denotes the species equilibrium vibrational energy that is obtained by calculating at the translational-rotational temperature T, and τ_s^{vib} is the molar averaged species characteristic vibrational relaxation time that is given based on the *Millikan–White model* as follows [122, 138]:

$$\tau_s^{vib} = \frac{\sum_{r=1}^{N_s} \chi_r}{\sum_{r=1}^{N_s} \frac{\chi_r}{\tau_{sr}^{vib}}}. \tag{2.77}$$

The terms complementing Eq. (2.77) are given as follows [107]:

$$\tau_{sr}^{vib} = \frac{1}{p} \exp\left[A_{sr} \left(T^{-\frac{1}{3}} - 0.015 \mu_{sr}^{\frac{1}{4}} \right) - 18.42 \right], \tag{2.78}$$

$$A_{sr} = 1.16 \times 10^{-3} \mu_{sr}^{\frac{1}{2}} \theta_{v,s}^{\frac{4}{3}}, \tag{2.79}$$

$$\mu_{sr} = \frac{M_s M_r}{M_s + M_r}. \tag{2.80}$$

According to the Landau–Teller model, a molecule's vibrational energy level can change by only one quantum level at a time [200]. The term $\theta_{v,s}$ stands for the characteristic vibrational temperature of species s (see Appendix A), μ_{sr} denotes the reduced mass of colliding particles, and τ_{sr}^{vib} is the inter-species relaxation time. The pressure, p, is expressed in atmospheric pressure (atm). Since 1 atm is $101,325$ Pascal, the term τ_{sr}^{vib} is multiplied by $101,325$ to convert the pressure unit from atm to Pascal.

Eventually, the vibrational energy source term reads [107]:

$$\omega_V = \sum_s Q_{s,\text{transfer}}^{t-v} + \sum_s \omega_s \left(e_s^{vib} + e_s^{elec} \right). \tag{2.81}$$

For temperatures between $3,000$ K and $8,000$ K, the Millikan–White model can be exploited. The *Park's correction term* [152] should be added to the formulation for temperatures higher than $8,000$ K since the Millikan–White model yields unrealistically large relaxation rates [30] due to the over-predicted collision cross-sections.

2.2 Quasi-Linear Form of the Governing Equations

Due to the complexity of the governing equations presented in the preceding sections, it is more convenient to rewrite them as a coupled system in the following form (see,

for example, [34, 72, 103, 208]:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} = \frac{\partial \mathbf{E}_1}{\partial x_1} + \frac{\partial \mathbf{E}_2}{\partial x_2} + \mathbf{S}, \tag{2.82}$$

where the conservation variables are in $\mathbf{U}$, the inviscid fluxes are in $\mathbf{F}_1$ and $\mathbf{F}_2$, the viscous fluxes are in $\mathbf{E}_1$ and $\mathbf{E}_2$, and eventually, the source terms are in $\mathbf{S}$:

$$\mathbf{U} = \begin{bmatrix} \rho_O \\ \rho_N \\ \rho_{NO} \\ \rho_{O_2} \\ \rho_{N_2} \\ \rho u_1 \\ \rho u_2 \\ \rho e \\ \rho e_V \end{bmatrix}, \quad \mathbf{F}_1 = \begin{bmatrix} \rho_O u_1 \\ \rho_N u_1 \\ \rho_{NO} u_1 \\ \rho_{O_2} u_1 \\ \rho_{N_2} u_1 \\ \rho u_1^2 + p \\ \rho u_1 u_2 \\ \rho u_1 h \\ \rho u_1 e_V \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} \rho_O u_2 \\ \rho_N u_2 \\ \rho_{NO} u_2 \\ \rho_{O_2} u_2 \\ \rho_{N_2} u_2 \\ \rho u_1 u_2 \\ \rho u_2^2 + p \\ \rho u_2 h \\ \rho u_2 e_V, \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \omega_O \\ \omega_N \\ \omega_{NO} \\ \omega_{O_2} \\ \omega_{N_2} \\ 0 \\ 0 \\ 0 \\ \omega_V \end{bmatrix}, \tag{2.83}$$

$$\mathbf{E}_1 = \begin{bmatrix} \rho D_O \frac{\partial Y_O}{\partial x_1} \\ \rho D_N \frac{\partial Y_N}{\partial x_1} \\ \rho D_{NO} \frac{\partial Y_{NO}}{\partial x_1} \\ \rho D_{O_2} \frac{\partial Y_{O_2}}{\partial x_1} \\ \rho D_{N_2} \frac{\partial Y_{N_2}}{\partial x_1} \\ \sigma_{11} \\ \sigma_{12} \\ u_1 \sigma_{11} + u_2 \sigma_{12} + q_1 + \sum_s \rho D_s h_s \frac{\partial Y_s}{\partial x_1} \\ q_{V,1} + \sum_s \rho D_s e_{V,s} \frac{\partial Y_s}{\partial x_1} \end{bmatrix}, \tag{2.84}$$

35

$$\mathbf{E}_2 = \begin{bmatrix} \rho D_\mathrm{O} \frac{\partial Y_\mathrm{O}}{\partial x_2} \\ \rho D_\mathrm{N} \frac{\partial Y_\mathrm{N}}{\partial x_2} \\ \rho D_\mathrm{NO} \frac{\partial Y_\mathrm{NO}}{\partial x_2} \\ \rho D_\mathrm{O_2} \frac{\partial Y_\mathrm{O_2}}{\partial x_2} \\ \rho D_\mathrm{N_2} \frac{\partial Y_\mathrm{N_2}}{\partial x_2} \\ \sigma_{21} \\ \sigma_{22} \\ u_1\sigma_{21} + u_2\sigma_{22} + q_2 + \sum_s \rho D_s h_s \frac{\partial Y_s}{\partial x_2} \\ q_{V,2} + \sum_s \rho D_s e_{V,s} \frac{\partial Y_s}{\partial x_2} \end{bmatrix}. \tag{2.85}$$

The translational-rotational heat fluxes are given as

$$q_1 = k_{tr} \frac{\partial T}{\partial x_1} + k_v \frac{\partial T_V}{\partial x_1}, \tag{2.86}$$

$$q_2 = k_{tr} \frac{\partial T}{\partial x_2} + k_v \frac{\partial T_V}{\partial x_2}, \tag{2.87}$$

and the electron-electronic-vibrational heat fluxes are as

$$q_{V,1} = k_v \frac{\partial T_V}{\partial x_1}, \tag{2.88}$$

$$q_{V,2} = k_v \frac{\partial T_V}{\partial x_2}. \tag{2.89}$$

The numerical treatment of inviscid (advective) fluxes is crucial since they convey the essential information of wave transport. The inviscid fluxes can be expressed as follows:

$$\frac{\partial \mathbf{F}_1}{\partial x_1} = \frac{\partial \mathbf{F}_1}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial x_1} \tag{2.90}$$

and

$$\frac{\partial \mathbf{F}_2}{\partial x_2} = \frac{\partial \mathbf{F}_2}{\partial \mathbf{U}} \frac{\partial \mathbf{U}}{\partial x_2}. \tag{2.91}$$

Letting

$$\mathbf{A}_1 = \frac{\partial \mathbf{F}_1}{\partial \mathbf{U}} \tag{2.92}$$

and

$$\mathbf{A}_2 = \frac{\partial \mathbf{F}_2}{\partial \mathbf{U}}, \tag{2.93}$$

Eq. (2.82) can be recast as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_1 \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{A}_2 \frac{\partial \mathbf{U}}{\partial x_2} = \frac{\partial \mathbf{E}_1}{\partial x_1} + \frac{\partial \mathbf{E}_2}{\partial x_2} + \mathbf{S}. \tag{2.94}$$

Here, the advective Jacobians $\mathbf{A}_1$ and $\mathbf{A}_2$ have real eigenvalues and corresponding linearly independent eigenvectors. One can find more on the advective Jacobians in Appendix B.

Similarly, the viscous fluxes, $\mathbf{E}_1$ and $\mathbf{E}_2$, can also be expressed as follows:

$$\frac{\partial \mathbf{E}_i}{\partial x_i} = \frac{\partial}{\partial x_i} \left[\mathbf{K}_{ij} \frac{\partial \mathbf{U}}{\partial x_j} \right], \tag{2.95}$$

where the positive semi-definite matrices $\mathbf{K}_{ij}$ are called *diffusivity matrices*. Explicitly,

$$\mathbf{E}_1 = \mathbf{K}_{11} \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{K}_{12} \frac{\partial \mathbf{U}}{\partial x_2}, \tag{2.96}$$

$$\mathbf{E}_2 = \mathbf{K}_{21} \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{K}_{22} \frac{\partial \mathbf{U}}{\partial x_2}. \tag{2.97}$$

Consequently, Eq. (2.82), and equivalently Eq. (2.94), can be recast as

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_1 \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{A}_2 \frac{\partial \mathbf{U}}{\partial x_2} = \frac{\partial}{\partial x_1} \left[\mathbf{K}_{11} \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{K}_{12} \frac{\partial \mathbf{U}}{\partial x_2} \right]$$
$$+ \frac{\partial}{\partial x_2} \left[\mathbf{K}_{21} \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{K}_{22} \frac{\partial \mathbf{U}}{\partial x_2} \right] + \mathbf{S}. \tag{2.98}$$

The advective Jacobians $\mathbf{A}_1$ and $\mathbf{A}_2$ are given explicitly by the matrices in Eq. (2.99) and Eq. (2.100).

$$
\mathbf{A}_1 =
\begin{bmatrix}
(1-Y_\text{O})u_1 & -Y_\text{O}u_1 & -Y_\text{O}u_1 & -Y_\text{O}u_1 & -Y_\text{O}u_1 & Y_\text{O} & 0 & 0 & 0 \\
-Y_\text{N}u_1 & (1-Y_\text{N})u_1 & -Y_\text{N}u_1 & -Y_\text{N}u_1 & -Y_\text{N}u_1 & Y_\text{N} & 0 & 0 & 0 \\
-Y_\text{NO}u_1 & -Y_\text{NO}u_1 & (1-Y_\text{NO})u_1 & -Y_\text{NO}u_1 & -Y_\text{NO}u_1 & Y_\text{NO} & 0 & 0 & 0 \\
-Y_{\text{O}_2}u_1 & -Y_{\text{O}_2}u_1 & -Y_{\text{O}_2}u_1 & (1-Y_{\text{O}_2})u_1 & -Y_{\text{O}_2}u_1 & Y_{\text{O}_2} & 0 & 0 & 0 \\
-Y_{\text{N}_2}u_1 & -Y_{\text{N}_2}u_1 & -Y_{\text{N}_2}u_1 & -Y_{\text{N}_2}u_1 & (1-Y_{\text{N}_2})u_1 & Y_{\text{N}_2} & 0 & 0 & 0 \\
\gamma_\text{O}-u_1^2 & \gamma_\text{N}-u_1^2 & \gamma_\text{NO}-u_1^2 & \gamma_{\text{O}_2}-u_1^2 & \gamma_{\text{N}_2}-u_1^2 & u_1(2-\beta) & -\beta u_2 & \beta & \Phi \\
-u_1u_2 & -u_1u_2 & -u_1u_2 & -u_1u_2 & -u_1u_2 & u_2 & u_1 & 0 & 0 \\
u_1(\gamma_\text{O}-h) & u_1(\gamma_\text{N}-h) & u_1(\gamma_\text{NO}-h) & u_1(\gamma_{\text{O}_2}-h) & u_1(\gamma_{\text{N}_2}-h) & h-\beta u_1^2 & -\beta u_1u_2 & u_1(\beta+1) & \Phi u_1 \\
-u_1e_V & -u_1e_V & -u_1e_V & -u_1e_V & -u_1e_V & e_V & 0 & 0 & u_1
\end{bmatrix}, \tag{2.99}
$$

$$
\mathbf{A}_2 =
\begin{bmatrix}
(1-Y_\text{O})u_2 & -Y_\text{O}u_2 & -Y_\text{O}u_2 & -Y_\text{O}u_2 & -Y_\text{O}u_2 & 0 & Y_\text{O} & 0 & 0 \\
-Y_\text{N}u_2 & (1-Y_\text{N})u_2 & -Y_\text{N}u_2 & -Y_\text{N}u_2 & -Y_\text{N}u_2 & 0 & Y_\text{N} & 0 & 0 \\
-Y_\text{NO}u_2 & -Y_\text{NO}u_2 & (1-Y_\text{NO})u_2 & -c_\text{NO}u_2 & -Y_\text{NO}u_2 & 0 & Y_\text{NO} & 0 & 0 \\
-Y_{\text{O}_2}u_2 & -Y_{\text{O}_2}u_2 & -Y_{\text{O}_2}u_2 & (1-Y_{\text{O}_2})u_2 & -Y_{\text{O}_2}u_2 & 0 & Y_{\text{O}_2} & 0 & 0 \\
-Y_{\text{N}_2}u_2 & -Y_{\text{N}_2}u_2 & -Y_{\text{N}_2}u_2 & -Y_{\text{N}_2}u_2 & (1-Y_{\text{N}_2})u_2 & 0 & Y_{\text{N}_2} & 0 & 0 \\
-u_1u_2 & -u_1u_2 & -u_1u_2 & -u_1u_2 & -u_1u_2 & u_2 & u_1 & 0 & 0 \\
\gamma_\text{O}-u_2^2 & \gamma_\text{N}-u_2^2 & \gamma_\text{NO}-u_2^2 & \gamma_{\text{O}_2}-u_2^2 & \gamma_{\text{N}_2}-u_2^2 & -\beta u_1 & u_2(2-\beta) & \beta & \Phi \\
u_2(\gamma_\text{O}-h) & u_2(\gamma_\text{N}-h) & u_2(\gamma_\text{NO}-h) & u_2(\gamma_{\text{O}_2}-h) & u_2(\gamma_{\text{N}_2}-h) & -\beta u_1u_2 & h-\beta u_2^2 & u_2(\beta+1) & \Phi u_2 \\
-u_2e_V & -u_2e_V & -u_2e_V & -u_2e_V & -u_2e_V & 0 & e_V & 0 & u_2
\end{bmatrix}. \tag{2.100}
$$

The terms appearing in matrices given in Eqs. (2.99)–(2.100) are defined as follows [72]:

$$\beta = \frac{\partial p}{\partial \rho e}, \tag{2.101}$$

$$\Phi = \frac{\partial p}{\partial \rho e_V} = \frac{R_u}{\rho C_{vib,v}} \frac{\rho_e}{M_e} - \beta, \tag{2.102}$$

$$\gamma_s = \frac{\partial p}{\partial \rho_s} = \frac{R_u T_q}{M_s} + \beta \frac{\|\mathbf{u}\|^2}{2} - \beta e_s - \Phi e_{V,s}, \tag{2.103}$$

where the term $C_{vib,v}$ is the vibrational specific heat of the mixture at constant volume:

$$C_{vib,v} = \sum_s Y_s C_{vib,v,s}. \tag{2.104}$$

The term T_q is equal to T_V for electrons, otherwise, $T_q = T$. The frozen speed of sound, c_{sound}, is given by the following relation [72]:

$$c_{\text{sound}}^2 = \sum_s Y_s \gamma_s + \beta \left[h - \left(u_1^2 + u_2^2 \right) \right] + \Phi e_V = (1 + \beta) \frac{p}{\rho}. \tag{2.105}$$

Note that Eq. (2.102) reduces to

$$\Phi = -\beta, \tag{2.106}$$

for a non-ionized gas mixture.

2.3 Governing Equations of One-Species Inviscid Hypersonic Flow

Consider a spatial domain $\Omega \subset \mathbb{R}^{n_{\text{sd}}}$ with boundary Γ over a time interval $I_t = (0, t_{\text{f}}]$, where n_{sd} denotes the spatial dimension, and t_{f} is the final time. Euler equations of compressible flows can be given as (see, for example, [44, 193])

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \qquad \text{in } \Omega \times I_t, \tag{2.107}$$

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = \mathbf{0} \qquad \text{in } \Omega \times I_t, \tag{2.108}$$

$$\frac{\partial (\rho e)}{\partial t} + \nabla \cdot (\rho \mathbf{u} h) = 0 \qquad \text{in } \Omega \times I_t. \tag{2.109}$$

The governing equations given by Eqs. (2.107)–(2.109) can be written in a more compact way as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} = \mathbf{0}, \tag{2.110}$$

39

where the vector of the conservation variables $\mathbf{U}$, and the inviscid fluxes $\mathbf{F}_i$ are

$$
\mathbf{U} = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho e \end{bmatrix}, \quad \mathbf{F}_i = \begin{bmatrix} \rho u_i \\ \rho u_i u_1 + \delta_{i1} p \\ \rho u_i u_2 + \delta_{i2} p \\ \rho u_i h \end{bmatrix}. \tag{2.111}
$$

Here, the δ_{ij} are the components of the identity matrix $\mathbf{I}$. Consider the quasi-linear form of Eq. (2.110) given as follows:

$$
\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_1 \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{A}_2 \frac{\partial \mathbf{U}}{\partial x_2} = \mathbf{0}, \tag{2.112}
$$

where the advective Jacobians are explicitly written as

$$
\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ (\gamma-1)\|\mathbf{u}\|^2/2 - u_1^2 & (3-\gamma)u_1 & (1-\gamma)u_2 & \gamma-1 \\ -u_1 u_2 & u_2 & u_1 & 0 \\ (\gamma-1)\|\mathbf{u}\|^2 u_1/2 - u_1 h & h - (\gamma-1)u_1^2 & (1-\gamma)u_1 u_2 & \gamma u_1 \end{bmatrix} \tag{2.113}
$$

and

$$
\mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -u_1 u_2 & u_2 & u_1 & 0 \\ (\gamma-1)\|\mathbf{u}\|^2/2 - u_2^2 & (1-\gamma)u_1 & (3-\gamma)u_2 & \gamma-1 \\ (\gamma-1)\|\mathbf{u}\|^2 u_2/2 - u_2 h & (1-\gamma)u_1 u_2 & h - (\gamma-1)u_2^2 & \gamma u_2 \end{bmatrix}, \tag{2.114}
$$

where the total enthalpy, h, is defined by Eq. (2.22).

The boundary and initial conditions associated with Eq. (2.112) are in the form of $\mathbf{U}(\mathbf{x}, t) = \mathbf{G}$ for an inflow boundary, $\mathbf{u} \cdot \mathbf{n} = 0$ for a slip surface, and $\mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0(\mathbf{x})$, where $\mathbf{G}$ and $\mathbf{U}_0$ are given functions. Here, $\mathbf{n}$ is the unit normal vector, $\mathbf{u}$ is the velocity vector, and $\|\cdot\|$ is the standard Euclidean norm.

2.4 Governing Equations of Five-Species Inviscid Hypersonic Flow

Considering similar spatial and temporal domains given for one-species Euler equations (2.107)–(2.109), five-species Euler equations of thermochemical nonequilibrium

flows can be given as follows (see, for example, [103, 208]):

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \mathbf{u}) = \omega_s \qquad \text{in } \Omega \times I_t, \qquad (2.115)$$

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = \mathbf{0} \qquad \text{in } \Omega \times I_t, \qquad (2.116)$$

$$\frac{\partial (\rho e)}{\partial t} + \nabla \cdot (\rho \mathbf{u} h) = 0 \qquad \text{in } \Omega \times I_t, \qquad (2.117)$$

$$\frac{\partial (\rho e_V)}{\partial t} + \nabla \cdot (\rho \mathbf{u} e_V) = \omega_V \qquad \text{in } \Omega \times I_t, \qquad (2.118)$$

where the terms appearing in the system are introduced in the previous sections. A quasi-linear form of Eqs. (2.115)–(2.118) reads

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} = \mathbf{S}, \qquad (2.119)$$

where the vectors are given as

$$\mathbf{U} = \begin{bmatrix} \rho_O \\ \rho_N \\ \rho_{NO} \\ \rho_{O_2} \\ \rho_{N_2} \\ \rho u_1 \\ \rho u_2 \\ \rho e \\ \rho e_V \end{bmatrix}, \quad \mathbf{F}_1 = \begin{bmatrix} \rho_O u_1 \\ \rho_N u_1 \\ \rho_{NO} u_1 \\ \rho_{O_2} u_1 \\ \rho_{N_2} u_1 \\ \rho u_1^2 + p \\ \rho u_1 u_2 \\ \rho u_1 h \\ \rho u_1 e_V \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} \rho_O u_2 \\ \rho_N u_2 \\ \rho_{NO} u_2 \\ \rho_{O_2} u_2 \\ \rho_{N_2} u_2 \\ \rho u_1 u_2 \\ \rho u_2^2 + p \\ \rho u_2 h \\ \rho u_2 e_V \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \omega_O \\ \omega_N \\ \omega_{NO} \\ \omega_{O_2} \\ \omega_{N_2} \\ 0 \\ 0 \\ 0 \\ \omega_V \end{bmatrix}. \qquad (2.120)$$

A quasi-linear form of Eq. (2.119) reads

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_1 \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{A}_2 \frac{\partial \mathbf{U}}{\partial x_2} = \mathbf{S}. \qquad (2.121)$$

where the advective Jacobians are defined by Eqs. (2.99)–(2.100). The species source terms, ω_s, and the vibrational-electronic source term, ω_V, are computed as explained in Sections 2.1.3–2.1.5.

The boundary and initial conditions associated with Eq. (2.121) are similar to those given for Eq. (2.112) and will be discussed in detail in Chapter 4.

Although the existence and uniqueness of solutions are essential concepts in studying all kinds of PDEs, they are even more vital for (hyperbolic) conservation equations

(laws) governed by PDEs since they rarely have classical solutions [58]. If they have classical solutions, then the question of whether this solution is unique arises naturally. To this end, Appendix C is devoted to reviewing the theory on the existence and uniqueness of the solutions in the continuous setting for conservation equations briefly.

CHAPTER 3

THE STREAMLINE-UPWIND/PETROV–GALERKIN FORMULATION

In this chapter, a general framework for the stabilized finite element formulations and shock-capturing techniques is sketched first. Then, the historical development of the Streamline-Upwind/Petrov–Galerkin (SUPG) method and its role in high-speed flow research are investigated. Finally, a comprehensive set of 2D test computations is provided to demonstrate the importance of stabilized formulations and shock-capturing techniques in solving convection-dominated problems.

3.1   General Framework

The numerical methods employed for simulating CFD problems are usually determined according to whether the problem under consideration is convection-dominated. In the world of finite elements, since the convection operators are not self-adjoint, the (stiffness) matrices associated with the convection terms are not symmetric. Therefore, the "best approximation" property does not hold in the presence of the convection terms. Besides, the standard methods may also cause violation of basic physical principles such as the entropy condition and the positive definiteness of species concentrations (see Section 1.4). Hence the classical discretization methods often lead to instability problems (spurious oscillations) in the simulation of convection-dominated flows, and stabilized formulations are strictly needed. Contrary to popular opinion, such non-physical oscillations can occur not only in convection-dominated problems but also in reaction-dominated problems with strong gradients due to high reaction rates [190].

Stabilized finite element formulations are obtained by adding mesh-dependent stabilization term(s) into the standard (Bubnov–) Galerkin finite element formulation, in general. The common idea underlying these methods is to add the products of appropriate perturbation terms and residuals to the standard Galerkin formulation. The primary goal is to increase stability without degrading accuracy [85]. Therefore, in stabilized formulations, the correct determination of the stabilization parameter(s), conventionally denoted by "τ" in its scalar form and by "$\boldsymbol{\tau}$" in matrix form, plays a crucial role. These parameters should be determined in a way that a good compromise between accuracy and stability is achieved [79].

There have been numerous studies on efficient and correct determination of stabilization parameters. Although many studies have been devoted to the subject from computational and theoretical perspectives, the challenge of determining appropriate and optimal stabilization parameters is still an active area of research today. In the following parts of the thesis, we restrict our attention to stabilized formulations in terms of the Streamline-Upwind/Petrov–Galerkin (SUPG) method. For more on various stabilization techniques, the interested readers are referred to the studies [33, 43, 64, 83, 89, 92–96, 104, 177, 189] and the references therein. Besides, one can also refer to [27, 156] for further discussions on the need for stabilized formulations.

A numerical scheme is called *monotone* if the solution generated retains the sign of the previous time-step at all nodes of the spatial discretization. If it only preserves the monotonicity of the initial data, it is called *monotonicity-preserving* [42]. According to the "Godunov's order barrier theorem," proved in [73], the class of higher-order accurate linear difference methods cannot be monotone, and the SUPG method falls into that category [85]. Since the SUPG is neither monotone nor monotonicity-preserving, the stabilized approximations may still contain spurious local oscillations near sharp gradients [42].

Shock-capturing techniques aim to introduce some artificial (numerical) diffusion in the crosswind direction, where rapid and abrupt changes in the numerical solution occur; in comparison, the SUPG formulation introduces numerical diffusion in the streamline direction. The interested reader may also refer to [42, 87, 158, 177, 190] for more on shock-capturing operators.

Consequently, while the SUPG alone is not a sufficiently robust method (in capturing spurious oscillations localized in narrow layers) for shock wave phenomena in fluids, when it is supplemented with the so-called "shock-capturing" operators, it has proven to be an industrial-strength technology [36].

3.2 A Brief History of the SUPG

The SUPG formulation was first introduced for simulating incompressible flows by Hughes and Brooks in a 1979 ASME paper [84]. The work was later published as a journal article by the same authors [28] with further details in 1982. One can refer to [181] and the material therein for more on the stabilized formulations for computing incompressible flows.

Subsequently, again in 1982, Tezduyar and Hughes introduced the compressible-flow SUPG formulation in the context of the conservation variables in a NASA technical report [187]. The authors published the work with further examples as an AIAA paper [188], and then as a journal paper in [90]. Later on, various compressible-flow methods similar to the SUPG were developed, such as the Taylor–Galerkin method [54] and the streamline-diffusion method [96]. The compressible-flow SUPG formulation introduced in [90, 187, 188] is now called "$(SUPG)_{82}$," and the class of the stabilization parameters used is denoted by "τ_{82}."

Although promising results were obtained with the $(SUPG)_{82}$, the test computations revealed that the formulation needed to be improved near the regions where shocks occur. Thus, the idea of adding shock-capturing term(s) to the stabilized formulations was emerged to achieve better solution profiles at shocks. The SUPG formulation was supplemented with a shock-capturing term for simulating compressible flows for the first time in [88] by Hughes et al. in 1986. In 1987, Hughes et al. [86] reformulated the $(SUPG)_{82}$ in terms of the entropy variables, and supplemented with a shock-capturing term. Later, in 1991, Le Beau and Tezduyar [121], and then, in 1993, Le Beau et al. [120] reformulated the $(SUPG)_{82}$ in terms of the conservation variables adding a shock-capturing term very similar to one introduced in [86]. The shock-capturing parameter introduced in [121] in 1991 is today called "δ_{91}."

45

The SUPG-stabilized formulation for solving advection-diffusion-reaction equations introduced in [190] included a shock-capturing term and a stabilization parameter that took into account the interaction between the shock-capturing and SUPG terms. Thus, the shock-capturing term does not increase the SUPG effect when the advection and shock directions coincide. In [121], the stabilization parameter τ_{82} was slightly modified. Although the definition of (SUPG)$_{82}$ underwent some minor modifications in subsequent years, they were still used with the same shock-capturing parameter, δ_{91}, until 2004.

In 2004, new ways of determining the stabilization and shock-capturing parameters in the (SUPG)$_{82}$ were introduced in [183–185]. These new stabilization parameters are today called "τ_{04}," and they are in matrix form for viscous flows and are reduced to scalar for inviscid flows. The new shock-capturing parameters defined can be divided into two categories: in a style the discontinuity-capturing directional dissipation (DCDD) [158, 182, 185], and through the residual-based YZβ [183–185] shock-capturing.

In the remaining parts of this thesis, we restrict our attention only to the YZβ shock-capturing technique because it is one of the major components of the computations performed. It is simpler to calculate the YZβ shock-capturing parameter than δ_{91}. Besides, the parameter β of YZβ offers options for mild and sharp shocks. It was also reported in [16, 159, 191–193] that the YZβ parameter yields more accurate results than δ_{91} does. The details are provided in the following parts of the thesis (see Section 3.4 and Section 4.1).

In recent years, many new developments have been published in determining the stabilization parameters and element length scales. The interested reader may refer to [113, 148, 175–178, 196] for further details.

3.3 The SUPG in High-Speed Flow Research

For the last four decades, the methods used in computational fluid dynamics (CFD) reported in the literature have mostly been stabilized methods. Since the classical discretization methods result in numerical instabilities in simulating flows at high

Reynolds and Mach numbers, as well as in the presence of shocks, stabilization is required. The following paragraphs summarize the SUPG literature for computing high-speed flows briefly.

Chalot et al. [34] presented advection-diffusion systems for computing hypersonic flows and their symmetrization in terms of the entropy variables taking into account high-temperature effects as a result of their study [86], in which they developed and analyzed a space-time finite element formulation for linear symmetric advection-diffusion systems. Mallet et al. performed hypersonic computations on the geometry of the Hermes vehicle in [134]. The compressible-flow SUPG formulation was employed for computing flows in thermochemical nonequilibrium by Chalot et al. in [35]. The compressible-flow SUPG methods in terms of the conservation and entropy variables were compared in simulating subsonic, transonic, and supersonic flows in [120], with the results obtained being almost the same. The compressible-flow SUPG method of [121] was employed in [180, 186] in parallel computations.

In [203], Wienken et al. introduced a N–S solver supplemented with the SUPG formulation for simulating laminar and turbulent large-eddy simulations (LES). The authors considered a turbulent channel flow and a transonic flow past a sphere at Mach 1.53. In [159, 191–193], Tezduyar et al. computed inviscid supersonic flows employing the compressible-flow SUPG formulation supplemented with the YZβ shock-capturing. Kirk et al. analyzed the effects of several types of stabilization parameters used in the SUPG formulations for computing hypersonic flows in [106]. A SUPG formulation was employed for simulating nonequilibrium hypersonic flows around a cylinder and 3D blunt-nosed configurations by Bova et al. in [107].

Bova and Kirk analyzed the effects of artificial diffusion introduced in the SUPG formulation for computing Euler and N–S equations in [25]. Kirk and Oliver investigated the validity of the SUPG formulation for shockwave/turbulent boundary layer interaction problems in [108]. In [82], Howard utilized a SUPG-stabilized formulation for high-speed aerothermoelastic systems. Kirk et al. [109] employed a SUPG formulation for investigating aerothermal dynamics of atmospheric entry vehicles with thermal protection systems (TPS). They considered the Favre-averaged N–S equations coupled with the Spalart–Allmaras turbulence model. In [8], Anderson et al. implemented a

SUPG formulation into the FUN3D, a CFD suite of tools actively developed at NASA Langley Research Center. The authors employed a SUPG formulation augmented with a shock-capturing technique to solve problems ranging from 1D to 3D flows and supersonic to hypersonic regimes in [44]. Most recently, Rajanna et al. [157] employed a compressible-flow SUPG formulation for hypersonic flows governed by N–S equations.

In this present thesis, the compressible-flow SUPG formulation is used in combination with the YZβ shock-capturing, i.e., SUPG-YZβ, for simulating high-speed flows in thermochemical equilibrium/nonequilibrium ranging from Mach 2 to Mach 8 in the light of studies [191–193] by Tezduyar et al. (see Chapters 4–5).

3.4 Illustrative Computations with the SUPG Formulation

In this section, a comprehensive set of test computations is presented to demonstrate the implementation and effectiveness of the SUPG formulation supplemented with the YZβ shock-capturing, SUPG-YZβ, prior to hypersonic flow simulations performed in Chapter 5. For theoretical aspects of each problem, one can refer to the provided references and the material therein.

For all computations performed here, the relative error tolerances for Newton–Raphson iterations are set to 1.0×10^{-10}. Since the systems of equations are not large, a direct method, i.e., sparse LU decomposition, is employed for solving linear systems. For more on the solution of algebraic equation systems, the interested reader can refer to Section 4.6.

3.4.1 A Convection-Dominated Linear Convection-Diffusion Equation

Consider the following singularly perturbed steady-state convection-diffusion equation [210]:

$$-\nabla \cdot (\varepsilon \nabla u) + \mathbf{b} \cdot \nabla u = f \qquad \text{in } \Omega = (0,1)^2, \qquad (3.1)$$

$$u = g^D \qquad \text{on } \partial\Omega, \qquad (3.2)$$

where the source term, $f = f(x_1, x_2)$, is given as

$$f(x_1, x_2) = (x_1 + x_2)\left(1 - \exp\left(\frac{x_1 - 1}{\varepsilon}\right)\exp\left(\frac{x_2 - 1}{\varepsilon}\right)\right)$$
$$+ (x_1 - x_2)\left(\exp\left(\frac{x_2 - 1}{\varepsilon}\right) - \exp\left(\frac{x_1 - 1}{\varepsilon}\right)\right). \quad (3.3)$$

The term ε is the diffusion parameter (frequently called the singular perturbation parameter) arising from the nature of the problem under consideration, with $0 < \varepsilon \ll 1$. The Dirichlet boundary condition associated with Eq. (3.1), specified by Eq. (3.2), is given as $g^D = g^D(x_1, x_2) = 0$. Then, the exact solution to Eqs. (3.1)–(3.2) is [210]:

$$u(x_1, x_2) = x_1 x_2 \left(1 - \exp\left(\frac{x_1 - 1}{\varepsilon}\right)\right)\left(1 - \exp\left(\frac{x_2 - 1}{\varepsilon}\right)\right). \quad (3.4)$$

The discrete SUPG formulation augmented with shock-capturing can be given as follows: find $u^h \in \mathcal{S}^h$ such that for all test functions $w^h \in \mathcal{V}^h$,

$$\int_\Omega w^h \left[-\nabla \cdot \left(\varepsilon \nabla u^h\right) + \mathbf{b}^h \cdot \nabla u^h - f^h\right] \, d\Omega$$
$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \tau_{\text{SUPG}} \left(-\nabla \cdot \left(\varepsilon \nabla u^h\right) + \mathbf{b}^h \cdot \nabla u^h - f^h\right) \left(\mathbf{b}^h \cdot \nabla w^h\right) \, d\Omega$$
$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \nu_{\text{SHOC}} \left(\nabla u^h \cdot \nabla w^h\right) \, d\Omega = 0. \quad (3.5)$$

The solution and test function spaces are defined as follows:

$$\mathcal{S}^h = \{u^h \,|\, u^h \in H^{h1}(\Omega), \, u^h = g^D \text{ on } \Gamma^D\}, \quad (3.6)$$
$$\mathcal{V}^h = \{w^h \,|\, w^h \in H^{h1}(\Omega), \, w^h = 0 \text{ on } \Gamma^D\}. \quad (3.7)$$

Here, the finite-dimensional space $H^{1h}(\Omega)$ is defined as

$$H^{h1}(\Omega) = \{\Phi^h | \Phi^h \in \mathcal{C}^0\left(\overline{\Omega}\right), \Phi^h|_{\Omega^e} \in \mathcal{P}_1(\Omega^e), \forall \Omega^e \in \mathcal{T}^h\}, \quad (3.8)$$

where $\mathcal{T}^h$ is the set of elements arising from the finite element discretization of the computational domain Ω, $\mathcal{C}^0\left(\overline{\Omega}\right)$ denotes the class of all continuous functions defined on the closure of Ω, and the space $\mathcal{P}_1$ represents the set of linear polynomials. It is also assumed that $\overline{\Omega} = \bigcup_{\Omega^e \in \mathcal{T}^h} \overline{\Omega^e}$. The term Γ^D represents the part of boundary $\Gamma = \partial\Omega$ where Dirichlet-type boundary conditions apply. If Γ^N corresponds to the part of the boundary Γ where Neumann-type boundary conditions are prescribed, then it is clear that $\Gamma = \overline{\Gamma^D \cup \Gamma^N}$ and $\Gamma^D \cap \Gamma^N = \emptyset$.

The first line in Eq. (3.5) is from the standard Galerkin finite element formulation, the second one is for stabilization, and the third is for shock-capturing. In this formulation, the convection (advection) vector is defined as $\mathbf{b} = [b_1, b_2]^T = [1, 1]^T$, and n_{el} represents the number of triangular elements. The terms τ_{SUPG} and ν_{SHOC} are the stabilization and shock-capturing parameters, respectively. How these parameters are determined directly affects the accuracy of approximations. Here, we use the stabilization parameter τ_{SUPG}, which was introduced in a Ph.D. thesis [167] first and later analyzed in a journal paper [168]:

$$\tau_{\text{SUPG}} = \left[\left(\frac{2\|\mathbf{b}^h\|}{h^e} \right)^2 + \left(\frac{4\varepsilon}{(h^e)^2} \right)^2 \right]^{-\frac{1}{2}}, \tag{3.9}$$

where h^e is the smallest edge length associated with element e [140]. The superscript "h" indicates that the function is coming from a finite-dimensional space, and the norm $\|\cdot\|$ refers to the standard Euclidean norm. For time-dependent problems, Eq. (3.9) is modified as follows [167]:

$$\tau_{\text{SUPG}} = \left[\left(\frac{2}{\varDelta t} \right)^2 + \left(\frac{2\|\mathbf{b}^h\|}{h^e} \right)^2 + \left(\frac{4\varepsilon}{(h^e)^2} \right)^2 \right]^{-\frac{1}{2}}, \tag{3.10}$$

where the term $\varDelta t$ represents the time-step length.

The shock-capturing parameter, ν_{SHOC}, is modified from [191–193] by replacing the density ρ with the unknown solution u, in the term $\mathbf{j}$ (see Eqs. (3.15)–(3.16)), and can be written as

$$\nu_{\text{SHOC}} = |\mathbf{Y}^{-1}\mathbf{Z}| \left(\mathbf{Y}^{-1} \sum_{i=1}^{n_{\text{sd}}} \left| \frac{\partial u^h}{\partial x_i} \right|^2 \right)^{\frac{\beta}{2}-1} \left(\frac{h_{\text{SHOC}}}{2} \right)^{\beta}, \tag{3.11}$$

where n_{sd} denotes the number of space dimensions, and Y is a scaling parameter. Note that for coupled problems, the parameter Y becomes a matrix. In our computations, we set $\beta = 2$:

$$\nu_{\text{SHOC}} = |\mathbf{Y}^{-1}\mathbf{Z}| \left(\frac{h_{\text{SHOC}}}{2} \right)^2, \tag{3.12}$$

since the problems we deal with are highly dominated by convection. The quantity Z is given as

$$\mathbf{Z} = b_1^h \frac{\partial u^h}{\partial x_1} + b_2^h \frac{\partial u^h}{\partial x_2} - f^h = \mathbf{b}^h \cdot \nabla u^h - f^h. \tag{3.13}$$

For time-dependent problems, Eq. (3.13) can be modified as

$$\mathbf{Z} = \frac{\partial u^h}{\partial t} + b_1^h \frac{\partial u^h}{\partial x_1} + b_2^h \frac{\partial u^h}{\partial x_2} - f^h = \frac{\partial u^h}{\partial t} + \mathbf{b}^h \cdot \nabla u^h - f^h. \tag{3.14}$$
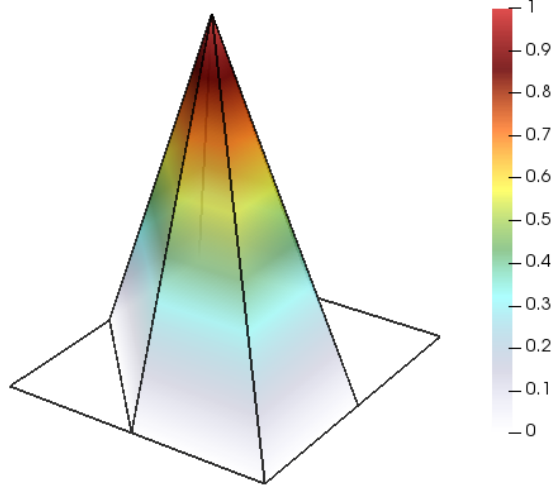
Figure 3.1: A 2D linear shape function on a generic triangular mesh.

The element length scale, h_{SHOC}, and the unit vector $\mathbf{j}$ in the direction of ∇u^h, are defined as follows:

$$h_{\text{SHOC}} = 2 \left(\sum_{a=1}^{n_{\text{en}}} |\mathbf{j} \cdot \nabla N_a| \right)^{-1}, \tag{3.15}$$

$$\mathbf{j} = \frac{\nabla u^h}{\|\nabla u^h\|}, \tag{3.16}$$

where the norm given in Eq. (3.16) is the standard Euclidean norm, and N_a denotes the shape (interpolation, basis, hat) function associated with element node a. Without loss of generality, the 2D linear shape functions associated with a triangular element e and its three nodes can be defined as follows:

$$N_i^{(e)}(x_1, x_2) = \frac{1}{2A^{(e)}} \left[a_i^{(e)} + b_i^{(e)} x_1 + c_i^{(e)} x_2 \right], \quad i = 1, 2, 3, \tag{3.17}$$

where $A^{(e)}$ is the area of element e. Note that Eq. (3.17) should satisfy the following condition [116]:

$$N_i^{(e)}\left(x_1^{(j)}, x_2^{(j)}\right) = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j, \end{cases} \tag{3.18}$$

where the point $\left(x_1^{(j)}, x_2^{(j)}\right)$ represents the coordinate of the jth node of element e, and $N_i^{(e)}$ is the shape function associated with the ith node of element e. Then, the constants $a_i^{(e)}, b_i^{(e)}, c_i^{(e)}$, given in Eq. (3.17), can be determined uniquely. Figure 3.1 illustrates an example 2D linear shape function associated with an arbitrary node on a generic triangular mesh. For more on the shape functions, one can refer to [116].

51

Notice that since the stabilization terms and shock-capturing parameter include the residual of Eq. (3.1) as a factor, these added terms vanish when the exact solution is substituted into the SUPG-YZβ formulation given by Eq. (3.5).

The computations for solving Eqs. (3.1)–(3.2) are performed on two different meshes of 625 nodes and $1,152$ elements, and $1,369$ nodes and $2,592$ elements (see Figure 3.2). The scaling term Y of YZβ is set as Y= 1.0. The numbers of Newton–Raphson iterations in SUPG-YZβ computations are 5 and 4 for meshes with $1,152$ elements and $2,592$ elements, respectively.

(a) (b)
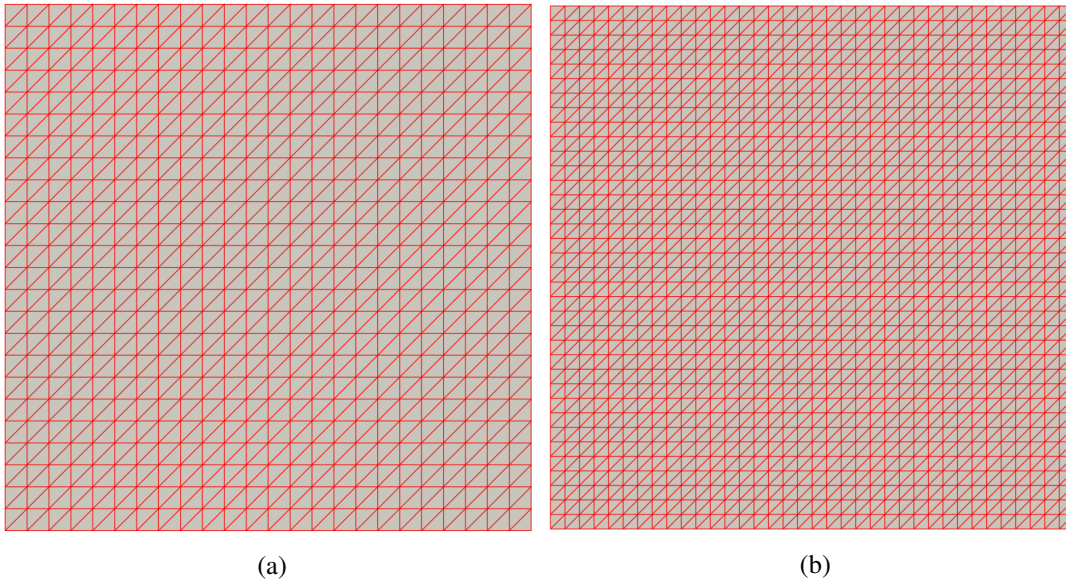
Figure 3.2: Meshes used for solving Eqs. (3.1)–(3.2): (a) $n_{en} = 625$ and $n_{el} = 1,152$, and (b) $n_{en} = 1,369$ and $n_{el} = 2,592$.

The GFEM, SUPG, and SUPG-SC (the SUPG formulation supplemented with YZβ shock-capturing, i.e., SUPG-YZβ) approximations to the solution of Eqs. (3.1)–(3.2) are shown in Figure 3.3, Figure 3.4, and Figure 3.5, respectively. As can be seen from these elevation plots, the GFEM solutions involve spurious oscillations, especially as the problem becomes more convection dominated, particularly for the values of ε less than 1.0×10^{-3} (see Figure 3.3). The approximations obtained with the SUPG formulation do not contain globally-propagated non-physical oscillations, unlike the GFEM yields (see Figure 3.4). However, it is clearly observed that the SUPG-stabilized formulation needs special treatment for resolving steep gradients. As can be seen in Figure 3.5, when the SUPG formulation is augmented with YZβ shock-capturing, sharp gradients are captured successfully. In Figure 3.6, the SUPG and SUPG-YZβ

approximations are compared for $\varepsilon = 10^{-8}$, and corresponding absolute errors are presented.
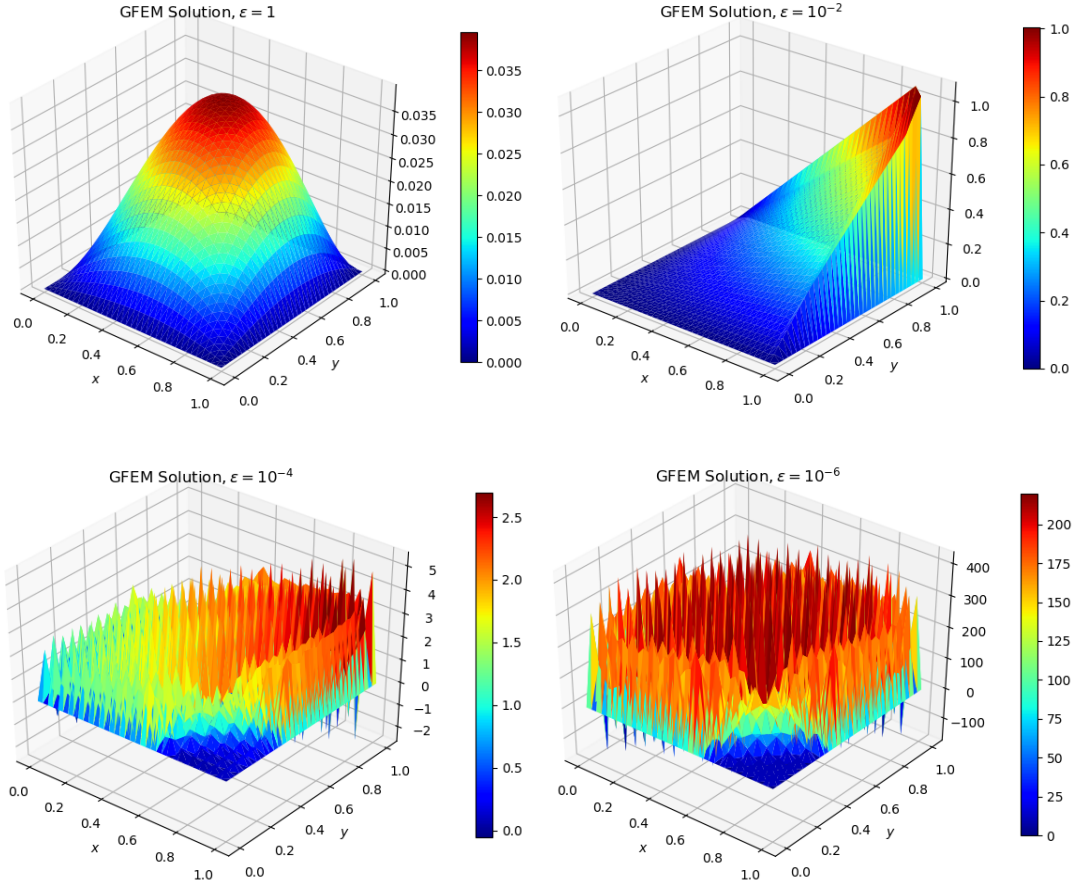
Figure 3.3: GFEM solutions of Eqs. (3.1)–(3.2) for various values of ε; $n_{\mathrm{el}} = 2,592$.

In computations, three different norms, L^2 norm $\|\cdot\|_{L^2}$, maximum norm $\|\cdot\|_{L^\infty}$, and ε-weighted (standard) energy norm [29, 160, 161] $\|\cdot\|_{L^\varepsilon}$ are used to measure the errors in approximations obtained by the proposed formulations. These norms are defined as follows:

$$\|v\|_{L^2} = \sqrt{\int_\Omega v^2 dx}, \tag{3.19}$$

$$\|v\|_{L^\infty} = \max_i |v_i|, \tag{3.20}$$

$$\|v\|_{L^\varepsilon} = \sqrt{\varepsilon \|\nabla v\|_{L^2}^2 + \|v\|_{L^2}^2}. \tag{3.21}$$

The convergence rate, ρ^N, can be also calculated in the sense of the *double-mesh*
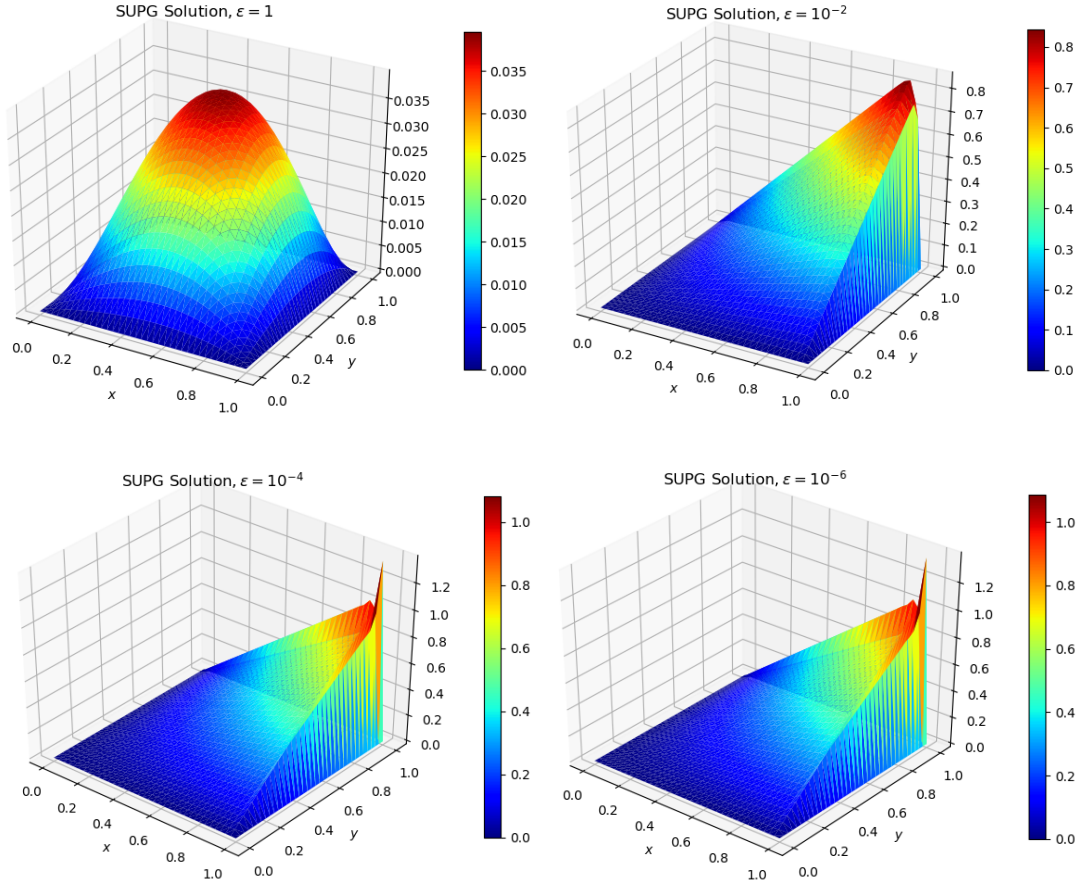
Figure 3.4: SUPG solutions of Eqs. (3.1)–(3.2) for various values of ε; $n_{\text{el}} = 2,592$.

principle [59] from

$$\rho^N = \frac{\ln E^N - \ln E^{2N}}{\ln 2}, \tag{3.22}$$

where $E^N = u - u^N$ and $E^{2N} = u - u^{2N}$ are pointwise (nodal) errors associated with the meshes constructed with $N \times N$ and $2N \times 2N$ elements. Here, u represents the exact solution.

Table 3.1 and Table 3.2 compare the errors in approximations obtained in various norms for varying values of the diffusion parameter ε. Note that the errors measured in the ε-weighted norms are slightly larger than those measured in the L^2 norm. For decreasing values of ε, the errors measured in both norms are identical. It should also be noticed that the errors in the SUPG approximations are smaller than those obtained by employing the SUPG-YZβ formulation for the values of ε smaller larger than $\varepsilon = 10^{-3}$. The reason for this is that for values less than $\varepsilon = 10^{-3}$, convection-dominance becomes significant. In Figure 3.7, semi-logarithmic plots comparing the
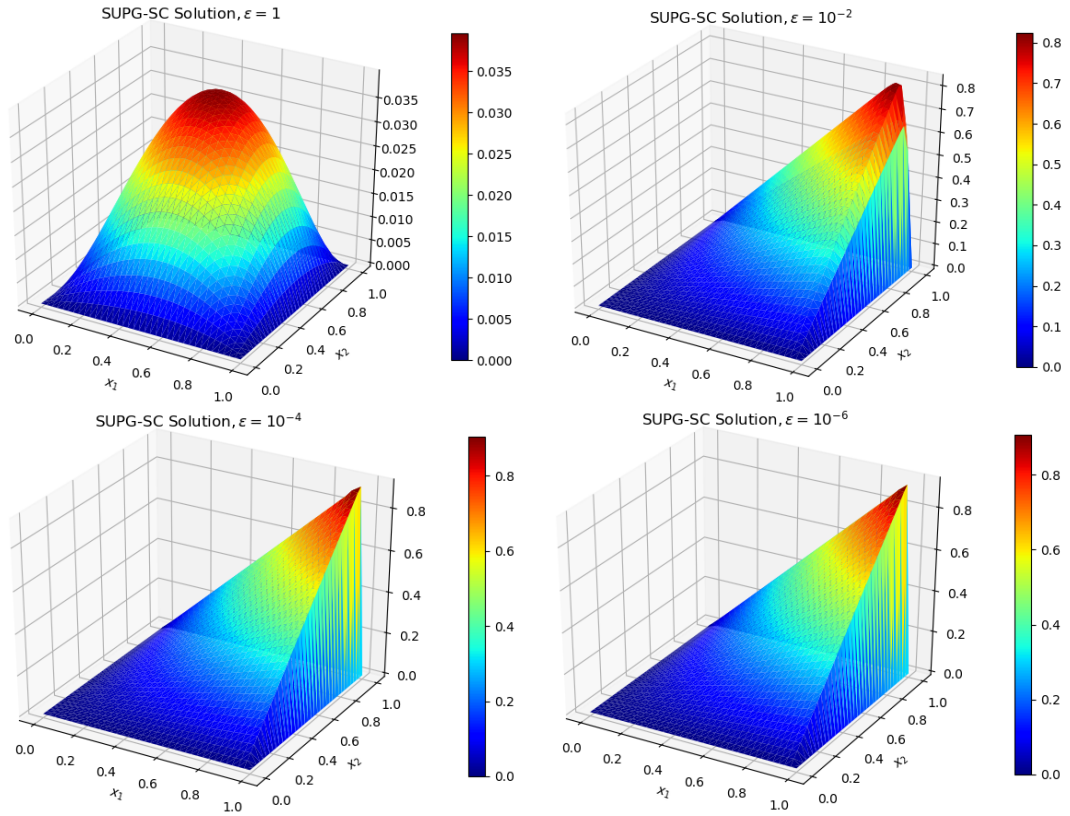
Figure 3.5: SUPG-YZβ solutions of Eqs. (3.1)–(3.2) for various values of ε; $n_{\text{el}} = 2,592$.

degrees of freedom and errors in approximations are presented. From these figures, the superiority of the SUPG-YZβ solutions is clearly observed.

Finally, in Figure 3.8, the solutions obtained by employing the proposed methods are compared along the line $x_2 = 0.2$, and the success of YZβ technique in capturing strong gradients is clearly observed.

In comparison to the results obtained in [210], the problem given by Eqs. (3.1)–(3.2) is considered with much lower diffusivity here, i.e., on the order of $\varepsilon = 10^{-8}$, while it is on the order of $\varepsilon = 10^{-4}$ at most in [210]. However, it is still observed that the SUPG-YZβ formulation eliminates nonphysical oscillations entirely. In addition, it accomplishes this without using any adaptive mesh strategy.
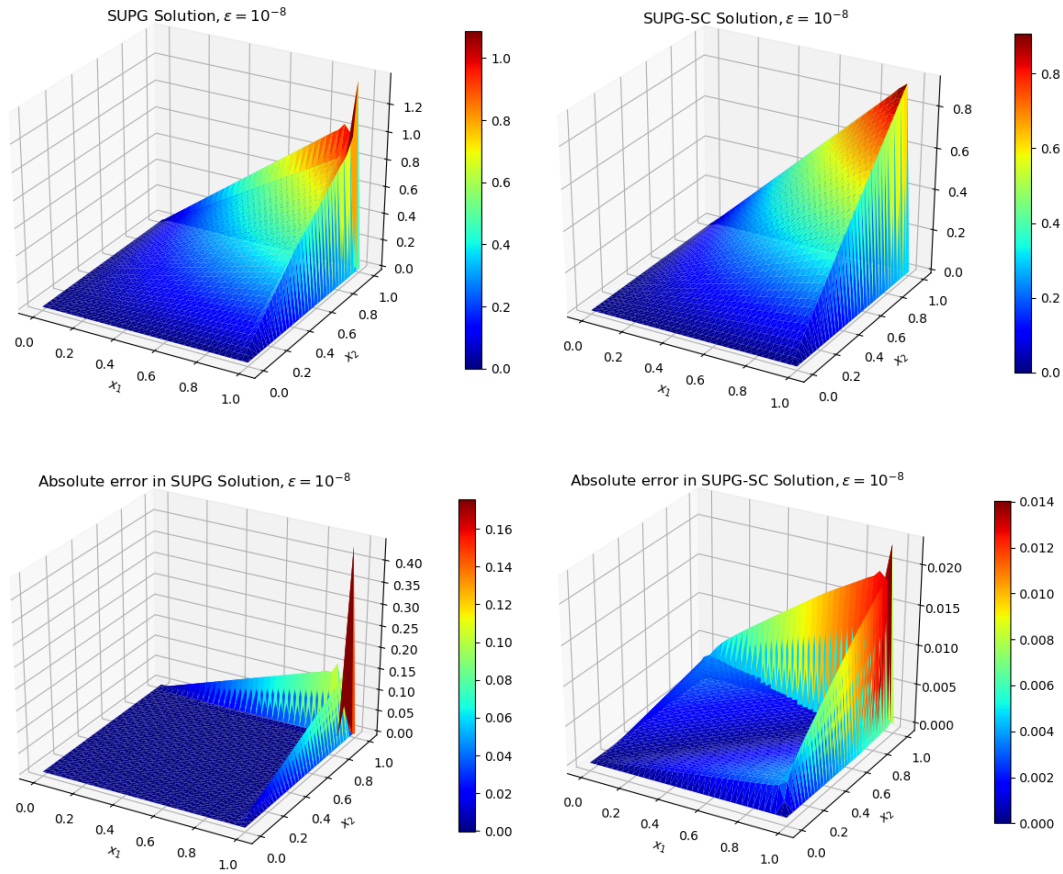
Figure 3.6: Effect of shock-capturing technique in solving Eqs. (3.1)–(3.2) with absolute errors; $\varepsilon = 10^{-8}$, and $n_{\mathrm{el}} = 2,592$.

(a)

(b)
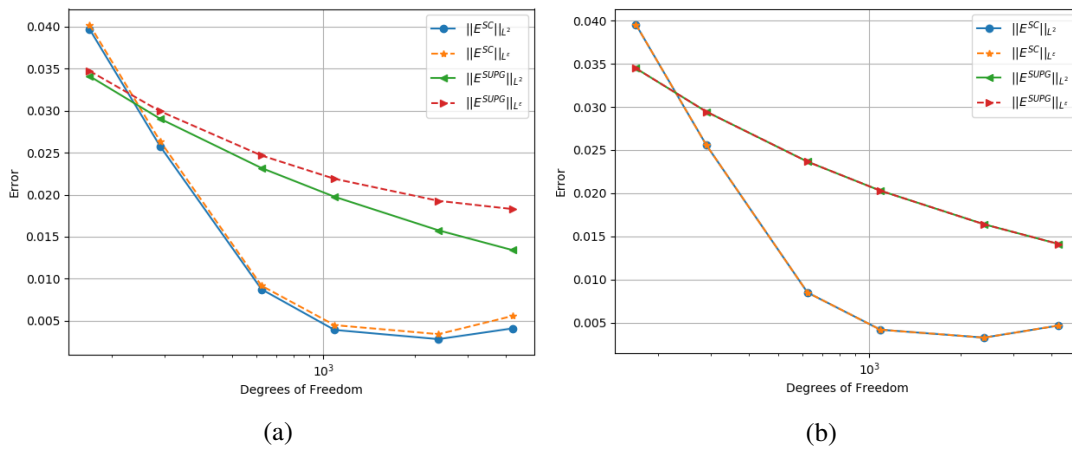
Figure 3.7: Comparison of errors in solving Eqs. (3.1)–(3.2): (a) $\varepsilon = 10^{-4}$, and (b) $\varepsilon = 10^{-8}$.

Table 3.1: Comparison of approximations for solving Eqs. (3.1)–(3.2) in various norms; $n_{\mathrm{el}} = 1,152$.

ε	$\|E^{\mathrm{SUPG}}\|_{L^2}$	$\|E^{\mathrm{SC}}\|_{L^2}$	$\|E^{\mathrm{SUPG}}\|_{L^\infty}$	$\|E^{\mathrm{SC}}\|_{L^\infty}$	$\|E^{\mathrm{SUPG}}\|_{L^\varepsilon}$	$\|E^{\mathrm{SC}}\|_{L^\varepsilon}$
10^0	3.725e-5	9.199e-5	7.165e-5	1.475e-4	1.843e-4	5.586e-4
10^{-1}	3.139e-3	7.376e-3	1.229e-2	2.658e-2	1.176e-2	3.323e-2
10^{-2}	7.460e-3	2.830e-2	9.516e-2	3.233e-1	2.478e-2	9.192e-2
10^{-3}	1.911e-2	1.104e-2	3.353e-1	1.264e-1	2.888e-2	1.574e-2
10^{-4}	2.317e-2	8.749e-3	4.240e-1	9.442e-2	2.466e-2	9.169e-3
10^{-5}	2.236e-2	8.517e-3	4.343e-1	9.085e-2	2.376e-2	8.558e-3
10^{-6}	2.365e-2	8.493e-3	4.354e-1	9.051e-2	2.366e-2	8.498e-3
10^{-7}	2.365e-2	8.491e-3	4.355e-1	9.048e-2	2.365e-2	8.492e-3
10^{-8}	2.365e-2	8.491e-3	4.355e-1	9.047e-2	2.365e-2	8.491e-3

Table 3.2: Comparison of approximations for solving Eqs. (3.1)–(3.2) in various norms; $n_{\mathrm{el}} = 2,592$.

ε	$\|E^{\mathrm{SUPG}}\|_{L^2}$	$\|E^{\mathrm{SC}}\|_{L^2}$	$\|E^{\mathrm{SUPG}}\|_{L^\infty}$	$\|E^{\mathrm{SC}}\|_{L^\infty}$	$\|E^{\mathrm{SUPG}}\|_{L^\varepsilon}$	$\|E^{\mathrm{SC}}\|_{L^\varepsilon}$
10^0	1.660e-5	3.747e-5	3.189e-5	6.039e-5	8.224e-5	2.399e-4
10^{-1}	1.453e-3	3.207e-3	5.694e-3	1.143e-2	5.617e-3	1.621e-2
10^{-2}	1.033e-2	2.236e-2	1.508e-1	2.974e-1	4.665e-2	1.007e-1
10^{-3}	1.380e-2	5.755e-3	2.986e-1	7.809e-2	2.667e-2	1.038e-2
10^{-4}	1.850e-2	2.869e-3	4.250e-1	2.205e-2	2.097e-2	3.086e-3
10^{-5}	1.901e-2	2.613e-3	4.405e-1	1.634e-2	1.928e-2	2.631e-3
10^{-6}	1.906e-2	2.588e-3	4.421e-1	1.609e-2	1.909e-2	2.590e-3
10^{-7}	1.907e-2	2.583e-3	4.423e-1	1.607e-2	1.907e-2	2.585e-3
10^{-8}	1.907e-2	2.585e-3	4.423e-1	1.607e-2	1.907e-2	2.585e-3
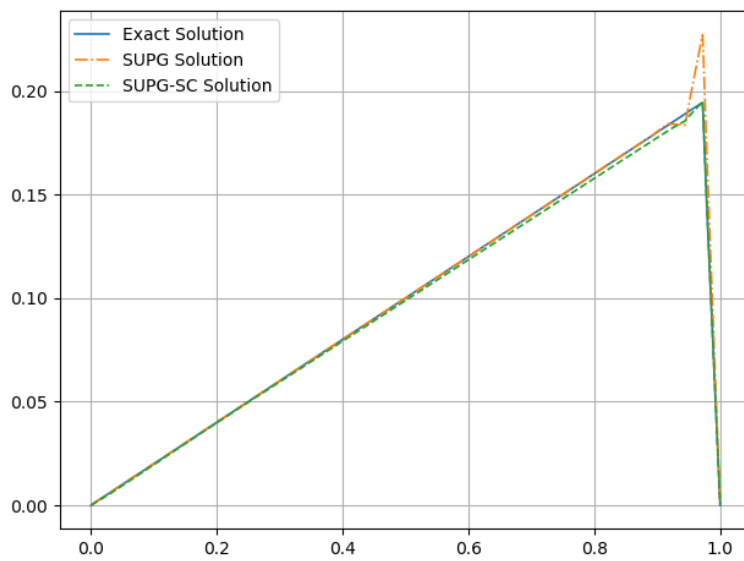
Figure 3.8: Comparison of solutions along line $x_2 = 0.2$ for solving Eqs. (3.1)–(3.2); $\varepsilon = 10^{-6}$, and $n_{\text{el}} = 2,592$.

3.4.2 A Convection-Dominated Linear Reaction-Convection-Diffusion Equation

Consider the following singularly perturbed reaction-convection-diffusion equation [125]:

$$-\nabla \cdot (\varepsilon \nabla u) + \mathbf{b} \cdot \nabla u + u = f \quad \text{in} \quad \Omega = (0, 1)^2, \tag{3.23}$$

$$u = 0 \quad \text{on} \quad \partial\Omega, \tag{3.24}$$

where the source function, $f = f(x_1, x_2)$, is given as follows:

$$f(x_1, x_2) = x_1(1 - x_1) + x_2(1 - x_2). \tag{3.25}$$

There is no known analytic solution of this problem [125]. The SUPG-YZβ formulation can be deduced from Eq. (3.5) presented for solving the previous linear problem. Here, the convection vector $\mathbf{b} = [1, 11/7]^T$, and unlike the previous problem, Eq. (3.23) contains a linear reaction term, i.e., u.

For solving Eqs. (3.23)–(3.24), the mesh shown in Figure 3.2b is used. In computations, the scaling term Y of YZβ is set as Y$= 1.0$. The number of Newton–Raphson iterations in SUPG-YZβ computations is 5 for both meshes with $1,152$ elements and $2,592$ elements.

One can point out from Figure 3.9 that the GFEM yields unacceptable solutions in solving Eqs. (3.23)–(3.24) for the values of the perturbation parameter less than $\varepsilon = 10^{-2}$. In Figure 3.10, the solutions obtained with the SUPG and SUPG-YZβ formulations are compared, and the effect of shock-capturing term is illustrated. In Figure 3.11, semi-logarithmic plots comparing the degrees of freedom and errors in approximations are given.

Finally, in Figure 3.12, the solutions obtained by employing the proposed methods are compared along the line $x_2 = 0.9$, and the success of YZβ technique in capturing strong gradients is clearly observed.

The authors of [125] perform their computations with linear elements on Shiskin-type meshes for solving Eqs. (3.23)–(3.24). The results obtained in [125] are found to be in good agreement with those obtained by employing the SUPG-YZβ formulation without exhibiting any oscillatory behavior.
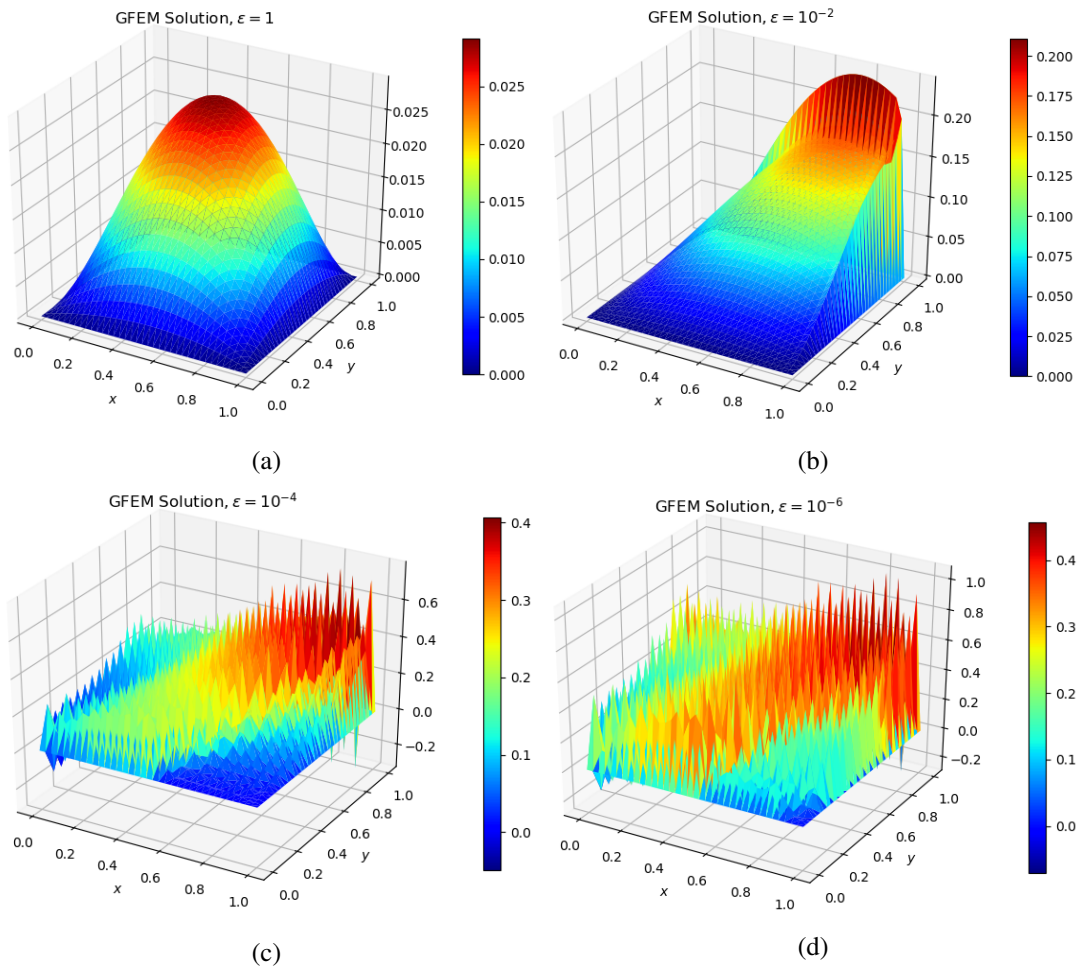
Figure 3.9: GFEM approximations for solving Eqs. (3.23)–(3.24): (a) $\varepsilon = 1.0$, (b) $\varepsilon = 10^{-2}$, (c) $\varepsilon = 10^{-4}$, and (d) $\varepsilon = 10^{-6}$; $n_{\text{el}} = 2,592$.
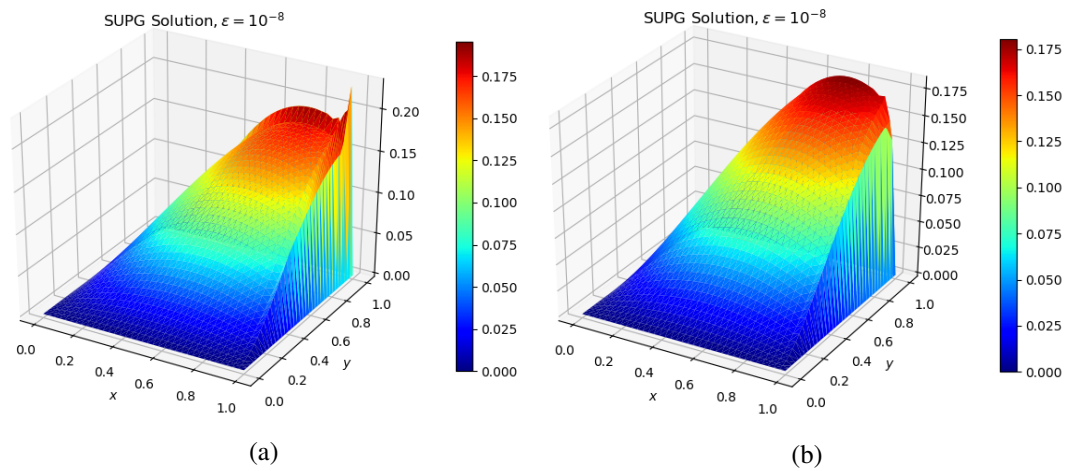
Figure 3.10: Effect of shock-capturing in solving Eqs. (3.23)–(3.24): (a) SUPG solution, (b) SUPG-YZβ solution; $\varepsilon = 10^{-8}$, and $n_{\text{el}} = 2,592$.

(a)	(b)
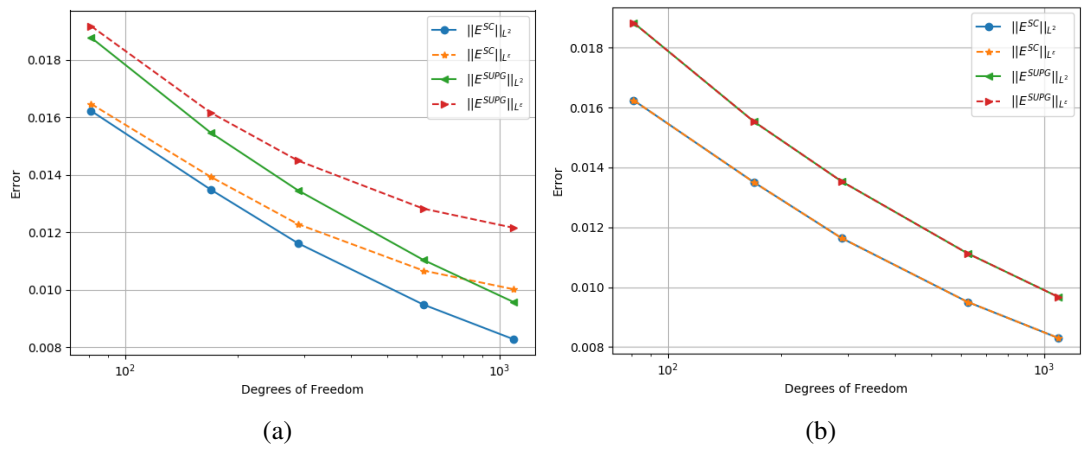
Figure 3.11: Comparison of errors in solving Eqs. (3.23)–(3.24) for (a) $\varepsilon = 10^{-4}$, and (b) $\varepsilon = 10^{-8}$.
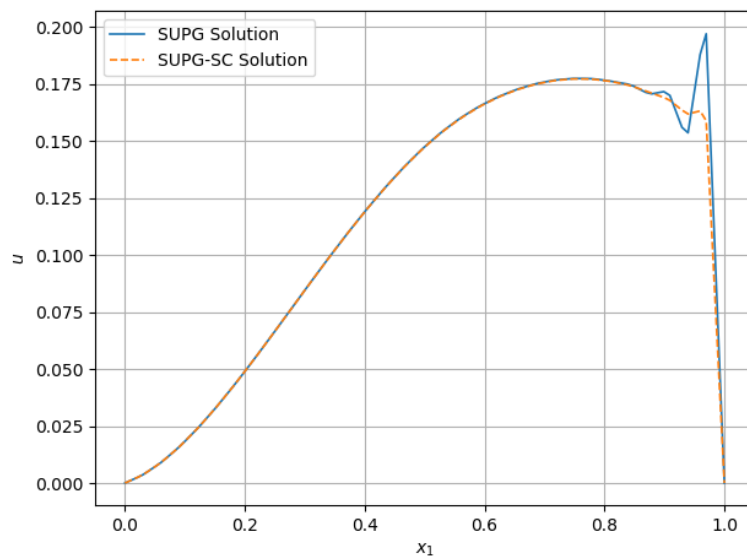
Figure 3.12: Comparison of solutions along line $x_2 = 0.9$ for solving Eqs. (3.23)–(3.24); $\varepsilon = 10^{-4}$, and $n_{\mathrm{el}} = 2,592$.

3.4.3 A Convection-Dominated Nonlinear Reaction-Convection-Diffusion Equation

Consider the following singularly perturbed nonlinear stationary reaction-convection-diffusion equation [13, 198, 209]:

$$-\nabla \cdot (\varepsilon \nabla u) + \boldsymbol{\beta} \cdot \nabla u + \alpha u + r(u) = 0 \quad \text{in} \quad \Omega = (0,1)^2, \qquad (3.26)$$

where

$$\boldsymbol{\beta} = [-x_2, x_1]^T, \quad \alpha = 1, \quad r(u) = -\frac{u}{u+1}. \qquad (3.27)$$

Here, $\boldsymbol{\beta}$ is the convection vector, α is the linear reaction constant, and the function $r(u)$ represents the nonlinear reaction term. For specific assumptions that should be made on these terms, one can refer to [13, 198, 209].

The Dirichlet and Neumann boundary conditions are given as follows:

$$g^D(x_1, x_2) = \begin{cases} 1, & \text{if } x_1 \in [1/3, 2/3] \text{ and } x_2 = 0, \\ 0, & \text{if } x_1 \in [0, 1/3] \cup [2/3, 1] \text{ and } x_2 = 0, \\ 0, & \text{if } x_1 = 1, \\ 0, & \text{if } x_2 = 1, \end{cases} \qquad (3.28)$$

and $g^N(x_1, x_2) = 0$ for $x_1 = 0$ and $x_2 \in [0, 1]$, respectively.

The SUPG formulation with shock-capturing can be given as follows: find $u^h \in \mathcal{S}^h$ such that for all test functions $w^h \in \mathcal{V}^h$,

$$\int_\Omega w^h \left[-\nabla \cdot \left(\varepsilon \nabla u^h \right) + \boldsymbol{\beta}^h \cdot \nabla u^h + \alpha u^h + r(u^h) \right] \, d\Omega$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \tau_{\text{SUPG}} \left(\boldsymbol{\beta}^h \cdot \nabla u^h + \alpha u^h + r(u^h) \right) \left(\boldsymbol{\beta}^h \cdot \nabla w^h \right) \, d\Omega$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \nu_{\text{SHOC}} \left(\nabla u^h \cdot \nabla w^h \right) \, d\Omega = 0, \quad (3.29)$$

where the solution and test function spaces are defined as follows:

$$\mathcal{S}^h = \{ u^h | \, u^h \in H^{h1}(\Omega), \, u^h = g^D \text{ on } \Gamma^D \}, \qquad (3.30)$$

$$\mathcal{V}^h = \{ w^h | \, w^h \in H^{h1}(\Omega), \, w^h = 0 \text{ on } \Gamma^D \}, \qquad (3.31)$$

Here, the finite-dimensional space H^{1h} is defined as given in Eq. (3.8), and the function $g^D = g^D(x_1, x_2)$ defined in Eq. (3.28) represents the Dirichlet boundary condition
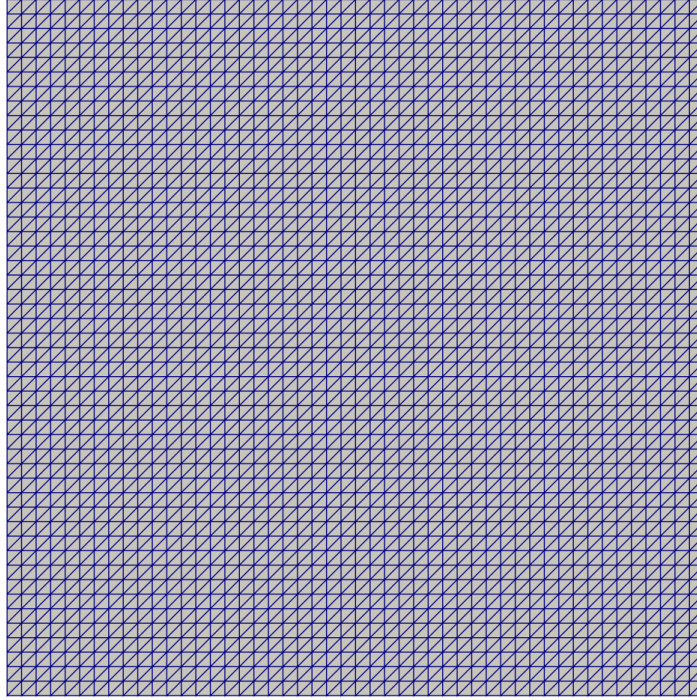
Figure 3.13: Mesh used for solving Eq. (3.26); $n_{\text{en}} = 2,401$ and $n_{\text{el}} = 4,608$.

that Eq. (3.26) is subject to. The term Γ^D represents the part of boundary $\Gamma = \partial\Omega$ where Dirichlet-type boundary conditions apply.

The stabilization and shock-capturing parameters are determined as in the linear problems studied in Sections 3.4.1–3.4.2. The mesh constructed from $2,401$ nodes and $4,608$ triangular elements, shown in Figure 3.13, is used for solving Eq. (3.26). The Newton-Raphson process fails to converge in GFEM computations for the values smaller than $\varepsilon = 10^{-3}$. The scaling parameter Y of YZβ is set as Y $= 0.25$ in computations. The number of the Newton–Raphson iterations in SUPG-YZβ computation is 6.

In Figure 3.14, the GFEM solutions of Eq. (3.26) for various values of the diffusion parameter, ε, are presented. Figure 3.15 compares the approximations obtained by the GFEM and SUPG formulations for $\varepsilon = 10^{-8}$. It is seen that the solution obtained by using the SUPG formulation eliminates the spurious oscillations significantly. However, similar to the linear examples, it is observed that the SUPG formulation needs to be strengthened for the regions where strong gradients occur, even though the spurious oscillations are significantly reduced. As can be seen in Figure 3.16, when the SUPG formulation is supplemented with YZβ shock-capturing, the spurious

oscillations are eliminated almost completely. Figure 3.17 compares the SUPG and SUPG-YZβ solutions obtained along line $x_2 = 0.2$, and it reveals that that SUPG-YZβ formulation captures sharp gradients accurately without spurious oscillations.

While the studies existing in the literature, for example, see [13,198,209], can generally eliminate the spurious oscillations by using higher-order polynomials or much finer meshes, or a combination of these, and thus with a much higher degree of freedom, the SUPG-YZβ formulation is found to be successful in eliminating nonphysical oscillations even with linear elements and relatively much coarser meshes.
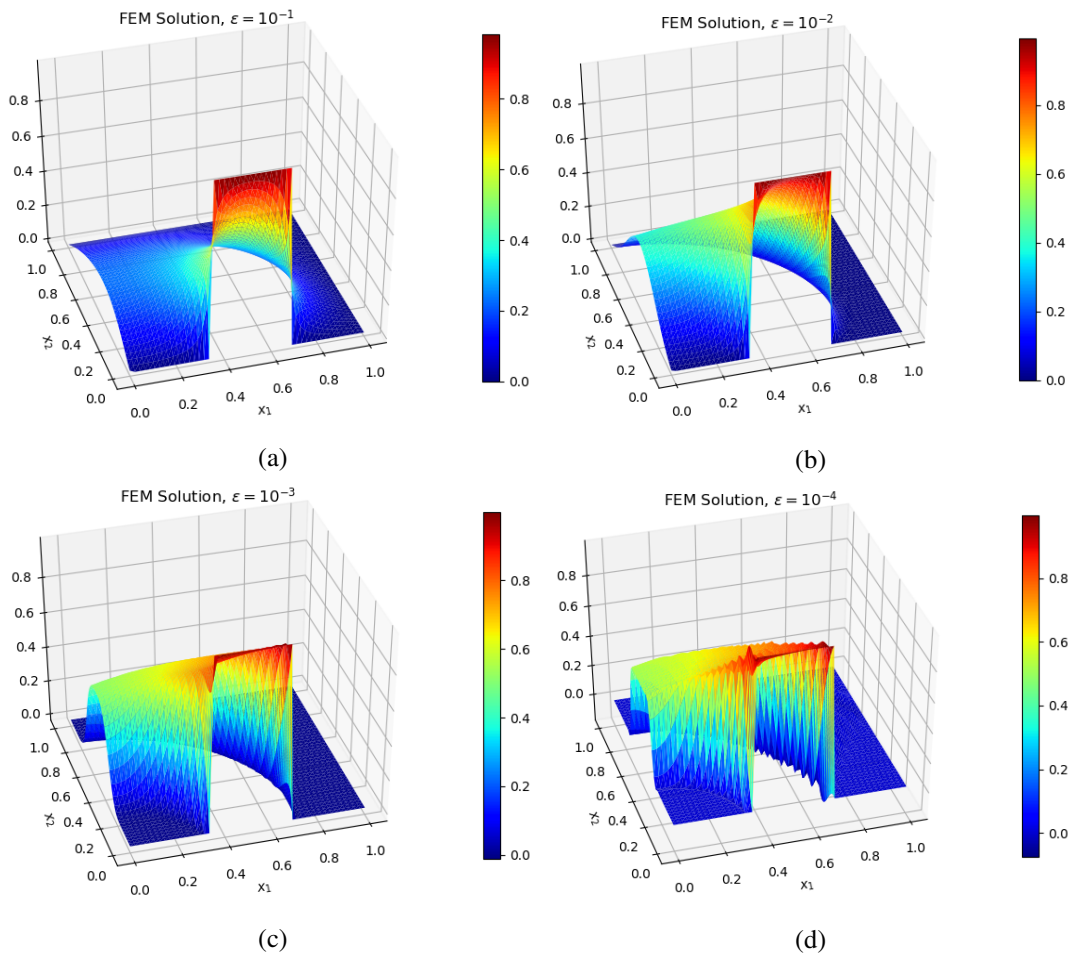
Figure 3.14: GFEM solutions of Eq. (3.26) for (a) $\varepsilon = 10^{-1}$, (b) $\varepsilon = 10^{-2}$, (c) $\varepsilon = 10^{-3}$, and (d) $\varepsilon = 10^{-4}$; $n_{\mathrm{el}} = 4,608$.

(a)   (b)
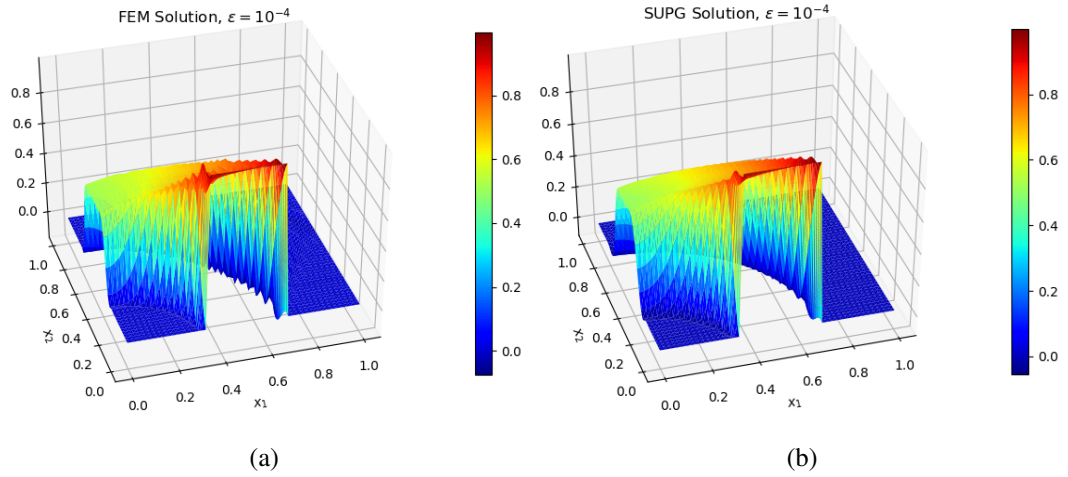
Figure 3.15: Comparison of the GFEM and SUPG solutions in solving Eq. (3.26): (a) GFEM solution, and (b) SUPG solution; $\varepsilon = 10^{-4}$, and $n_{\mathrm{el}} = 4,608$.

(a)   (b)

(c)   (d)
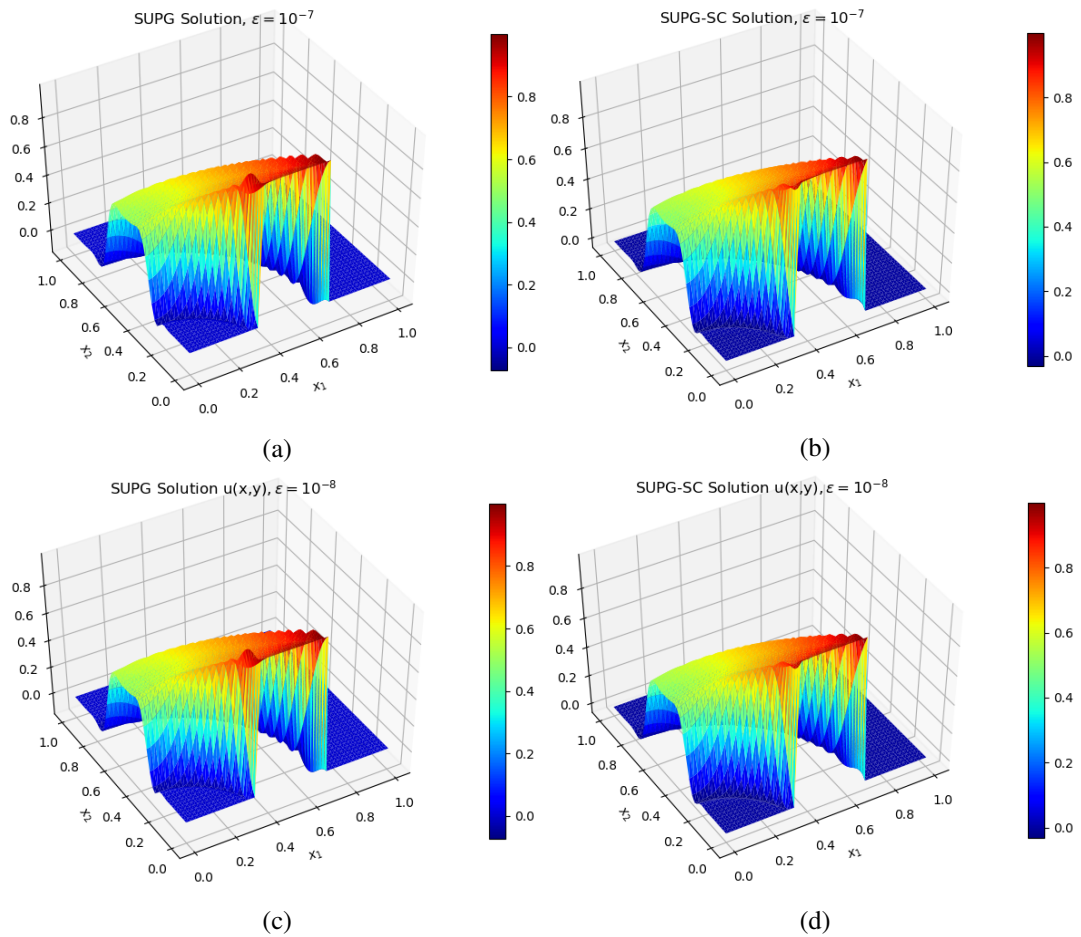
Figure 3.16: Effect of shock-capturing in solving Eq. (3.26): (a) SUPG solution for $\varepsilon = 10^{-7}$, (b) SUPG-YZβ solution for $\varepsilon = 10^{-7}$, (c) SUPG solution for $\varepsilon = 10^{-8}$, and (d) SUPG-YZβ solution for $\varepsilon = 10^{-8}$; $n_{\mathrm{el}} = 4,608$.
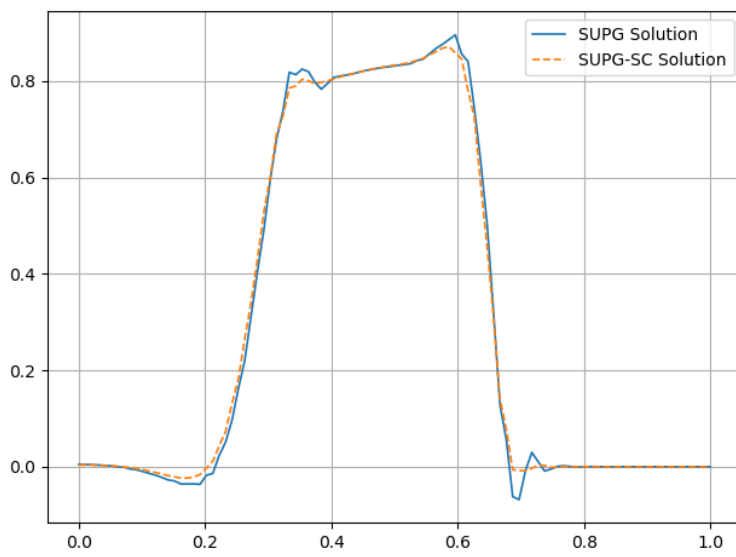
Figure 3.17: Comparison of solutions along line $x_2 = 0.2$ for solving Eq. (3.26); $n_{el} = 4,608$.

3.4.4 A Burgers'-type Equation at High Reynolds Number

Consider the following 2D uncoupled Burgers' equation [10, 139]:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x_1} + u\frac{\partial u}{\partial x_2} = \frac{1}{Re}\left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2}\right), \quad t \in (0,1], \quad \Omega = [0,1]^2, \quad (3.32)$$

where the initial and Dirichlet boundary conditions are prescribed such that the exact solution [139], given by

$$u(x_1, x_2, t) = \frac{1}{1 + \exp\left(\frac{Re(x_1 + x_2 - t)}{2}\right)}, \quad (3.33)$$

is satisfied.

The SUPG-YZβ formulation of Eq. (3.32) can be derived in the light of the procedures followed for previous test computations. Note that the diffusion parameter ε is defined as $\varepsilon = \frac{1}{Re}$ (see definitions of the stabilization parameters given by Eqs. (3.9)–(3.10)). One can also refer to Section 3.4.5 in which the derivation of the SUPG-YZβ formulation of n_{sd}-dimensional coupled Burgers' equations is presented since the formulations for the decoupled cases can be easily deduced from it.

In computations, the mesh shown in Figure 3.18, constructed with $1,849$ nodes and $3,528$ triangular elements, is used. The scaling term Y of YZβ is set as Y$= 1.0$. Time integration is performed by using the implicit Euler method: as we step from time-step n to $n+1$,

$$\frac{\partial u}{\partial t} \approx \frac{u_{n+1} - u_n}{\Delta t}. \quad (3.34)$$

The average number of the Newton–Raphson iterations at each time step for the GFEM, SUPG, and SUPG-YZβ are 4, 4, and 5, respectively, where $\Delta t = 5.0 \times 10^{-3}$ s. It is observed that the average number of nonlinear iterations in SUPG-YZβ computations reduces to 4 when the time-step length is set as $\Delta t = 1.0 \times 10^{-3}$ s.

In Figure 3.19, the effect of the shock-capturing term is clearly observed; steep gradients are captured successfully. Since the GFEM approximation does not contain globally propagated spurious oscillations (only local oscillations are observed about steep gradients, see Figure 3.19a), the SUPG formulation has no significant effect in stabilizing the GFEM solution, and both formulations fail to capture rapid and abrupt changes. Figure 3.20 compares the absolute errors in approximations. Even though
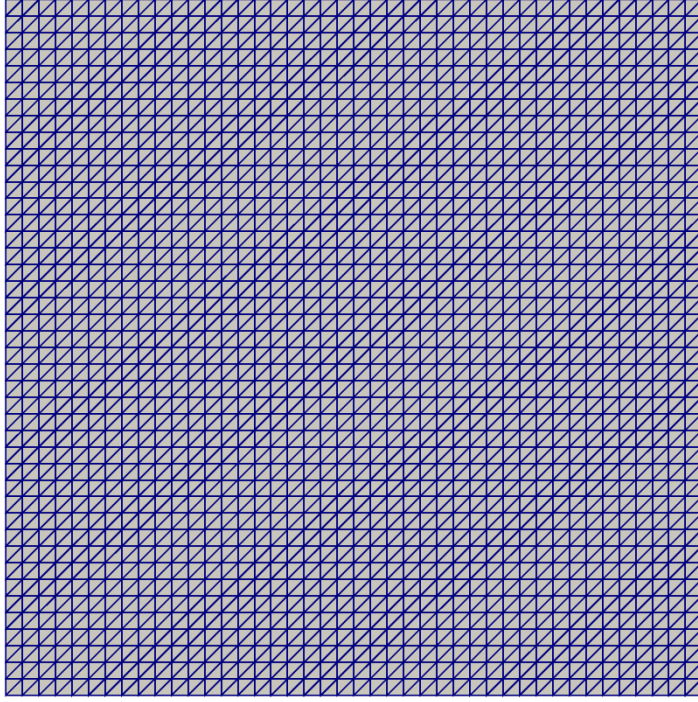
Figure 3.18: Mesh used for solving Eq. (3.32); $n_{\text{en}} = 1,849$, and $n_{\text{el}} = 3,528$.

the absolute error in the SUPG-YZβ approximation is higher than the others, it does not exhibit any oscillatory behavior, unlike the others. It is important to note that the primary goal is to find a compromise between stability and accuracy. In Figure 3.21, the solutions obtained by employing the proposed methods are compared along the line $x_2 = 0.45$, and the success of YZβ technique in capturing strong gradients is clearly observed.

The problem given by Eq. (3.32) is treated by employing a numerical scheme based on modified bi-cubic B-spline functions in [139] and using a numerical formulation based on trigonometric cubic B-spline functions in [10]. The computations in these studies are performed for very moderate Reynolds numbers, i.e., $Re = 100$ and $Re = 50$, respectively. In our computations, although we work with much higher Reynolds numbers, e.g., $Re = 10^4$, the solutions obtained with the SUPG-YZβ formulation do not contain any spurious oscillations.
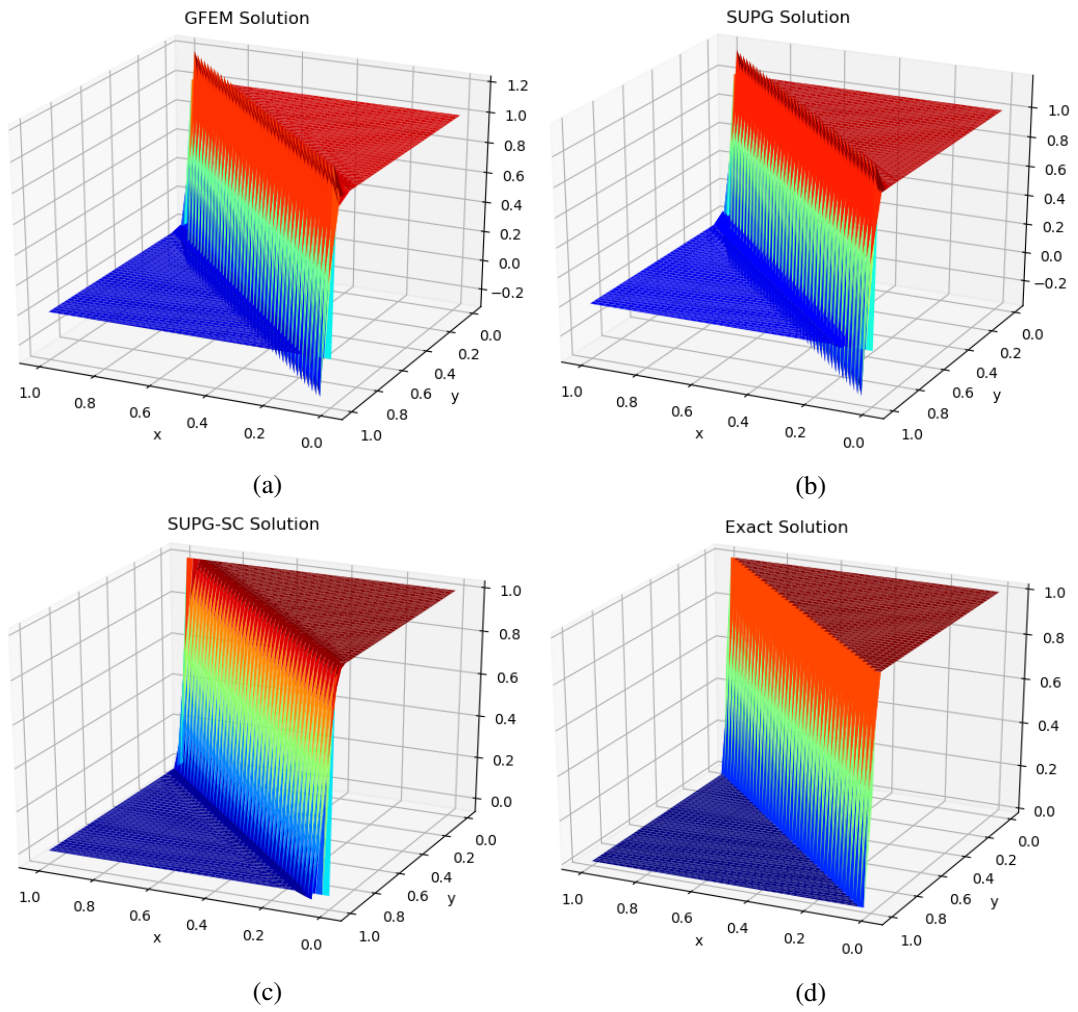
Figure 3.19: Solution of Eq. (3.32) for $t_{\mathrm{f}} = 1.0$ s: (a) GFEM, (b) SUPG, (c) SUPG-YZβ, and (d) exact solution; $Re = 10^4$, $n_{\mathrm{el}} = 3,528$, and $\Delta t = 5.0 \times 10^{-3}$ s.
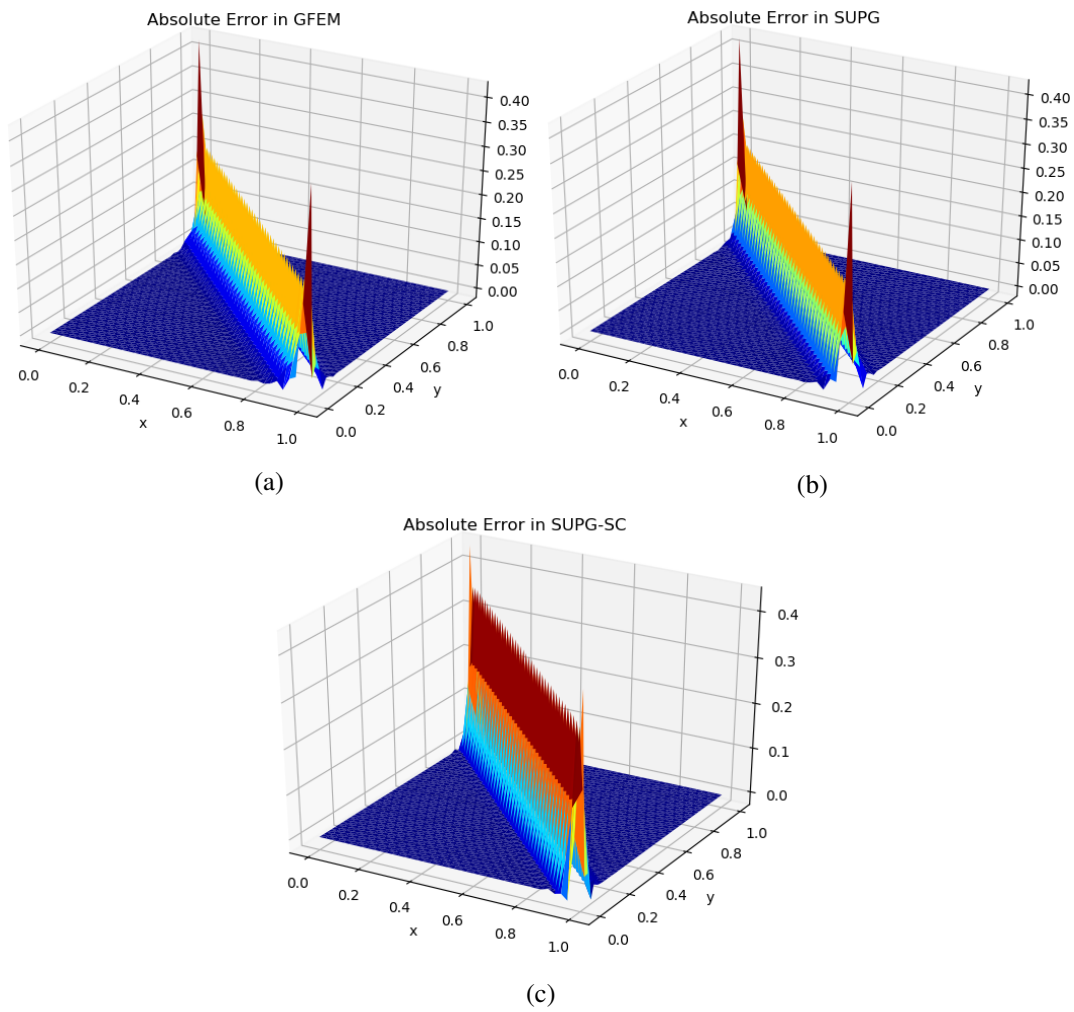
Figure 3.20: Absolute errors in solving Eq. (3.32) for $t = 1.0$ s: (a) GFEM, (b) SUPG, and (c) SUPG-YZβ; $Re = 10^4$, $n_{\text{el}} = 3,528$, and $\Delta t = 5.0 \times 10^{-3}$ s.
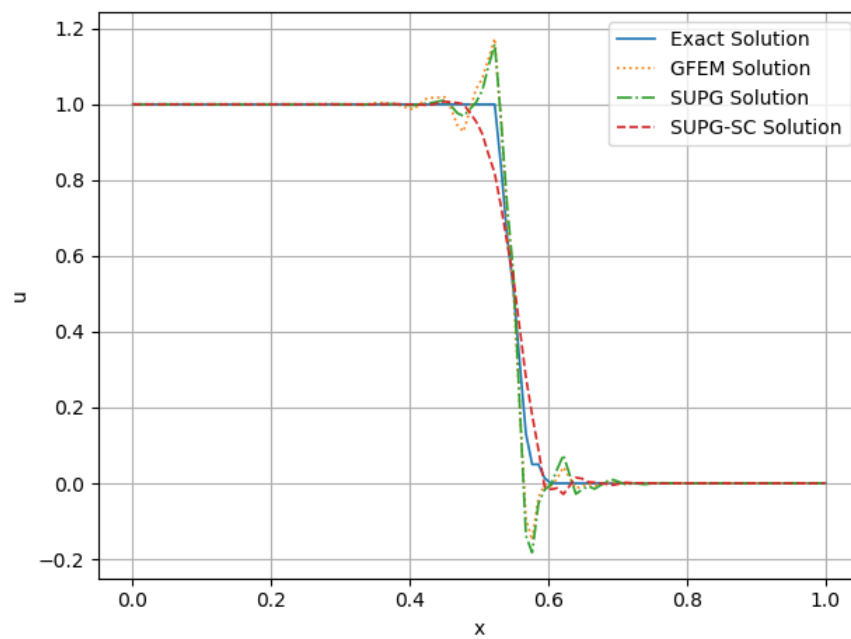
Figure 3.21: Comparison of solutions along line $x_2 = 0.45$ for solving Eq. (3.32); $Re = 10^4$, $n_{\mathrm{el}} = 3,528$, $t = 1.0$ s, and $\Delta t = 5.0 \times 10^{-3}$ s.

3.4.5 A Coupled System of Burgers'-type Equations at High Reynolds Numbers

Consider the following 2D coupled Burgers' equations [169, 179]:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x_1} + v\frac{\partial u}{\partial x_2} = \frac{1}{Re}\left(\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2}\right), \tag{3.35}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x_1} + v\frac{\partial v}{\partial x_2} = \frac{1}{Re}\left(\frac{\partial^2 v}{\partial x_1^2} + \frac{\partial^2 v}{\partial x_2^2}\right), \tag{3.36}$$

where $(x_1, x_2) \in \Omega = [0, 1]^2$ and $t \in (0, t_f]$, with $t_f > 0$. The initial and Dirichlet boundary conditions are prescribed such that the component-wise exact solutions, given as [62]

$$u(x_1, x_2, t) = \frac{3}{4} - \frac{1}{4\left[1 + \exp\left[\frac{Re}{32}\left(-4x_1 + 4x_2 - t\right)\right]\right]}, \tag{3.37}$$

$$v(x_1, x_2, t) = \frac{3}{4} + \frac{1}{4\left[1 + \exp\left[\frac{Re}{32}\left(-4x_1 + 4x_2 - t\right)\right]\right]}, \tag{3.38}$$

are satisfied.

It is possible to recast the n_{sd}-dimensional coupled Burgers' equations as follows:

$$\frac{\partial u_i}{\partial t} + (\mathbf{u} \cdot \nabla u_i) = \frac{1}{Re}\Delta u_i, \quad i = 1, 2, \dots, n_{\text{sd}}, \tag{3.39}$$

where

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n_{\text{sd}}} \end{bmatrix}, \quad \nabla = \begin{bmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \vdots \\ \frac{\partial}{\partial x_{n_{\text{sd}}}} \end{bmatrix}, \quad \text{and} \quad \Delta = \nabla \cdot \nabla. \tag{3.40}$$

Then, a semi-discrete GFEM formulation of Eq. (3.39) can be given as follows: find $u_i^h \in \mathcal{S}_{u_i}^h$ such that $\forall w_i^h \in \mathcal{V}_{u_i}^h$,

$$\int_\Omega w_i^h \left(\frac{\partial u_i^h}{\partial t} + \mathbf{u}^h \cdot \nabla u_i^h\right) d\Omega + \frac{1}{Re}\int_\Omega \nabla w_i^h \cdot \nabla u_i^h \, d\Omega - \int_{\Gamma_{h_i}} w_i^h \mathbf{h}_i^h \, d\Gamma = 0, \tag{3.41}$$

where $i = 1, 2, \dots, n_{\text{sd}}$. The given function "h$_i$" denotes the Neumann boundary condition associated with the ith equation, and Γ_{h_i} is the part of the boundary where

such conditions are prescribed. The finite-dimensional solution and test function spaces are defined as follows:

$$\mathcal{S}_{u_i}^h = \{u_i^h \in H^{h1}(\Omega) | u_i^h = g_i^D \text{ on } \Gamma^D\}, \tag{3.42}$$

$$\mathcal{V}_{u_i}^h = \{w_i^h \in H^{h1}(\Omega) | w_i^h = 0 \text{ on } \Gamma^D\}, \tag{3.43}$$

where the finite element space H^{1h} is defined as given in Eq. (3.8). The term Γ^D represents the part of boundary $\Gamma = \partial\Omega$ where Dirichlet-type boundary conditions apply. If Γ^N corresponds to the part of the boundary Γ where Neumann-type boundary conditions are prescribed, then it is clear that $\Gamma = \overline{\Gamma^D \cup \Gamma^N}$ and $\Gamma^D \cap \Gamma^N = \emptyset$.

A semi-discrete SUPG formulation of Eq. (3.39) can be given as follows: find $u_i^h \in \mathcal{S}_{u_i}^h$ such that $\forall w_i^h \in \mathcal{V}_{u_i}^h$,

$$\int_\Omega w_i^h \left(\frac{\partial u_i^h}{\partial t} + \mathbf{u}^h \cdot \nabla u_i^h \right) d\Omega + \frac{1}{Re} \int_\Omega \nabla w_i^h \cdot \nabla u_i^h \, d\Omega - \int_{\Gamma_{h_i}} w_i^h \mathbf{h}_i^h \, d\Gamma$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \tau_{\text{SUPG}}^i (\mathbf{u}^h \cdot \nabla w_i) \left(\frac{\partial u_i^h}{\partial t} + \mathbf{u}^h \cdot \nabla u_i^h - \frac{1}{Re} \nabla \cdot \nabla u_i^h \right) d\Omega$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \nu_{\text{SHOC}}^i (\nabla w_i^h \cdot \nabla u_i^h) \, d\Omega = 0, \quad (3.44)$$

for each $1, 2, \ldots, n_{\text{sd}}$. The terms τ_{SUPG}^i and ν_{SHOC}^i are the stabilization and shock-capturing parameters associated with the ith equation, respectively. Note that the first line in Eq. (3.44) represents the standard GFEM formulation introduced in Eq. (3.41), the second line contains the stabilization and residual terms, and the last one is for shock-capturing.

The stabilization parameters, τ_{SUPG}^i's, given as [167]

$$\tau_{\text{SUPG}}^i = \left[\left(\frac{2}{\Delta t} \right)^2 + \left(\frac{2\|\mathbf{u}^h\|}{h^e} \right)^2 + \left(\frac{4\nu}{(h^e)^2} \right)^2 \right]^{-\frac{1}{2}} \tag{3.45}$$

construct the diagonal stabilization matrix $\boldsymbol{\tau}_{\text{SUPG}}$, where $i = 1, 2, \ldots, n_{\text{sd}}$. Here, the term h^e is the smallest element edge associated with element e, and Δt is the time-step length. Note the viscosity, ν, is defined as $\nu = \frac{1}{Re}$.

The shock-capturing parameters, ν_{SHOC}^i's, are modified as follows:

$$\nu_{\text{SHOC}}^i = |\mathbf{Y}_i^{-1} \mathbf{Z}_i| \left(\sum_{k=1}^{n_{\text{sd}}} \left| \mathbf{Y}_i^{-1} \frac{\partial u_i^h}{\partial x_k} \right|^2 \right)^{\frac{\beta}{2}-1} \left(\frac{h_{\text{SHOC}}^i}{2} \right)^\beta, \tag{3.46}$$

for each $i = 1, 2, \ldots, n_{sd}$. In this formulation, for stationary problems, the quantities Z_i's are given as

$$Z_i = \mathbf{u}^h \cdot \nabla u_i^h - \frac{1}{Re} \Delta u_i^h, \tag{3.47}$$

where $i = 1, 2, \ldots, n_{sd}$. For time-dependent problems, the quantities Z_i's can be defined as

$$Z_i = \frac{\partial u_i^h}{\partial t} + \mathbf{u}^h \cdot \nabla u_i^h - \frac{1}{Re} \Delta u_i^h. \tag{3.48}$$

We use Eq. (3.48) in our computations. The element length scales, h_{SHOC}^i's, and the unit vectors $\mathbf{j}_i$'s are given as follows:

$$h_{SHOC}^i = 2 \left(\sum_{a=1}^{n_{en}} |\mathbf{j}_i \cdot \nabla N_a^i| \right)^{-1}, \tag{3.49}$$

$$\mathbf{j}_i = \frac{\nabla u_i^h}{\|\nabla u_i^h\|}, \tag{3.50}$$

for each $i = 1, 2, \ldots, n_{sd}$. In Eq. (3.49), the term N_a^i represents the shape function associated with element node "a," and $\mathbf{j}_i$ is the unit vector in the direction of the gradient of u_i.

Note that because the stabilization terms and shock-capturing parameter include the residual of Eq. (3.39) as a factor, these added terms vanish when the exact solution is substituted into the SUPG-YZβ formulation given by Eq. (3.44).

The time integration is performed through the implicit Euler scheme (see Eq. (3.34)). The average number of the Newton–Raphson iterations at each time step for the GFEM, SUPG, and SUPG-YZβ are 4. The scaling matrix $\mathbf{Y}$ of YZβ is set as $\mathbf{Y} = 2.5\,\mathbf{I}$.

The results obtained by using the shock-capturing technique at various time steps were depicted in Figure 3.22 and Figure 3.24. Figure 3.23 and Figure 3.25 show that the GFEM gives unacceptable results. Again, by comparison, it is clear that when shock-capturing is used, nonphysical oscillations are almost completely eliminated. In Figures 3.26–3.27, the solutions obtained by using the proposed methods are compared along the line $x_2 = 0.9$, and the success of YZβ technique in capturing strong gradients is clearly observed.

Figure 3.28 depicts the stabilization (see Eq. (3.45)) and shock-capturing (see Eq. (3.46)) parameters computed at $t = 2.0$ s. Figure 3.28b corresponds to shock-capturing pa-

rameter for Eq. (3.35) and Figure 3.28c corresponds to that of Eq. (3.36). One can point out from these figures that the component-wise shock-capturing parameters introduce artificial dissipation only for the regions where localized sharp gradients (rapid changes) occur. Thus, as expected, the effect of shock-capturing parameters vanishes where the solution is smooth. Besides that, it is observed that the stabilization parameter introduces more artificial diffusion where advection dominates the flow field, as expected.

The problem given by Eqs. (3.35)–(3.36) is treated by using a numerical scheme based on exponential cubic B-spline functions in [169] and by employing an implicit logarithmic finite-difference method in [179]. In both studies, the authors carry out the computations for $Re = 100$, which is not sufficiently high to evaluate the performance of the proposed techniques in eliminating spurious oscillations. Even though the Reynolds number is set to $Re = 10^4$ in our computations, the SUPG-YZβ formulation manages to capture strong gradients accurately.
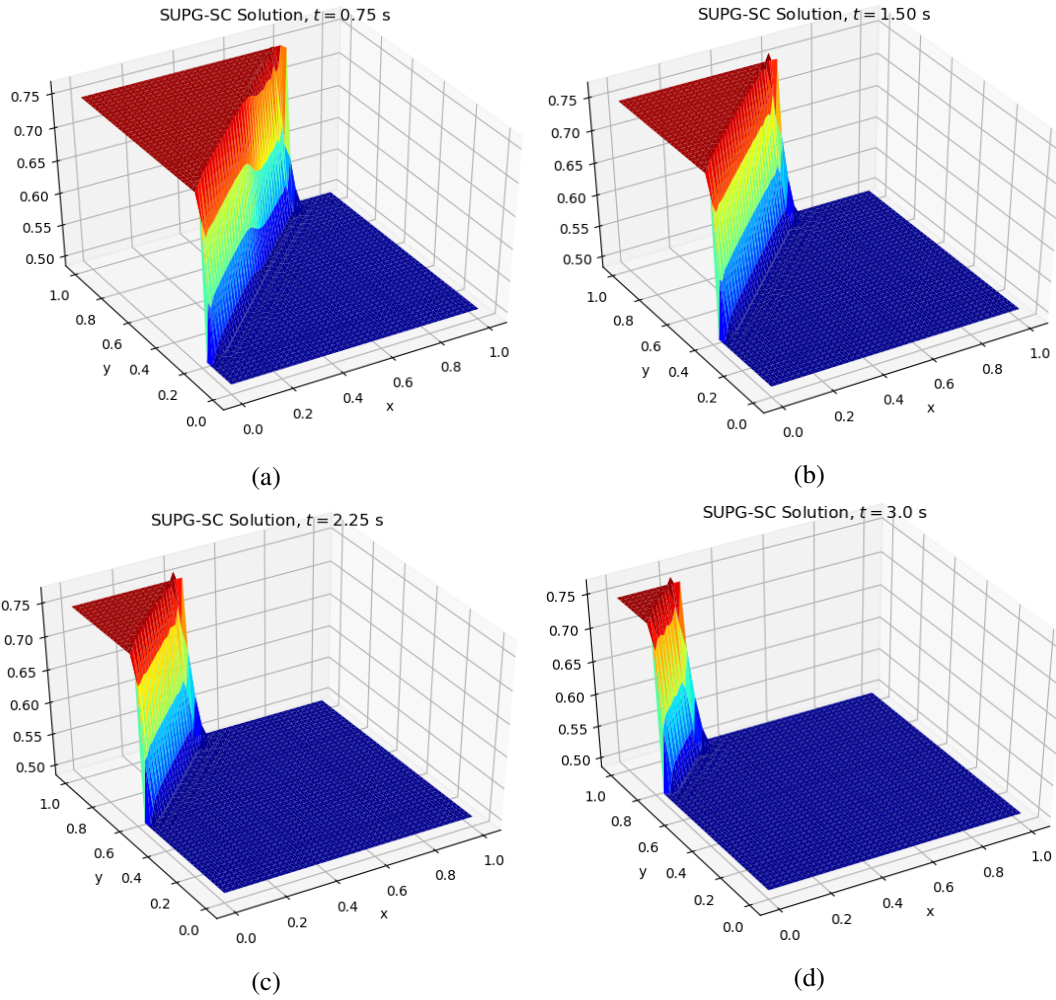
Figure 3.22: Solution of $u(x_1, x_2, t)$ in Eqs. (3.35)–(3.36) obtained with SUPG-YZβ at (a) $t_{\mathrm{f}} = 0.75$ s, (b) $t_{\mathrm{f}} = 1.5$ s, (c) $t_{\mathrm{f}} = 2.25$ s, and (d) $t_{\mathrm{f}} = 3.0$ s; $Re = 10^4$, and $\Delta t = 5.0 \times 10^{-3}$ s.
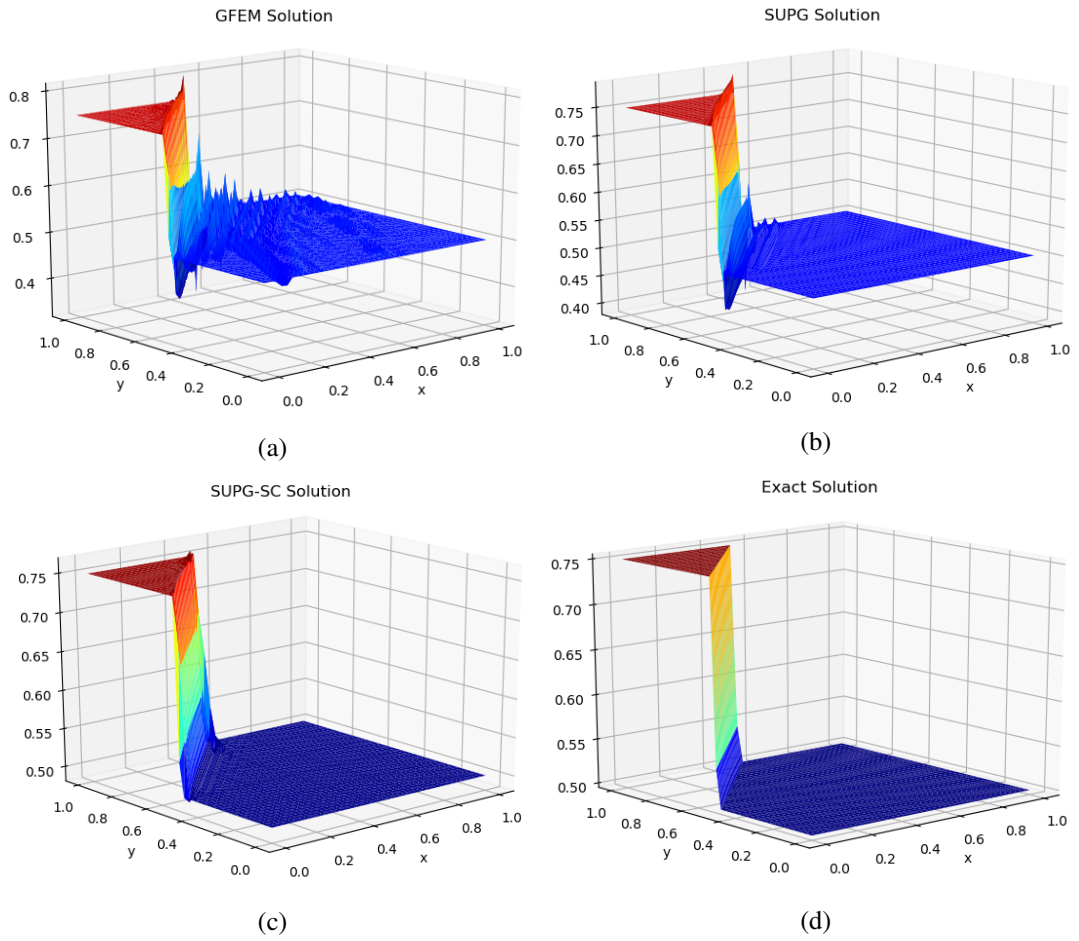
Figure 3.23: Solution of $u(x_1, x_2, t)$ in Eqs. (3.35)–(3.36) at $t_{\mathrm{f}} = 2.0$ s: (a) GFEM, (b) SUPG, (c) SUPG-YZβ, and (d) exact solution; $Re = 10^4$, and $\Delta t = 5.0 \times 10^{-3}$ s.
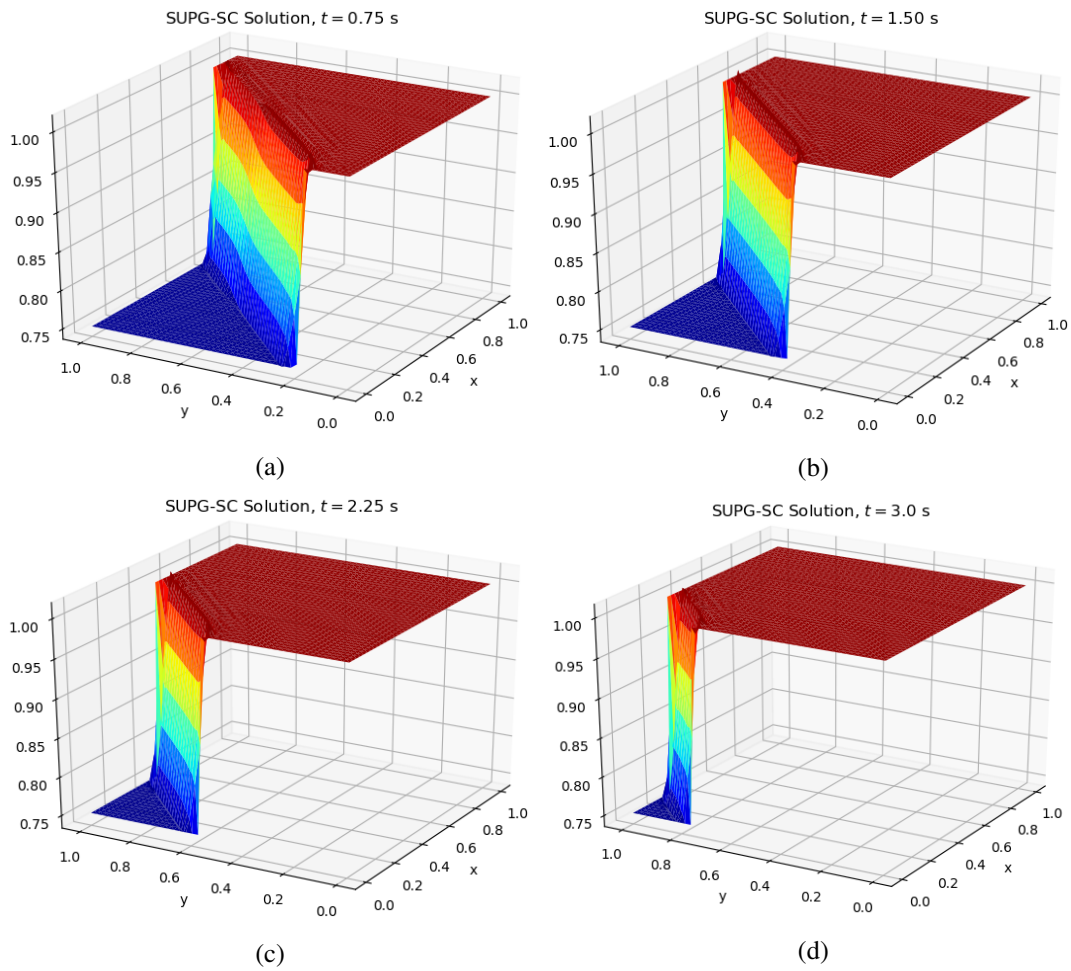
Figure 3.24: Solution of $v(x_1, x_2, t)$ in Eqs. (3.35)–(3.36) obtained with SUPG-YZβ at (a) $t_{\mathrm{f}} = 0.75$ s, (b) $t_{\mathrm{f}} = 1.5$ s, (c) $t_{\mathrm{f}} = 2.25$ s, and (d) $t_{\mathrm{f}} = 3.0$ s; $Re = 10^4$, and $\Delta t = 5.0 \times 10^{-3}$ s.
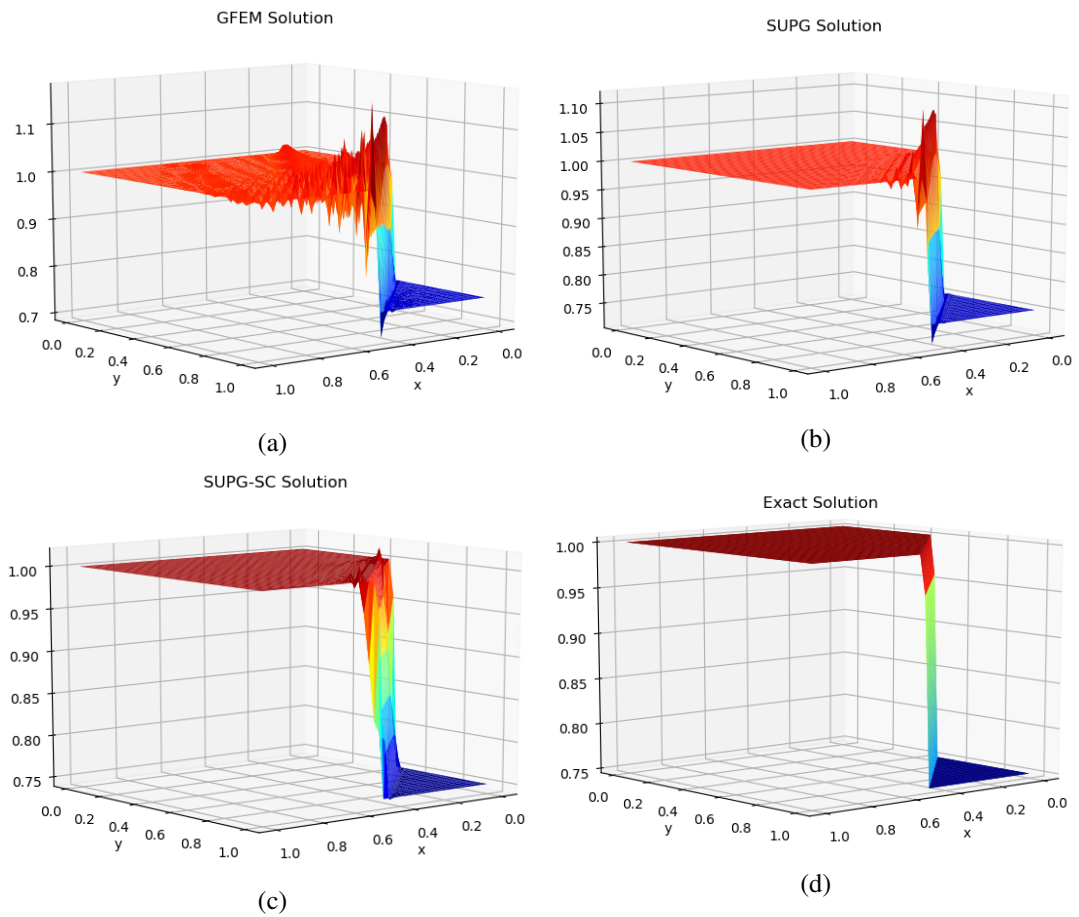
Figure 3.25: Solution of $v(x_1, x_2, t)$ in Eqs. (3.35)–(3.36) at $t_{\mathrm{f}} = 2.0$ s: (a) GFEM, (b) SUPG, (c) SUPG-YZβ, and (d) exact solution; $Re = 10^4$, and $\Delta t = 5.0 \times 10^{-3}$ s.
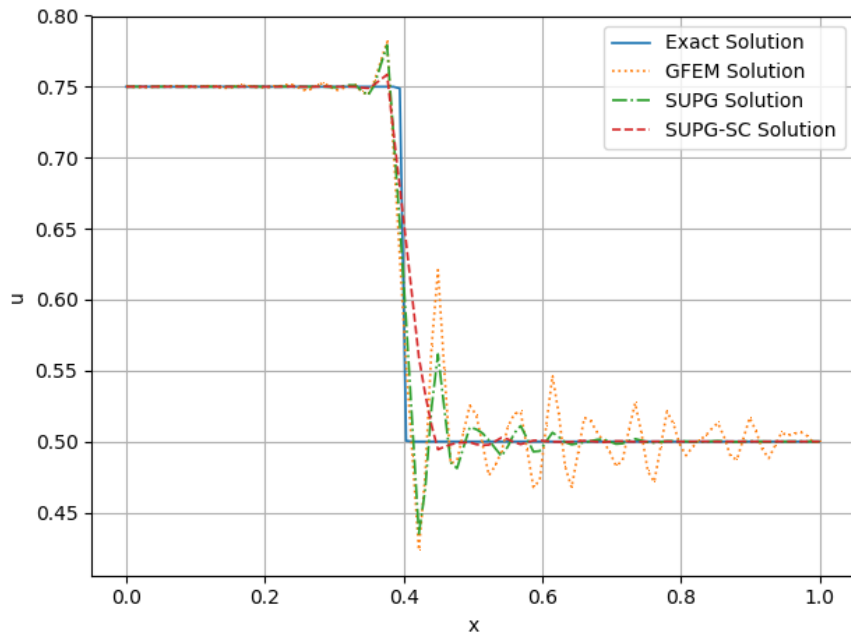
Figure 3.26: Comparison of solutions for $u(x_1, x_2, t)$ along line $x_2 = 0.9$ for solving Eqs. (3.35)–(3.36); $Re = 10^4$, $n_{\text{el}} = 3,528$, $t_{\text{f}} = 2.0$ s, and $\Delta t = 5.0 \times 10^{-3}$ s.
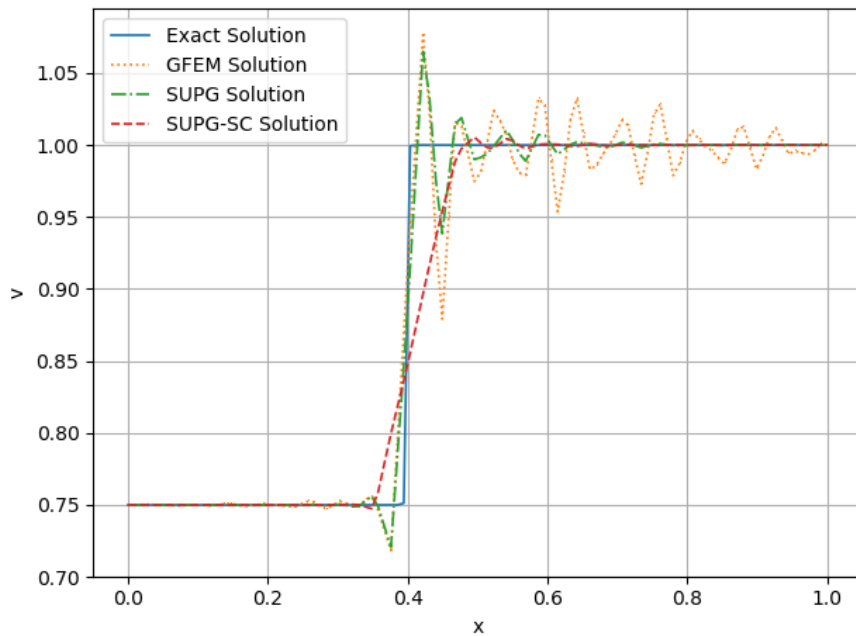
Figure 3.27: Comparison of solutions for $v(x_1, x_2, t)$ along line $x_2 = 0.9$ for solving Eqs. (3.35)–(3.36); $Re = 10^4$, $n_{\text{el}} = 3,528$, $t_{\text{f}} = 2.0$ s, and $\Delta t = 5.0 \times 10^{-3}$ s.
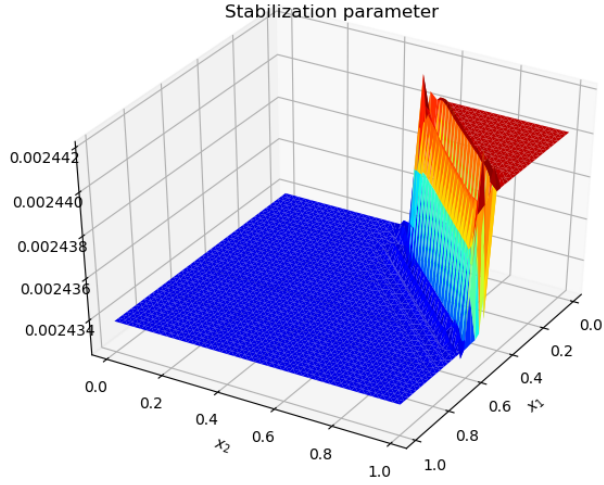
(a)
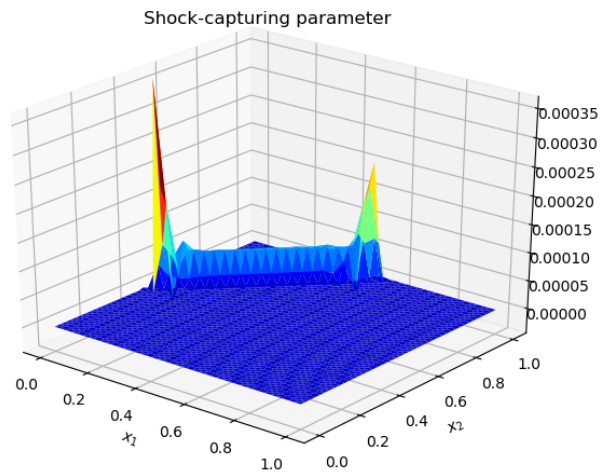
(b)
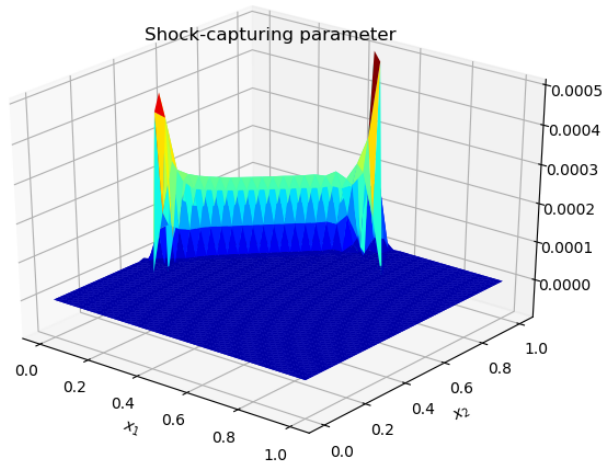
(c)

Figure 3.28: Stabilization and shock-capturing parameters in solving Eqs. (3.35)–(3.36): (a) Stabilization parameter, (b) shock-capturing parameter for u-component, and (c) shock-capturing parameter for v-component; $Re = 10^4$, $n_{\mathrm{el}} = 3,528$, $t_{\mathrm{f}} = 2.0$ s, and $\Delta t = 5.0 \times 10^{-3}$ s.

CHAPTER 4

COMPUTATIONAL SETUP FOR HIGH-SPEED FLOW SIMULATIONS

The compressible-flow SUPG formulations of the governing equations are introduced in this chapter, as well as the computational details of the simulations performed. To that end, the one-species thermochemical equilibrium equations are studied first, followed by the multispecies nonequilibrium equations. Subsequently, the computational details are provided.

4.1 SUPG Formulations of the Governing Equations

In this section, the SUPG formulations of the governing equations for equilibrium and nonequilibrium flows are given, the stabilization and shock-capturing parameters associated with these formulations are introduced.

4.1.1 SUPG Formulation: Thermochemical Equilibrium Flows

Let us recast the 2D one-species Euler equations of compressible flow (for details, see Section 2.3):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \qquad \text{in } \Omega \times I_t, \qquad (4.1)$$

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = \mathbf{0} \qquad \text{in } \Omega \times I_t, \qquad (4.2)$$

$$\frac{\partial (\rho e)}{\partial t} + \nabla \cdot (\rho \mathbf{u} h) = 0 \qquad \text{in } \Omega \times I_t. \qquad (4.3)$$

83

As given before by Eq. (2.112), the governing equations can be expressed in a quasi-linear form as

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_1 \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{A}_2 \frac{\partial \mathbf{U}}{\partial x_2} = \mathbf{0}. \tag{4.4}$$

Then, the GFEM formulation of Eq. (4.4) can be given as follows: find $\mathbf{U}^h \in \boldsymbol{\mathcal{S}}_{\mathbf{U}}^h$ such that for all test functions $\mathbf{W}^h \in \boldsymbol{\mathcal{V}}_{\mathbf{U}}^h$,

$$\int_\Omega \mathbf{W}^h \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} \right) d\Omega = 0, \tag{4.5}$$

where the solution and test function spaces are defined as follows:

$$\boldsymbol{\mathcal{S}}_{\mathbf{U}}^h = \{ \mathbf{U}^h : \mathbf{U}^h \in [H^{1h}(\Omega)]^{n_{\text{sd}}+2}, \mathbf{U}^h = \mathbf{G}(\mathbf{x}, t) \ \text{on} \ \Gamma^D \}, \tag{4.6}$$

$$\boldsymbol{\mathcal{V}}_{\mathbf{U}}^h = \{ \mathbf{W}^h : \mathbf{W}^h \in [H^{1h}(\Omega)]^{n_{\text{sd}}+2}, \mathbf{W}^h = \mathbf{0} \ \text{on} \ \Gamma^D \}. \tag{4.7}$$

Here, the finite-dimensional space $H^{1h}(\Omega)$ is defined as

$$H^{1h} = \{ \Phi^h : \Phi^h \in \mathcal{C}^0\left(\overline{\Omega}\right), \Phi^h|_{\Omega^e} \in \mathcal{P}_1\left(\Omega^e\right), \forall \Omega^e \in \mathcal{T}^h \}, \tag{4.8}$$

where $\mathcal{T}^h$ is the set of elements arising from the finite element discretization of the computational domain Ω, $\mathcal{C}^0\left(\overline{\Omega}\right)$ denotes the class of all continuous functions defined on the closure of Ω, the space $\mathcal{P}_k$ represents the set of polynomials of order at most k, and the function $\mathbf{G}(\mathbf{x}, t)$ is the Dirichlet boundary conditions associated with Eq. (4.4). It is assumed that $\overline{\Omega} = \bigcup_{\Omega^e \in \mathcal{T}^h} \overline{\Omega^e}$. The term Γ^D represents the part of boundary $\Gamma = \partial\Omega$ where Dirichlet-type boundary conditions apply.

The compressible-flow SUPG formulation of Eq. (4.4) is obtained by adding stabilization terms to the standard GFEM formulation as follows: find $\mathbf{U}^h \in \boldsymbol{\mathcal{S}}_{\mathbf{U}}^h$ such that for all test functions $\mathbf{W}^h \in \boldsymbol{\mathcal{V}}_{\mathbf{U}}^h$,

$$\int_\Omega \mathbf{W}^h \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} \right) d\Omega$$
$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \boldsymbol{\tau}_{\text{SUPG}} \left(\frac{\partial \mathbf{W}^h}{\partial x_1} \mathbf{A}_1^h + \frac{\partial \mathbf{W}^h}{\partial x_2} \mathbf{A}_2^h \right)$$
$$\cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} \right) d\Omega = 0. \tag{4.9}$$

The solution and test function spaces are defined as given in Eqs. (4.6)–(4.7), n_{el} is the number of elements, and e is the element counter. The stabilization parameter, $\boldsymbol{\tau}_{\text{SUPG}}$, is constructed from the following components [191–193]:

$$\tau_{\text{SUGN1}}^\rho = \tau_{\text{SUGN1}}^{\mathbf{u}} = \tau_{\text{SUGN1}}^e = \left(\sum_{a=1}^{n_{\text{en}}} |\mathbf{u}^h \cdot \nabla N_a| \right)^{-1}, \tag{4.10}$$

$$\tau^\rho_{\text{SUGN2}} = \tau^{\mathbf{u}}_{\text{SUGN2}} = \tau^e_{\text{SUGN2}} = \frac{\Delta t}{2}, \tag{4.11}$$

where Δt is the time-step size, the superscripts ρ, $\mathbf{u}$, and e denote the variables that the parameters are associated with, n_{en} is the number of element nodes, and N_a is the shape function associated with node a (for details, see Section 3.4.1). Then, the diagonal stabilization matrix is given as follows:

$$\boldsymbol{\tau}^{\text{SUPG}} = \begin{bmatrix} \tau^\rho & 0 & 0 & 0 \\ 0 & \tau^{\mathbf{u},x} & 0 & 0 \\ 0 & 0 & \tau^{\mathbf{u},y} & 0 \\ 0 & 0 & 0 & \tau^e \end{bmatrix}. \tag{4.12}$$

The stabilization parameters, using the "*r-switch*" combination [189], are given as

$$\tau^\rho = (\tau^\rho_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{(\tau^\rho_{\text{SUGN1}})^r} + \frac{1}{(\tau^\rho_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}}, \tag{4.13}$$

$$\tau^{\mathbf{u}} = (\tau^{\mathbf{u}}_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{(\tau^{\mathbf{u}}_{\text{SUGN1}})^r} + \frac{1}{(\tau^{\mathbf{u}}_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}}, \tag{4.14}$$

$$\tau^e = (\tau^e_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{(\tau^e_{\text{SUGN1}})^r} + \frac{1}{(\tau^e_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}}, \tag{4.15}$$

where, typically, $r = 2$ [189].

Notice that since the stabilizing terms added in the SUPG-stabilized formulation given by Eq. (4.9) include the residual of the original problem given by Eq. (4.4) as a factor, these added terms vanish when the exact solution is substituted into the SUPG-stabilized formulation.

The SUPG formulation of Eq. (4.4) supplemented with the YZβ shock-capturing is given as follows: find $\mathbf{U}^h \in \mathcal{S}^h_{\mathbf{U}}$ such that for all test functions $\mathbf{W}^h \in \mathcal{V}^h_{\mathbf{U}}$,

$$\int_\Omega \mathbf{W}^h \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}^h_1 \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}^h_2 \frac{\partial \mathbf{U}^h}{\partial x_2} \right) d\Omega$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \boldsymbol{\tau}_{\text{SUPG}} \left(\frac{\partial \mathbf{W}^h}{\partial x_1} \mathbf{A}^h_1 + \frac{\partial \mathbf{W}^h}{\partial x_2} \mathbf{A}^h_2 \right) \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}^h_1 \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}^h_2 \frac{\partial \mathbf{U}^h}{\partial x_2} \right) d\Omega$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \nu_{\text{SHOC}} \left(\frac{\partial \mathbf{W}^h}{\partial x_1} \cdot \frac{\partial \mathbf{U}^h}{\partial x_1} + \frac{\partial \mathbf{W}^h}{\partial x_2} \cdot \frac{\partial \mathbf{U}^h}{\partial x_2} \right) d\Omega = 0, \tag{4.16}$$

where the solution and test function spaces are defined as given in Eqs. (4.6)–(4.7).

The shock-capturing parameter, ν_{SHOC}, is from [191–193]:

$$\nu_{\text{SHOC}} = \left\| \mathbf{Y}^{-1}\mathbf{Z} \right\| \left(\sum_{i=1}^{n_{\text{sd}}} \left\| \mathbf{Y}^{-1}\frac{\partial \mathbf{U}^h}{\partial x_i} \right\|^2 \right)^{\frac{\beta}{2}-1} \left(\frac{h_{\text{SHOC}}}{2} \right)^{\beta}, \qquad (4.17)$$

where the element length scale, h_{SHOC}, is given as

$$h_{\text{SHOC}} = 2 \left(\sum_{a=1}^{n_{\text{en}}} |\mathbf{j} \cdot \nabla N_a| \right)^{-1} \qquad (4.18)$$

and

$$\mathbf{j} = \frac{\nabla \rho^h}{\|\nabla \rho^h\|}. \qquad (4.19)$$

The norm $\| \cdot \|$ and inner product "$\cdot$" stand for the standard Euclidean norm and inner product, respectively.

Tezduyar et al. reported in [191–193] that the parameter β of YZβ term can be set $\beta = 1$ for mild shocks, and set $\beta = 2$ for stronger shocks. A compromise can then be reached between the two situations by using an average as follows:

$$\nu_{\text{SHOC}} = \frac{1}{2} \left((\nu_{\text{SHOC}})_{\beta=1} + (\nu_{\text{SHOC}})_{\beta=2} \right). \qquad (4.20)$$

The scaling matrix $\mathbf{Y}$ reads

$$\mathbf{Y} = \text{diag}[(U_1)_{\text{ref}}, (U_2)_{\text{ref}}, (U_3)_{\text{ref}}, (U_4)_{\text{ref}}],$$

where the diagonal elements are reference values for ρ, $\rho\mathbf{u}$, and ρe. These reference values are set to the corresponding initial conditions (inflow values) but the x_2-component of the velocity vector since it is assumed to be zero. Instead, it is set as $(U_3)_{\text{ref}} = (U_2)_{\text{ref}}$, where $(U_2)_{\text{ref}}$ is the inflow value of ρu_1. Here, the term u_1 is the x_1-component of the velocity vector $\mathbf{u}$, and ρ is the gas density.

The vector $\mathbf{Z}$ can be defined in two ways:

$$\mathbf{Z} = \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} \qquad (4.21)$$

or

$$\mathbf{Z} = \frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2}. \qquad (4.22)$$

In this thesis, Eq. (4.22) is used in computations since we deal with time dependent simulations.

Note that the $\mathbf{Z}$ term of YZβ shock-capturing includes the residual of the original problem given by Eq. (4.4) as a factor. Recalling that the SUPG-stabilized formulation is also residual-based (see Eq. (4.9)), then one reaches the standard GFEM formulation when the exact solution is substituted into the SUPG-YZβ formulation given by Eq. (4.16).

4.1.2 SUPG Formulation: Thermochemical Nonequilibrium Flows

Let us recast the 2D multispecies Euler equations of compressible flow (see Section 2.4):

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \mathbf{u}) = \omega_s \qquad \text{in } \Omega \times I_t, \qquad (4.23)$$

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = \mathbf{0} \qquad \text{in } \Omega \times I_t, \qquad (4.24)$$

$$\frac{\partial (\rho e)}{\partial t} + \nabla \cdot (\rho \mathbf{u} h) = 0 \qquad \text{in } \Omega \times I_t, \qquad (4.25)$$

$$\frac{\partial (\rho e_V)}{\partial t} + \nabla \cdot (\rho \mathbf{u} e_V) = \omega_V \qquad \text{in } \Omega \times I_t. \qquad (4.26)$$

As given before by Eq. (2.121), the governing equations can be expressed in a quasi-linear form as

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_1 \frac{\partial \mathbf{U}}{\partial x_1} + \mathbf{A}_2 \frac{\partial \mathbf{U}}{\partial x_2} = \mathbf{S}. \qquad (4.27)$$

Then, the GFEM formulation of Eq. (4.27) can be given as follows: find $\mathbf{U}^h \in \mathcal{S}_\mathbf{U}^h$ such that for all test functions $\mathbf{W}^h \in \mathcal{V}_\mathbf{U}^h$,

$$\int_\Omega \mathbf{W}^h \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} - \mathbf{S}^h \right) d\Omega = 0, \qquad (4.28)$$

where the solution and test function spaces are defined as follows:

$$\mathcal{S}_\mathbf{U}^h = \{\mathbf{U}^h : \mathbf{U}^h \in [H^{1h}(\Omega)]^{n_{sd}+N_s+2}, \mathbf{U}^h = \mathbf{G}(\mathbf{x}, t) \text{ on } \Gamma^D\}, \qquad (4.29)$$

$$\mathcal{V}_\mathbf{U}^h = \{\mathbf{W}^h : \mathbf{W}^h \in [H^{1h}(\Omega)]^{n_{sd}+N_s+2}, \mathbf{W}^h = \mathbf{0} \text{ on } \Gamma^D\}. \qquad (4.30)$$

The term N_s denotes the number of species, and the finite-dimensional space $H^{1h}(\Omega)$ is defined by Eq. (4.8).

The compressible-flow SUPG formulation of Eq. (4.27) is obtained by adding stabilization terms to the standard GFEM formulation given by Eq. (4.28) as follows: find

$\mathbf{U}^h \in \boldsymbol{\mathcal{S}}_\mathbf{U}^h$ such that for all test functions $\mathbf{W}^h \in \boldsymbol{\mathcal{V}}_\mathbf{U}^h$,

$$
\int_\Omega \mathbf{W}^h \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} - \mathbf{S}^h \right) d\Omega
$$
$$
+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \boldsymbol{\tau}_{\text{SUPG}} \left(\frac{\partial \mathbf{W}^h}{\partial x_1} \mathbf{A}_1^h + \frac{\partial \mathbf{W}^h}{\partial x_2} \mathbf{A}_2^h \right)
$$
$$
\cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} - \mathbf{S}^h \right) d\Omega = 0, \quad (4.31)
$$

where the solution and test function spaces are defined as given in Eqs. (4.29)–(4.30), n_{el} is the number of elements and e is the element counter. The stabilization parameter, $\boldsymbol{\tau}_{\text{SUPG}}$, is constructed from the following components:

$$
\tau_{\text{SUGN1}}^{\rho,s} = \tau_{\text{SUGN1}}^\mathbf{u} = \tau_{\text{SUGN1}}^e = \tau_{\text{SUGN1}}^{e_V} = \left(\sum_{a=1}^{n_{\text{en}}} |\mathbf{u}^h \cdot \nabla N_a| \right)^{-1}, \quad (4.32)
$$

$$
\tau_{\text{SUGN2}}^{\rho,s} = \tau_{\text{SUGN2}}^\mathbf{u} = \tau_{\text{SUGN2}}^e = \tau_{\text{SUGN2}}^{e_V} = \frac{\Delta t}{2}. \quad (4.33)
$$

The superscripts ρ, $\mathbf{u}$, e, and e_V denote the variables that the parameters are associated with, as in the equilibrium case. Here, the difference between the equilibrium and nonequilibrium cases is that the stabilization matrix includes the stabilization parameters associated with the species densities ($\tau^{\rho,s}$'s) and vibrational-electron-electronic energy (τ^{e_V}). Then, the diagonal stabilization matrix is given as follows:

$$
\boldsymbol{\tau}_{\text{SUPG}} =
\begin{bmatrix}
\tau^{\rho,\mathbf{O}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \tau^{\rho,\mathbf{N}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \tau^{\rho,\mathbf{NO}} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \tau^{\rho,\mathbf{O_2}} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \tau^{\rho,\mathbf{N_2}} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \tau^{\mathbf{u},x} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \tau^{\mathbf{u},y} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \tau^e & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \tau^{e_V}
\end{bmatrix}. \quad (4.34)
$$

The stabilization parameters, using the "*r-switch*" concept [189], are given as

$$\tau^{\rho,s} = (\tau^{\rho,s}_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{(\tau^{\rho,s}_{\text{SUGN1}})^r} + \frac{1}{(\tau^{\rho,s}_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}}, \qquad (4.35)$$

$$\tau^{\mathbf{u}} = (\tau^{\mathbf{u}}_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{(\tau^{\mathbf{u}}_{\text{SUGN1}})^r} + \frac{1}{(\tau^{\mathbf{u}}_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}}, \qquad (4.36)$$

$$\tau^{e} = (\tau^{e}_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{(\tau^{e}_{\text{SUGN1}})^r} + \frac{1}{(\tau^{e}_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}}, \qquad (4.37)$$

$$\tau^{e_V} = (\tau^{e_V}_{\text{SUPG}})_{\text{UGN}} = \left(\frac{1}{(\tau^{e_V}_{\text{SUGN1}})^r} + \frac{1}{(\tau^{e_V}_{\text{SUGN2}})^r} \right)^{-\frac{1}{r}}, \qquad (4.38)$$

where, typically, $r = 2$.

The SUPG formulation of Eq. (4.27) supplemented with the YZβ shock-capturing is given as follows: find $\mathbf{U}^h \in \boldsymbol{\mathcal{S}}^h_{\mathbf{U}}$ such that for all test functions $\mathbf{W}^h \in \boldsymbol{\mathcal{V}}^h_{\mathbf{U}}$,

$$\int_\Omega \mathbf{W}^h \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}^h_1 \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}^h_2 \frac{\partial \mathbf{U}^h}{\partial x_2} - \mathbf{S}^h \right) d\Omega$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \boldsymbol{\tau}_{\text{SUPG}} \left(\frac{\partial \mathbf{W}^h}{\partial x_1} \mathbf{A}^h_1 + \frac{\partial \mathbf{W}^h}{\partial x_2} \mathbf{A}^h_2 \right) \cdot \left(\frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}^h_1 \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}^h_2 \frac{\partial \mathbf{U}^h}{\partial x_2} - \mathbf{S}^h \right) d\Omega$$

$$+ \sum_{e=1}^{n_{\text{el}}} \int_{\Omega^e} \nu_{\text{SHOC}} \left(\frac{\partial \mathbf{W}^h}{\partial x_1} \cdot \frac{\partial \mathbf{U}^h}{\partial x_1} + \frac{\partial \mathbf{W}^h}{\partial x_2} \cdot \frac{\partial \mathbf{U}^h}{\partial x_2} \right) d\Omega = 0, \quad (4.39)$$

where the solution and test function spaces are defined as given in Eqs. (4.29)–(4.30). In this formulation, the first line is the standard Galerkin finite element formulation, the second one is for stabilization, and the third is for shock-capturing.

The shock-capturing parameter, ν_{SHOC}, can be defined in a similar way that the equilibrium parameter is. The differences between the definitions of shock-capturing parameters given for equilibrium and nonequilibrium cases are in the scaling matrix, $\mathbf{Y}$, and in vector $\mathbf{Z}$. Here, the scaling matrix $\mathbf{Y}$ is defined as follows:

$$\mathbf{Y} = \text{diag}[(U_1)_{\text{ref}}, (U_2)_{\text{ref}}, \ldots, (U_9)_{\text{ref}}],$$

where the reference values are set to the corresponding initial conditions but the x_2-component of the velocity vector since it is assumed to be zero. Instead, it is set as $(U_7)_{\text{ref}} = (U_6)_{\text{ref}}$, where $(U_6)_{\text{ref}}$ is the inflow value of ρu_1. Here, u_1 is the x_1-component of the velocity vector $\mathbf{u}$, and ρ denotes the (gas) mixture density defined by Eq. (2.12).

Finally, the vector $\mathbf{Z}$ can be taken as

$$\mathbf{Z} = \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} - \mathbf{S}^h, \tag{4.40}$$

or

$$\mathbf{Z} = \frac{\partial \mathbf{U}^h}{\partial t} + \mathbf{A}_1^h \frac{\partial \mathbf{U}^h}{\partial x_1} + \mathbf{A}_2^h \frac{\partial \mathbf{U}^h}{\partial x_2} - \mathbf{S}^h. \tag{4.41}$$

We use Eq. (4.41) in nonequilibrium computations in a similar trend followed in [16].

Notice that both the SUPG-stabilized (see Eq. (4.31)) and SUPG-YZβ (see Eq. (4.39)) formulations are residual-based as in the equilibrium case, and thus, the SUPG-YZβ formulation is equivalent to the GFEM formulation given by Eq. (4.27) when the exact solution is substituted.

4.2 Computational Domain

The computational domain is shown in Figure 4.1, where $\mathbf{u}_\infty$ denotes the free-stream velocity. All essential boundary conditions are imposed at the inlet and are set to the free-stream conditions. No boundary condition is specified at the outlet since all the flow information can only travel downstream. At the top and bottom boundaries, $u_2 = 0$. The initial conditions are also set to the free-stream (inflow) conditions. The boundary conditions are studied in detail in the next section.
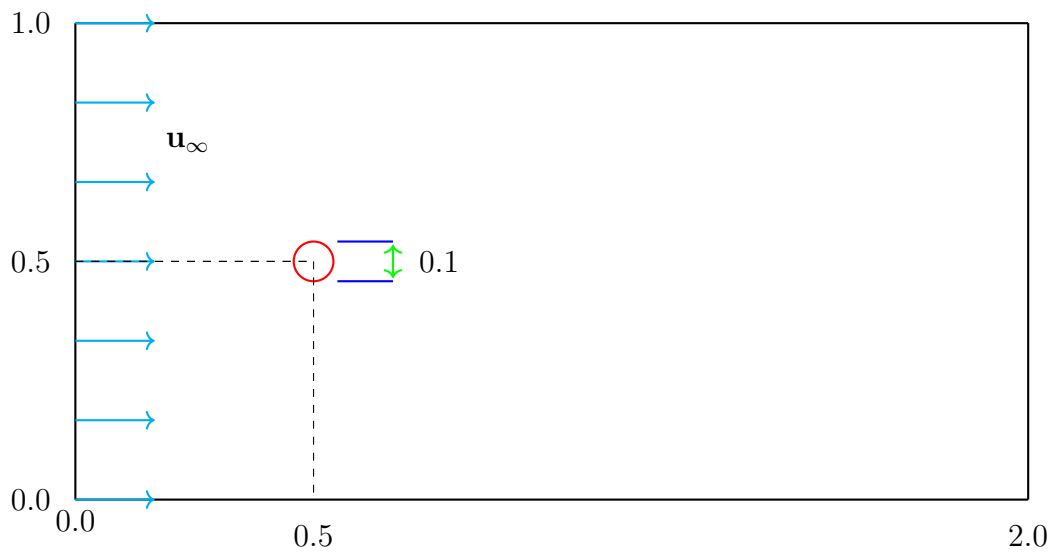
Figure 4.1: Computational domain. All dimensions are in meters.

The mesh is constructed in Gmsh [68] and has $6,059$ nodes and $11,816$ triangular elements (see Figure 4.2). Figure 4.3 shows the layers of constant-thickness elements near the cylinder. It is constructed by first creating constant-thickness element layers around the cylinder and then determining the number of element nodes on the walls.
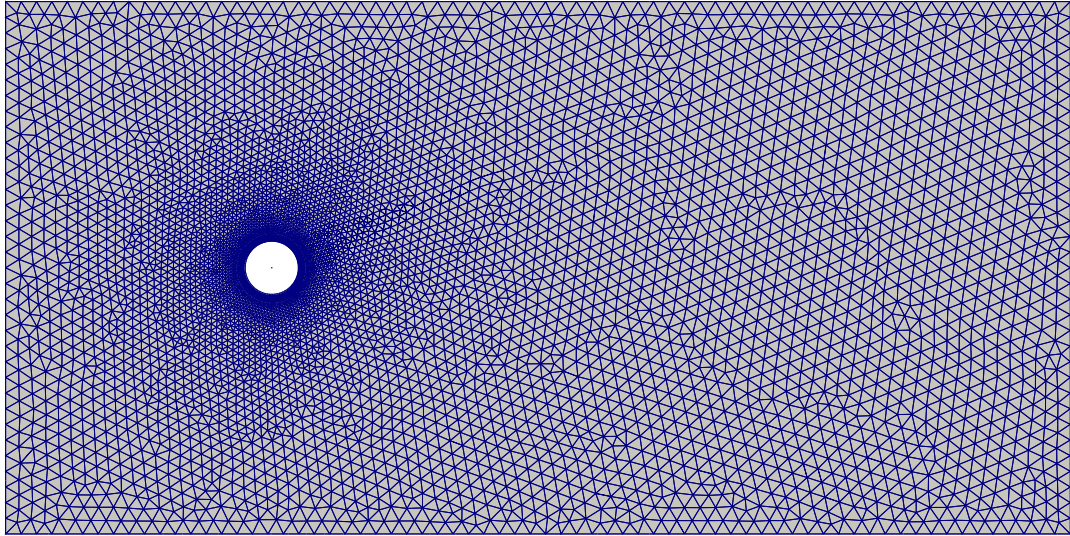
Figure 4.2: Mesh used in the computations. It has 6,059 nodes and 11,816 elements.
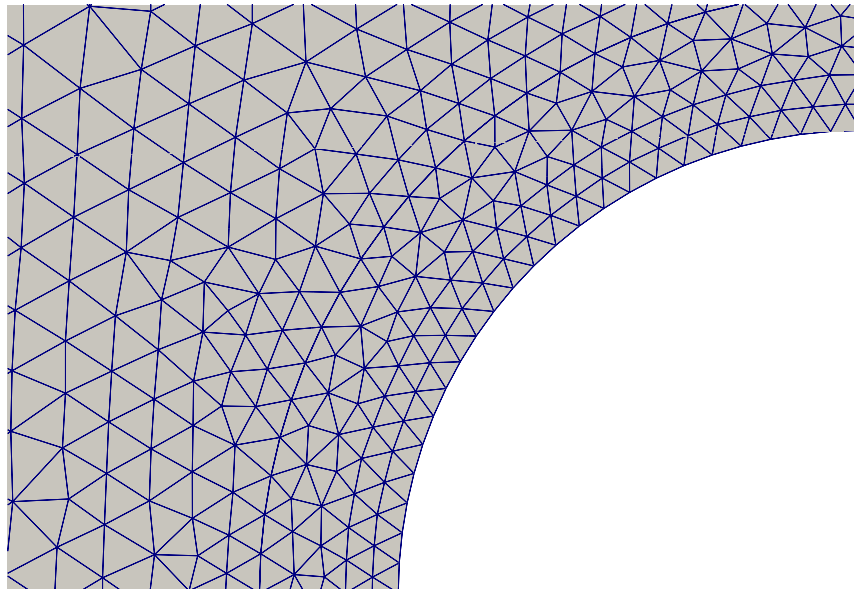
Figure 4.3: Mesh near the cylinder.

4.3 Boundary and Initial Conditions

The appropriate determination of the boundary and initial conditions is crucial in CFD simulations because the improper determination of these conditions may cause non-physical effects in simulations, as well as severe problems concerning the convergence speed and stability of the solutions. Various types of boundary conditions used in computations are briefly discussed in the following paragraphs.

4.3.1 Solid-Wall Conditions

In our case, the solid wall is a cylinder that is considered as a space vehicle's blunt nose. The details on how to apply the zero-normal-velocity condition is examined in Section 4.3.4. No wall temperature is specified on the cylinder surface since we deal with inviscid flows.

4.3.2 Far-Field Conditions

The far-field boundary conditions should have two characteristic properties [23]:

- the truncated (computational) domain should be a good representative of the real infinite domain,

- the computational domain should not reflect any effect into the flow field back.

The far-field conditions are classified as subsonic or supersonic depending on inflow (free-stream) and outflow velocities, resulting in four different scenarios. Let $\mathbf{u}_\infty$ represents the inflow velocity vector, $\mathbf{n}$ be the n_{sd}-dimensional outward-oriented unit normal vector to the boundary $\partial\Omega = \Gamma$ (see Figure 4.4 for a 2-dimensional illustration), and c_{sound} be the speed of sound. Then, an *inflow boundary (inlet)* is defined as the set of boundary points, i.e., $\mathbf{x} \in \partial\Omega$, such that $\mathbf{u}_\infty \cdot \mathbf{n} < 0$, and an *outflow boundary (outlet)* is defined as the set of boundary points such that $\mathbf{u}_\infty \cdot \mathbf{n} > 0$, where the dot product "·" represents the standard Euclidean inner product. Following that, classification of the boundary conditions is carried out as follows [23, 66, 144]:

- *Subsonic inflow*: $\mathbf{u}_\infty$ and $\mathbf{n}$ are in opposite directions with $|\mathbf{u}_\infty \cdot \mathbf{n}| < c_{\text{sound}}$. One characteristic variable is determined (extrapolated) from the interior (flow field) data. The rest is set to the inflow values.

- *Subsonic outflow*: $\mathbf{u}_\infty$ and $\mathbf{n}$ are in the same direction with $|\mathbf{u}_\infty \cdot \mathbf{n}| < c_{\text{sound}}$. One boundary condition (pressure) is required to be specified at the outlet. The others are determined (extrapolated) from the interior data.

- *Supersonic/hypersonic inflow*: $\mathbf{u}_\infty$ and $\mathbf{n}$ are in opposite directions with $|\mathbf{u}_\infty \cdot \mathbf{n}| > c_{\text{sound}}$. Then, all flow characteristics enter the flow domain. All boundary conditions should be specified at the inlet, setting as the inflow values.

- *Supersonic/hypersonic outflow*: $\mathbf{u}_\infty$ and $\mathbf{n}$ are in the same direction with $|\mathbf{u}_\infty \cdot \mathbf{n}| > c_{\text{sound}}$. Then, all flow characteristics leave the flow domain. No boundary condition needs to be specified at the outlet.
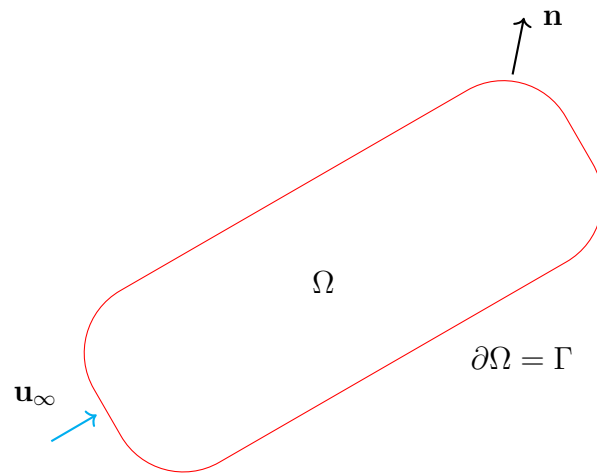
Figure 4.4: A 2D illustration of the inflow vector $\mathbf{u}_\infty$ and unit normal vector $\mathbf{n}$ to the boundary $\partial\Omega = \Gamma$.

In this study, all advective Jacobians have positive eigenvalues because both inflow and outflow velocities are assumed to be supersonic/hypersonic. Therefore, there is no need to specify any boundary condition at the outlet. However, all inflow values are imposed at the inlet. For further details, the interested reader can refer to Appendix B.

4.3.3 Initial Conditions

The initial conditions are determined according to the vehicle's speed and free-stream conditions where the flow occurs in the atmosphere. Properties of the air mixture and the corresponding Mach numbers for a vehicle moving at 5500 m/s are presented in Table 4.1. One can refer to the NASA Standard Atmosphere Report [197] for further details.

Table 4.1: The free-stream conditions and corresponding Mach numbers for 5500 m/s in Earth's atmosphere.

Altitude (km) H	Temperature (K) T_∞	Density (kg/m^3) ρ_∞	Speed of sound (m/s) c_{sound}	Mach M
39.60	249.930	4.08 $\times 10^{-3}$	316.49	17.38
50.00	270.650	7.87 $\times 10^{-4}$	329.80	16.68
60.50	244.050	2.70 $\times 10^{-4}$	313.17	17.56
71.00	214.650	6.42 $\times 10^{-5}$	293.70	18.73

The initial densities of the species are determined using their mass fractions in the air mixture. For instance, at an altitude of 50 km in the atmosphere, the initial values (densities) for oxygen and nitrogen are given as $\rho_{O_2} = 0.20946 \times 7.87 \times 10^{-4}$ kg/m^3 and $\rho_{N_2} = 0.78084 \times 7.87 \times 10^{-4}$ kg/m^3. Here, $Y_{O_2} = 0.20946$ and $Y_{N_2} = 0.78084$ are the mass fractions of oxygen and nitrogen at that altitude. For other species densities, i.e., for O, N, and NO, this value is set to zero. Then, the mixture density is obtained simply by summing the species densities as given in Eq. (2.12). The initial value of the velocity vector $\mathbf{u}$ is set $\mathbf{u} = (5500, 0)$ m/s that corresponds to different Mach numbers at different altitudes (see Table 4.1). Finally, the initial values of the energies e and e_V are determined by the free-stream temperature, T_∞, and velocity, $\mathbf{u}_\infty$.

4.3.4 Implementation of the Zero-Normal-Velocity Condition

Some CFD problems, such as the Navier–Stokes equations with free capillary boundaries [11], large-eddy simulations of turbulent flows [65], and inviscid flows around objects [145], require the enforcement of slip boundary conditions. Such boundary conditions are better suited than Dirichlet-type conditions for modeling many phenomena observed in nature, such as fluid penetration into pores during fluid flow in a

porous medium or the fact that hurricane vortices do not stick to the boundaries but move [91]. As these examples indicates, it is critical to take into account whether the fluid can pass through the boundaries. Such boundary conditions can be enforced (at least) four different ways [118]:

- a direct implementation,

- a Lagrange multiplier implementation,

- a penalty approach,

- replacing the "no-slip" condition with the "slip with friction" condition.

Here, we modify the model given in [91] that takes into account the penetration phenomenon in a similar way done in [145] to apply the zero-normal-velocity condition in a weak sense. In our case, the zero-normal-velocity (slip) boundary condition, i.e., $\mathbf{u} \cdot \mathbf{n} = 0$, applies on the cylinder surface. Algebraic manipulations can also be made in the assembled system of equations to enforce the condition strongly. Various approaches have been proposed for the direct enforcement of the slip boundary conditions, for example, Le Beau et al. [120] achieved it in a more direct way, by introducing a local coordinate rotation.

The test function $\mathbf{w}_u^h \subset \mathbf{W}^h$, corresponding to the momentum equation, can be decomposed into its orthonormal components on the cylinder surface [91, 118]:

$$\mathbf{w}_u^h = \left(\mathbf{w}_u^h \cdot \mathbf{n} \right) \mathbf{n} + \sum_{k=1}^{n_{\mathrm{sd}}-1} \left(\mathbf{w}_u^h \cdot \boldsymbol{\tau}_k \right) \boldsymbol{\tau}_k, \qquad (4.42)$$

where the tangential vectors, $\boldsymbol{\tau}_k$, are chosen such that the set $\{\mathbf{n}, \boldsymbol{\tau}_1, \dots \boldsymbol{\tau}_{n_{\mathrm{sd}}-1}\}$ builds an orthonormal system of vectors in n_{sd}-dimensions. Then, by the definitions of the slip with linear friction and penetration with resistance boundary conditions, the following boundary integrals are added to the variational formulation [91]:

$$\int_{\Gamma_{\mathrm{CYL}}} \alpha^{-1} \left(\mathbf{u}^h \cdot \mathbf{n} \right) \left(\mathbf{w}_u^h \cdot \mathbf{n} \right) \, d\Gamma + \int_{\Gamma_{\mathrm{CYL}}} \eta \sum_{k=1}^{n_{\mathrm{sd}}-1} \left(\mathbf{u}^h \cdot \boldsymbol{\tau}_k \right) \left(\mathbf{w}_u^h \cdot \boldsymbol{\tau}_k \right) \, d\Gamma. \qquad (4.43)$$

Here, the positive constants α and η represent the penetration and friction parameters, respectively, and Γ_{CYL} denotes the cylinder surface. As α tends to zero, the cylinder surface stands as fully impermeable. If $\alpha \to 0$ and $\eta \to \infty$, then the well-known

"no-slip" (homogeneous Dirichlet) boundary condition of the viscous flows (governed by Navier–Stokes equations) is prescribed. In this thesis, we enforce the zero-normal-velocity condition by adding the term

$$\int_{\Gamma_{\text{CYL}}} \alpha^{-1} \left(\mathbf{u}^h \cdot \mathbf{n} \right) \left(\mathbf{w}_u^h \cdot \mathbf{n} \right) \, d\Gamma \tag{4.44}$$

to the variational formulations. We also use a weak implementation of the zero-normal-velocity condition by adding the term

$$-\int_{\Gamma_{\text{CYL}}} \left(\mathbf{u}^h \cdot \mathbf{n} \right) \left(\mathbf{W}^h \cdot \mathbf{U}^h + w_e^h p^h \right) \, d\Gamma \tag{4.45}$$

to the variational formulations, as done in [174], where $w_e^h \subset \mathbf{W}^h$ is the test function corresponding to the energy equation.

The interested reader may also refer to [97] for more details on enforcing general boundary conditions, including slip boundary conditions.

4.4 Temporal Discretization

The time-stepping is performed by using the backward Euler method. As we step from time level n to $n + 1$,

$$\frac{\partial \mathbf{U}}{\partial t} \approx \frac{\mathbf{U}_{n+1} - \mathbf{U}_n}{\Delta t_n}. \tag{4.46}$$

The backward Euler scheme is an implicit method because the rest of the terms in discrete formulations resulting from substitution of Eq. (4.46) into spatially discretized (semi-discrete) formulations are also expressed in terms of time level $n + 1$.

Hyperbolic PDEs have *wave-like* solutions, which means that if a disturbance is introduced into the initial data, it does not affect every point in space (flow field) at the same time; instead, disturbances propagate at a finite speed relative to a fixed time coordinate (for more on hyperbolic PDEs, see Appendix C). Therefore, the time-step size must be kept at less than a certain limit when solving hyperbolic problems numerically; otherwise, stability issues related to the numerical scheme may arise.

The Courant–Friedrichs–Lewy (CFL), a necessary condition for stability, was introduced in a 1928 paper by Courant, Friedrichs, and Lewy [47]. It can be expressed as given in [114] by Laney: *"The full numerical domain of dependence must contain*

the physical domain of dependence." This expression says without using technical terms: "*The time-step should be kept small enough so that the information (wave) has enough time to propagate (travel) through the space discretization.*" In other words, in simulating unsteady problems, as stepping from one time-step to the next one, it must be ensured that the information is conveyed to the adjacent cells (elements) entirely within these consecutive time steps.

In our computations, we follow a similar way to that done in [145]. For a given Courant number, $C_{\Delta t}$, the local time-step size $(\Delta t_n)^e$ for element e is determined from the CFL condition:

$$(\Delta t_n)^e = \frac{C_{\Delta t} h^e}{\lambda^e}, \tag{4.47}$$

where h^e is the smallest edge length of the element, and λ^e is defined as

$$\lambda^e = \max_a (c_a^e + \|\mathbf{u}_a^e\|). \tag{4.48}$$

Here, c_a^e and $\mathbf{u}_a^e$ are the acoustic speed and velocity at element node a, and the norm is the standard Euclidean norm. Then, the time-step size, Δt_n, is determined as the smallest value among all the elements globally:

$$\Delta t_n = \min_e (\Delta t_n)^e. \tag{4.49}$$

For the first time-step, we use the initial conditions to estimate λ^e. In this study, we set $C_{\Delta t} = 10.0$ for equilibrium flows and $C_{\Delta t} = 1.0$ for nonequilibrium flows (see, e.g., [67, 119, 202]).

It is worth noting that, instead of the mesh shown in Figure 4.2, finer meshes can help for resolving strong gradients and capturing shocks more accurately, potentially obviating the need for stabilization and shock-capturing techniques. Finer meshes, on the other hand, result in smaller mesh elements, necessitating smaller time-step sizes. Therefore, the computational time significantly increases due to the increasing number of elements and smaller time steps.

Finally, note that as the Mach number (M) increases, the term λ^e also increases, and consequently, the time-step size (Δt_n) decreases. It is clear that smaller time-step sizes mean more iterations.

4.5   Numerical Integration

The solver environment, FEniCS, is able to determine an appropriate quadrature rule and the minimum number of points required for numerical integration through the Unified Form Language technology (see Section 4.8). If the integrand is a polynomial, FEniCS uses the exact quadrature rule automatically. However, when integrating the nonlinear terms (e.g., ratios of polynomials, transcendental functions, etc.) in high-speed flow computations, we manually set the quadrature degree to four, despite the fact that FEniCS can determine (estimate) an appropriate value automatically.

4.6   Solution of System of Algebraic Equations

This section studies the methods and techniques used for solving algebraic equation systems arising from the space and time discretizations of the governing equations. To this end, since the resulting algebraic systems are nonlinear, the Newton–Raphson (N–R) nonlinear solver is introduced first. The generalized minimal residual algorithm is then presented for solving the linearized systems resulting from the N–R process. Finally, preconditioning strategies are discussed briefly, with a particular emphasis on incomplete LU factorization.

4.6.1   Nonlinear Solver

The N–R method is a classical method for solving a system of nonlinear algebraic equations. The N–R process linearizes the system of nonlinear algebraic equations at hand using an initial estimate. If a good starting (initial) solution can be guessed, the N–R algorithm works almost flawlessly.

Let $\mathbf{u}$ be the vector of unknowns and $\mathbf{f}(\mathbf{u})$ be the vector of nonlinear equations, i.e.,

$$\mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix}, \quad \mathbf{f}(\mathbf{u}) = \begin{bmatrix} f_1(\mathbf{u}) \\ \vdots \\ f_N(\mathbf{u}) \end{bmatrix}, \tag{4.50}$$

where $\mathbf{f}(\mathbf{u}) = \mathbf{0}$, and N denotes the number of unknowns and equations. Then, the N–

R method for solving a system of nonlinear equations, $\mathbf{f}(\mathbf{u}) = \mathbf{0}$, can be described as follows: for a given initial vector $\mathbf{u}_0$, a vector $\Delta\mathbf{u}_0$ is sought so that $\mathbf{f}(\mathbf{u}_0 + \Delta\mathbf{u}_0) = \mathbf{0}$. By making use of the first-order Taylor series expansion, one obtains the following approximation:

$$\mathbf{0} = \mathbf{f}(\mathbf{u}_0 + \Delta\mathbf{u}_0) \approx \mathbf{f}(\mathbf{u}_0) + \frac{\partial\mathbf{f}}{\partial\mathbf{u}}(\mathbf{u}_0)\Delta\mathbf{u}_0, \qquad (4.51)$$

where the term $\mathbf{J} = \frac{\partial\mathbf{f}}{\partial\mathbf{u}}$ represents the Jacobian of $\mathbf{f}$ with respect to $\mathbf{u}$:

$$\mathbf{J} = \frac{\partial\mathbf{f}}{\partial\mathbf{u}} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \cdots & \frac{\partial f_1}{\partial u_N} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_N}{\partial u_1} & \frac{\partial f_N}{\partial u_2} & \cdots & \frac{\partial f_N}{\partial u_N} \end{bmatrix}. \qquad (4.52)$$

After simple algebra, the vector $\Delta\mathbf{u}_0$ can be given as

$$\Delta\mathbf{u}_0 \approx -\left[\mathbf{J}_0\right]^{-1}\mathbf{f}(\mathbf{u}_0), \qquad (4.53)$$

where $\mathbf{J}_0 = \mathbf{f}(\mathbf{u}_0)$. Note that the new vector $\mathbf{u}_0 + \Delta\mathbf{u}_0$ represents the next iteration, i.e., $\mathbf{u}_1 = \mathbf{u}_0 + \Delta\mathbf{u}_0$. This process can be carried out until a desired tolerance between two consecutive iterations is achieved. Then, the $(n + 1)$th iteration of the N–R algorithm can be given as follows:

$$\mathbf{u}_{n+1} = \mathbf{u}_n - \left[\mathbf{J}_n\right]^{-1}\mathbf{f}(\mathbf{u}_n), \qquad (4.54)$$

where $\mathbf{J}_n = \frac{\partial\mathbf{f}}{\partial\mathbf{u}}(\mathbf{u}_n)$.

It should be noted that the inverse of the Jacobian matrix $\mathbf{J}_n$ is mostly a dense matrix and should not be stored due to extensive memory usage. Besides, the algebraic operations required to compute the inverse matrices at each N–R iteration may increase the computation time considerably. Dealing with inverse matrices is, in a nutshell, an expensive process. In fact, Eq. (4.54) represents a system of linear equations and can be expressed in the form of $\mathbf{A}\mathbf{x} = \mathbf{b}$. Consequently, one can exploit the material presented in the next section for solving such (linear) systems. For more on the nonlinear solvers (e.g., quasi-Newton update methods), one can refer to [55].

4.6.2 Linear Solver

In general, there are two types of approaches for solving a system of linear equations: *direct solution methods* (e.g., Gauss elimination, inverse matrix method, matrix factor-

ization methods) and *iterative solution methods* (e.g., Jacobi method, Gauss–Siedel method, conjugate gradient method, etc.). The use of direct solution methods would be very expensive because the problems discussed in this thesis lead to relatively large systems. The direct solution methods also lead to fill-in (a phenomenon that occurs during matrix operations and causes the sparse structure of the matrices to be distorted) [55]. Therefore, using iterative methods is often preferable over direct methods.

In this thesis, the generalized minimal residual (GMRES) search technique is employed for solving linear systems. The GMRES was developed by Saad and Schultz in [163] as a generalization of the minimal residual method (MINRES) introduced by Paige and Saunders in [149]. The MINRES and GMRES do the same calculations in exact arithmetic for symmetric problems. Since the MINRES prone to suffer from orthogonality loss much more than the GMRES, the GMRES can be described as the best implementation of the MINRES. The following paragraphs explain the GMRES technique briefly.

Consider a system of linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$, where the coefficient matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is assumed to be a large, sparse, nonsingular, and nonsymmetric, and vector $\mathbf{b} \in \mathbb{R}^n$ is given. One of the most eminent and reliable classes of iterative methods for solving such a system is the Krylov subspace projection. Krylov methods seek approximations in the form of [162]

$$\mathbf{A}^{-1}\mathbf{b} \approx \mathbf{x}_k = \mathbf{x}_0 + \mathcal{K}_k\left(\mathbf{A}, \mathbf{r}_0\right) \tag{4.55}$$

satisfying the *Petrov–Galerkin orthogonality* condition

$$\left(\mathbf{b} - \mathbf{A}\mathbf{x}_k\right) \perp \mathcal{L}_k, \tag{4.56}$$

where $\mathcal{L}_k$ is a k-dimensional subspace other than the affine subspace $\mathbf{x}_0 + \mathcal{K}_k$, and $\mathbf{x}_0 \in \mathbb{R}^n$ is an appropriate initial guess. Different selections of the subspace $\mathcal{L}_k$ result in different methods, e.g., if $\mathcal{L}_k = \mathbf{A}\mathcal{K}_k\left(\mathbf{A}, \mathbf{r}_0\right)$, then the projection is called the *generalized minimal residual (GMRES)* algorithm, where the Krylov subspace is defined as follows [162]:

$$\mathcal{K}_k\left(\mathbf{A}, \mathbf{r}_0\right) = \mathcal{K}_k = \text{span}\left\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \ldots, \mathbf{A}^{k-1}\mathbf{r}_0\right\}. \tag{4.57}$$

Here, the initial residual vector, $\mathbf{r}_0$, is defined as $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$. As the iteration number k increases, it is natural that the set of vectors $\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \mathbf{A}^2\mathbf{r}_0, \ldots, \mathbf{A}^{k-1}\mathbf{r}_0$

spanning the Krylov subspace $\mathcal{K}_k$ lose orthogonality (become linearly dependent) and this causes accuracy issues. Therefore, an orthonormal basis spanning the Krylov subspace $\mathcal{K}_k$ is constructed with the help of an orthogonal projection method, e.g., Arnoldi's method, Householder process, Gram–Schmidt algorithm.

Recall that the GMRES computes the kth approximation (iteration) as given in Eq. (4.55) such that $(\mathbf{b} - \mathbf{A}\mathbf{x}_k) \perp \mathbf{A}\mathcal{K}_k(\mathbf{A}, \mathbf{r}_0)$. Then, the following relation is equivalent to Eq. (4.55):

$$\mathbf{x}_k = \operatorname*{argmin}_{\mathbf{x}_0 \in (\mathbf{x}_0 + \mathcal{K}_k)} \|\mathbf{b} - \mathbf{A}\mathbf{x}\| = \mathbf{x}_0 + \mathbf{V}_k \mathbf{y}_k, \tag{4.58}$$

where the matrix $\mathbf{V}_k$ is constructed from the columns that form an orthonormal basis for $\mathcal{K}_k$, and the vector $\mathbf{y}_k$ is defined as

$$\mathbf{y}_k = \operatorname*{argmin}_{\mathbf{y} \in \mathbb{R}^k} \|\mathbf{r}_0 - \mathbf{A}\mathbf{V}_k\mathbf{y}\|. \tag{4.59}$$

Letting $\beta = \|\mathbf{r}_0\|$ and $\mathbf{v}_1 = \mathbf{r}_0/\beta$, one finds

$$\mathbf{r}_0 - \mathbf{A}\mathbf{V}_k\mathbf{y} = \mathbf{V}_{k+1}\left(\beta\mathbf{e}_1 - \mathbf{H}_{k+1,k}\mathbf{y}\right), \tag{4.60}$$

where the matrix $\mathbf{H}_{k+1,k}$ is defined as

$$\mathbf{H}_{k+1,k} = \begin{bmatrix} \mathbf{H}_k \\ h_{k+1,k}\mathbf{e}_k^T \end{bmatrix}. \tag{4.61}$$

Here, $\mathbf{H}_k$ is the Hessenberg matrix of size $k \times k$, and the vector $\mathbf{e}_k$ represents the kth column of the identity matrix. Then, the original problem turns into the following least squares problem:

$$\mathbf{x}_k = \mathbf{x}_0 + \mathbf{V}_k\mathbf{y}_k, \text{ where} \tag{4.62}$$

$$\mathbf{y}_k = \operatorname*{argmin}_{\mathbf{y} \in \mathbb{R}^k} \|\beta\mathbf{e}_1 - \mathbf{H}_{k+1,k}\mathbf{y}\|. \tag{4.63}$$

Computing the minimizer of Eq. (4.63), $\mathbf{y}$, is not expensive since it requires the solution of the least squares problem of size $(k+1) \times k$, which is much smaller than the size of the original problem $(n \times n)$.

Notice that, at each GMRES iteration, the dimension of the Krylov subspace increases by one. This situation also increases the amount of memory required and the computing time. Fortunately, restarting techniques can be used to overcome this challenge. The

restarting process, on the other hand, may cause the convergence rate to slow down and stagnate. Again, fortunately, preconditioning techniques (see Section 4.6.3) can be exploited to accelerate the convergence rates. For further details, the interested reader can refer to [162].

4.6.3 Preconditioning

As any scientific computing or numerical analysis textbook would address, the efficiency of the methods used in solving a specific problem depends primarily on the nature of the problem, e.g., convection dominance, the degrees of freedom of the system, etc. This is also the case for the methods for solving systems of algebraic equations. For this reason, there is a need for special techniques that do not change the solution of the system at hand but improve the convergence characteristics, called the *preconditioning techniques*. These techniques generally work as follows: let the following system of linear equations is given:

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{4.64}$$

where the matrix $\mathbf{A}$ is a non-singular matrix. Then, the solution of the system

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b}, \tag{4.65}$$

and that of the original system are theoretically the same. The matrix $\mathbf{M}$ here should be chosen so that the convergence properties provided by the matrix $\mathbf{M}^{-1}\mathbf{A}$ are better than the properties provided by the matrix $\mathbf{A}$. A large class of preconditioning strategies is based on the lower upper (LU) factorization method, i.e., $\mathbf{M} = \mathbf{L}\mathbf{U}$, where the matrices $\mathbf{L}$ and $\mathbf{U}$ are lower and upper factorization matrices of $\mathbf{A}$. Then, Eq. (4.65) becomes

$$\mathbf{U}^{-1}\mathbf{L}^{-1}\mathbf{A}\mathbf{x} = \mathbf{U}^{-1}\mathbf{L}^{-1}\mathbf{b}. \tag{4.66}$$

Since an exact $\mathbf{L}\mathbf{U}$ factorization of matrix $\mathbf{A}$ usually results in a lot of fill-in, an incomplete version of this process, known as ILU, is used to generate approximate factorization matrices that limit the fill-in in exchange for not getting an exact factorization [55].

In summary, nonlinear equation systems encountered in hypersonic flow computations in this thesis are handled with the Newton–Raphson method. The GMRES

update technique, an iterative Krylov subspace method, is used to solve the linear equation systems. And finally, the GMRES technique is supplemented with ILU preconditioning.

Although the solver environment, FEniCS, uses direct (Gaussian elimination) methods by default unless otherwise specified, it allows many modern and classical techniques. The FEniCS book [126] can be examined for many examples for solving systems of linear and nonlinear algebraic equations in the FEniCS environment.

4.7 Computation of Vibrational-Electronic Temperature

Although it has easy access to a lot of information concerning hypersonic flow simulations, getting clear information about some stages, particularly how to compute the temperatures in Park's two-temperature model, is both time-consuming and tedious. Therefore, it is desired to address this issue under a separate section.

Let us consider the governing equations of nonequilibrium flows, Eqs. (2.115)–(2.118), introduced in Section 2.4. Also, recall the vibrational energy equation given in Section 2.1.3 by Eq. (2.44):

$$
e_s^{vib} = \begin{cases} \dfrac{R_s \theta_{v,s}}{\exp\left(\frac{\theta_{v,s}}{T_v}\right) - 1}, & \text{if } s \text{ is molecule}, \\[2em] 0, & \text{if } s \text{ is atom}. \end{cases}
$$

Here, the vibrational temperature, T_v, and the electronic temperature, T_e, are assumed to be the same by Eq. (2.46), i.e., $T_V = T_v = T_e$. To obtain the vibrational-electronic temperature, T_v, the governing equations (2.115)–(2.118) are solved, and the vibrational-electron-electronic energy e_V is obtained. Then, Eq. (2.44) is solved for T_V, with a nonlinear solver.

The translational-rotational temperature, T, is straightforward to compute since Eq. (2.42), given in Section 2.1.3, is linear in T:

$$
e_s^{tr} = e_s^{trans} + e_s^{rot} = C_{tr,v,s} T.
$$

103

4.8 The FEniCS Project

In this study, all computations are carried out in the FEniCS environment. FEniCS is a `C++`/`Python` library for solving (partial) differential equations and performing scientific computations. The acronym, FEniCS, is frequently referred to as "Finite Elements nurtured in Computer Science" or "For Everything new in Computational Science." The project was launched in 2003 at the University of Chicago with the goal of changing the state-of-the-art in scientific computing.

The FEniCS project is open-sourced and available for free, i.e., it can be easily downloaded, used, and modified. Its main advantage over traditional finite element packages is its ability to generate automated codes for local finite element tensors using high-level mathematical abstraction. Besides, the mathematical formulations and their corresponding FEniCS codes are so similar that usually no additional effort is required to use FEniCS, thanks to the Unified Form Language technology (UFL) [4].

Instead of comprising certain physical and mathematical models, FEniCS allows the user to set models and perform scientific computations through the Just-in-Time (JIT) compilations. In terms of solving PDEs by employing the FEM, FEniCS is the first platform in which the user can utilize all types of finite element spaces, interpolation functions, and degrees of freedom for discretizing the problem under consideration spatially [9].

At the early times, FEniCS had only two core components: the Dynamic Object-oriented Library for FINite element computation (DOLFIN) [128] and FInite element Automatic Tabulator (FIAT) [105]. Today, it includes various modern software components such as the FEniCS Form Compiler (FFC) [127], Unified Form-assembly Code (UFC) [3], Unified Form Language (UFL) [4], and Symbolic Finite Elements (SyFi) [5]. The interested reader can find more on FEniCS from various perspectives in [1, 115, 126].

CHAPTER 5

2D HYPERSONIC FLOW SIMULATIONS

The materials preceding this chapter aim to form a background for the computations and simulations performed here. The basic concepts and characteristics of hypersonic flows are discussed in Chapter 1. Following that, Chapter 2 studies the governing equations of compressible flows in thermochemical equilibrium/nonequilibrium in detail. The stabilization and shock-capturing concepts exploited here for compressible flow computations are tested on a comprehensive set of "relatively simpler" problems in Chapter 3, demonstrating both the need for and the success of such methods and techniques. Finally, the details concerning the computational setup in performing high-speed flow simulations here have been explained in Chapter 4. One can also refer to the Appendices for complementary and supplementary materials and discussions, i.e., the physical and chemical constants used in computations, a brief introduction to the characteristics of compressible Euler equations, and the existence and uniqueness of solutions.

In this chapter, thermochemical equilibrium/nonequilibrium flow simulations at supersonic and hypersonic speeds are performed. First, one-species nitrogen (N_2) flows in thermochemical equilibrium are considered. Since the flow field is assumed to be in thermochemical equilibrium, the one-species assumption is natural. The choice of N_2 is based on the fact that it is the major constituent of Earth's atmosphere in terms of the molar fraction. Next, the simulations of a five-species (O, N, NO, O_2, N_2) gas mixture in thermochemical nonequilibrium are carried out. Again, this five-species gas mixture is chosen because oxygen (O_2) is the secondary major component of Earth's atmosphere in terms of volume; thus, it is reasonable that this set of five species would

be a good representative for nonequilibrium simulations when chemical reactions are also taken into account. For free-stream velocities between 8 and 10 km/s, the five-species model is widely accepted and used in the literature [70]. For further details, one can refer to Section 1.2.1.

Since the simulations obtained with the weak implementation of the slip boundary condition on the cylinder surface, employing Eq. (4.45), and the penalty approach, using Eq. (4.44), do not contain significant differences in terms of the flow characteristics, only the simulations performed using the penalty approach are presented. However, we investigate how the weak formulation given by Eq. (4.45) for enforcing the zero-normal-velocity boundary condition works near the cylinder surface and compare the results obtained for various Mach numbers.

For all computations performed here, the relative error tolerances for Newton–Raphson and GMRES iterations are set to 1.0×10^{-6}. For more on the solution of algebraic equation systems, the interested reader can refer to Section 4.6.

5.1 Hypersonic Nitrogen Flow in Equilibrium

Recall the governing equations of 2D one-species hypersonic nitrogen (N_2) flow introduced in Section 2.3 by Eqs. (2.107)–(2.109):

$$\frac{\partial \rho_{N_2}}{\partial t} + \nabla \cdot (\rho_{N_2} \mathbf{u}) = 0, \tag{5.1}$$

$$\frac{\partial \rho_{N_2} \mathbf{u}}{\partial t} + \nabla \cdot (\rho_{N_2} \mathbf{u} \otimes \mathbf{u}) + \nabla p = \mathbf{0}, \tag{5.2}$$

$$\frac{\partial \rho_{N_2} e}{\partial t} + \nabla \cdot (\rho_{N_2} \mathbf{u} h) = 0, \tag{5.3}$$

where the system is equipped with boundary conditions introduced in Section 4.2. Note that the compressible-flow SUPG formulation of Eqs. (5.1)–(5.3) supplemented with YZβ shock-capturing is presented by Eq. (4.16) in Section 4.1.1.

The test computations are carried out for three different Mach numbers: $M = 2.0$, $M = 5.0$, and $M = 8.0$. In all three cases, the initial density and temperature are set to the free-stream values, i.e., $\rho_\infty = 1.165$ kg/m^3 and $T_\infty = 300.0$ K. We set the penalty parameter of zero-normal-velocity boundary condition as $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg. In the YZβ term, $\beta = 2$. All three computations are performed until $t = 1.0 \times 10^{-3}$ s.

The system of algebraic equations resulting from the space and time discretizations of Eqs. (5.1)–(5.3) has $24,236$ unknowns. One can refer to Section 4.6 for more on the linear and nonlinear solvers used in computations. We apply the CFL condition by setting the Courant number as $C_{\Delta t} = 10.0$ (see Section 4.4).

For all cases, test simulations show that despite very high speeds and strong gradients occurring in solutions, the compressible-flow SUPG formulation supplemented with YZβ shock-capturing works well in yielding shock representations without any non-physical oscillations. Although the results obtained here are in a quite agreement with those obtained in [159], more rapid decreases in the magnitude of the velocity vector are observed when applying the zero-normal-velocity condition with the penalty term.

5.1.1   Mach 2.0

We set the inflow velocity vector as $\mathbf{u}_\infty = (706.13, 0.0)$ m/s. Figure 5.1 shows the elevation plots for the pressure, density, and temperature. Figures 5.2a–5.2c show the velocity magnitude, density, and temperature around the cylinder.

At the stagnation point, the gas density rises to around 3.4 kg/m^3 while it drops to 0.5 kg/m^3 right behind the cylinder. The temperature increases in regions where the velocity magnitude decreases, as expected, and this increasing temperature spreads to the cylinder's front and rear. The temperature and pressure, like density, reach their highest values at the stagnation point.

It is seen that the temperature around the cylinder is about 600 K. As discussed in Section 1.2.2, for temperatures below 800 K, the gas mixture can be assumed to be calorically perfect gas, and chemical reactions can be ignored. Therefore, the thermochemical equilibrium assumption is still valid for such speeds.

The Mach elevation and line plots are shown in Figure 5.3, revealing that the penetration parameter of the slip boundary condition, which is set to $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg in computations, works well for $M = 2.0$. Figure 5.3 illustrates that the velocity vector completely vanishes at the stagnation point. Notice that for $\alpha = 1.0 \times 10^{-1}$ m^2·s/kg and $\alpha = 1.0 \times 10^{-2}$ m^2·s/kg, the penalty approach yields unacceptable results, i.e., impermeability of the cylinder surface cannot be achieved.

Figure 5.4 illustrates the behavior of the Courant number ($C_{\Delta t_n}$) as time evolves for $M = 2.0$. At each time step, Courant numbers stay below 1.0, indicating the choice of $C_{\Delta t} = 10.0$ works well at $M = 2.0$.

The average number of Newton–Raphson iterations is 4. The number of time steps required to reach the final time is 33.

Figure 5.1: Density (kg/m^3), pressure (P), and temperature (K) at $M = 2.0$.

(a)

(b)

(c)

Figure 5.2: (a) Velocity (m/s), (b) density $(\mathrm{kg/m^3})$, and (c) temperature (K) at $M = 2.0$.

110
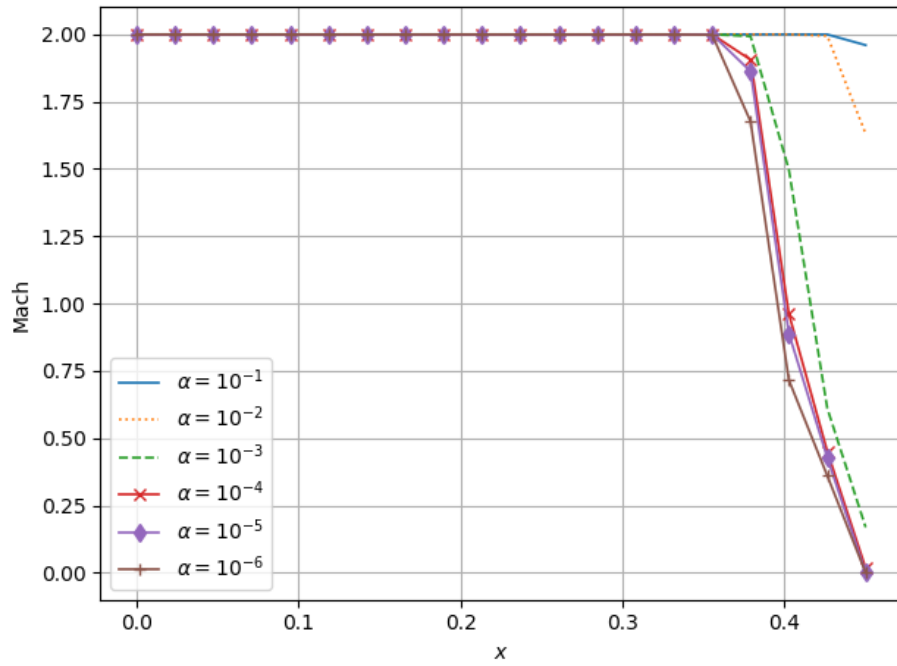
Figure 5.3: Mach elevation and line plots in front of the cylinder (along the stagnation line) at $M = 2.0$ for various values of penetration parameter α.
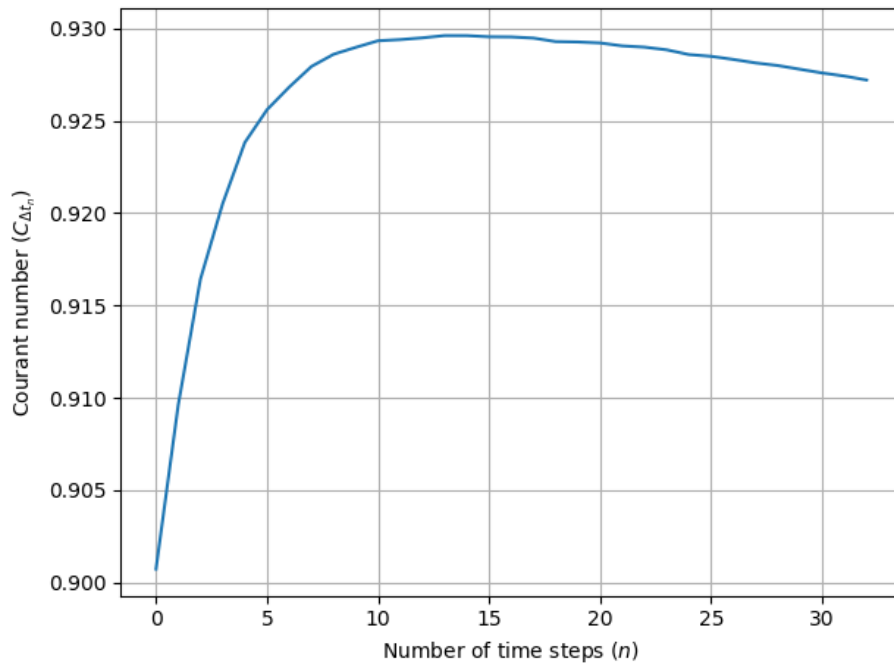
Figure 5.4: Courant number ($C_{\Delta t_n}$) versus number of time steps (n) for $M = 2.0$.

5.1.2 Mach 5.0

In computations, the free-stream velocity vector is taken as $\mathbf{u}_\infty = (1765.33, 0)$ m/s. Figure 5.5 shows the elevation plots for the pressure, density, and temperature. Figures 5.6a–5.6c show the velocity magnitude, density, and temperature around the cylinder.

The gas density rises to around 5.5 kg/m^3 at the stagnation point and drops below 1.0 kg/m^3 behind the cylinder. As expected, the temperature increases in areas where the velocity magnitude falls, and this increasing temperature spreads to the cylinder's front and back. At the stagnation point, temperature and pressure, as well as density, reach their highest values.

It can be seen that the temperature around the cylinder is about $2,000$ K. At these temperatures, the gas mixture can be no longer assumed to be calorically perfect gas, and as discussed in Section 1.2.2, dissociation processes begin for O_2 molecules. Therefore, the chemical reactions that may occur in the flow field should also be taken into consideration.

The Mach elevation and line plots are shown in Figure 5.7, revealing that the penetration parameter of the zero-normal-velocity boundary condition, which is set to $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg in computations, works well for $M = 5.0$. As can be seen in Figure 5.7, the velocity vector completely vanishes at the stagnation point. Note that for $\alpha = 1.0 \times 10^{-1}$ m^2·s/kg, $\alpha = 1.0 \times 10^{-2}$ m^2·s/kg, and $\alpha = 1.0 \times 10^{-3}$ m^2·s/kg, the penalty approach gives unacceptable results. Although it is observed that the flow penetrates the cylinder very slightly for $\alpha = 1.0 \times 10^{-4}$ m^2·s/kg, it is seen that the velocity vector disappears completely at the stagnation point for $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg and smaller values of the penalty parameter.

Figure 5.8 illustrates the behavior of the Courant number ($C_{\Delta t_n}$) as time evolves for $M = 5.0$. Except for the initial steps, Courant numbers stay below 1.0, demonstrating the selection of $C_{\Delta t} = 10.0$ works well at $M = 5.0$.

The number of time steps required to reach the final time is 65. The average number of Newton–Raphson iterations at each time-step is 5.
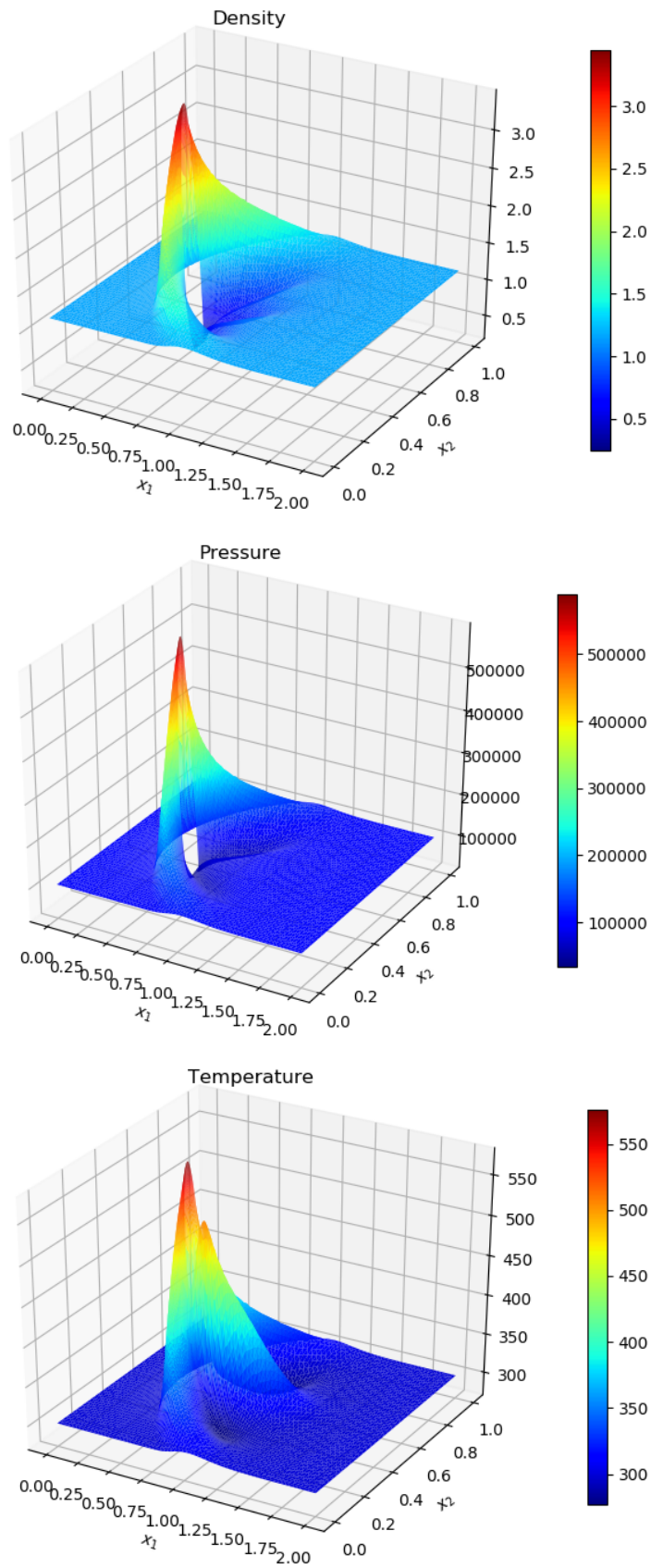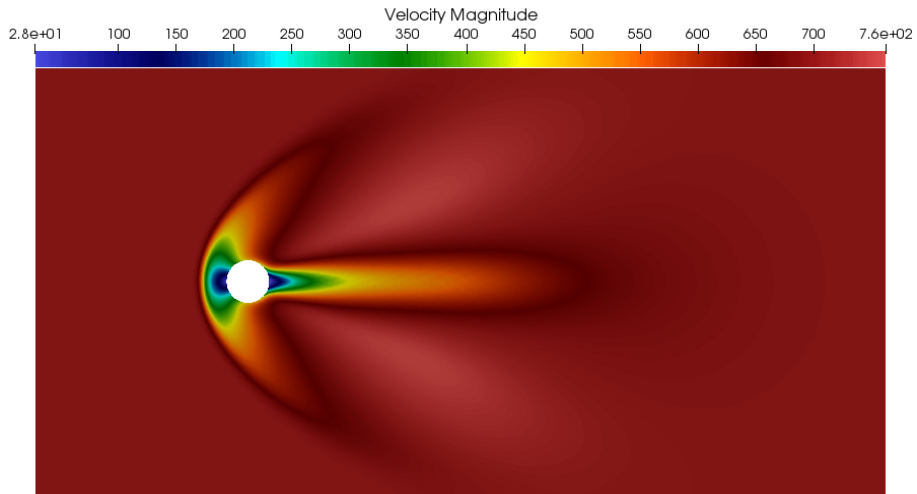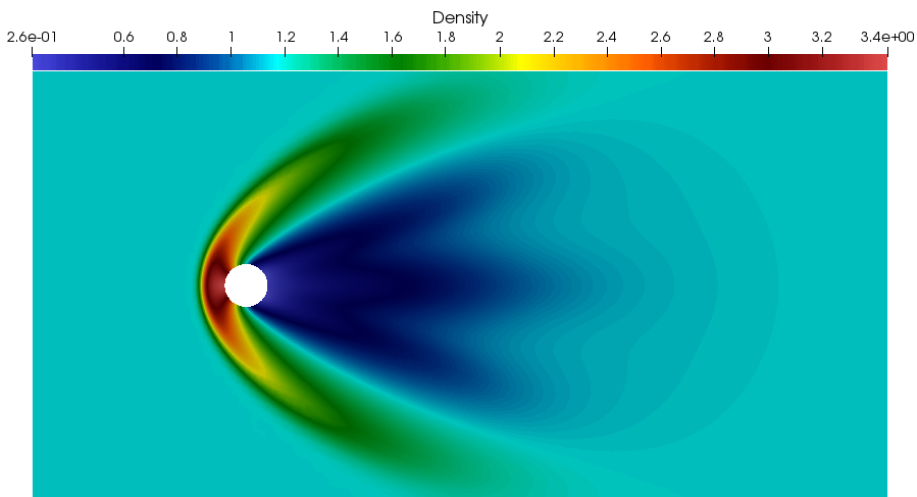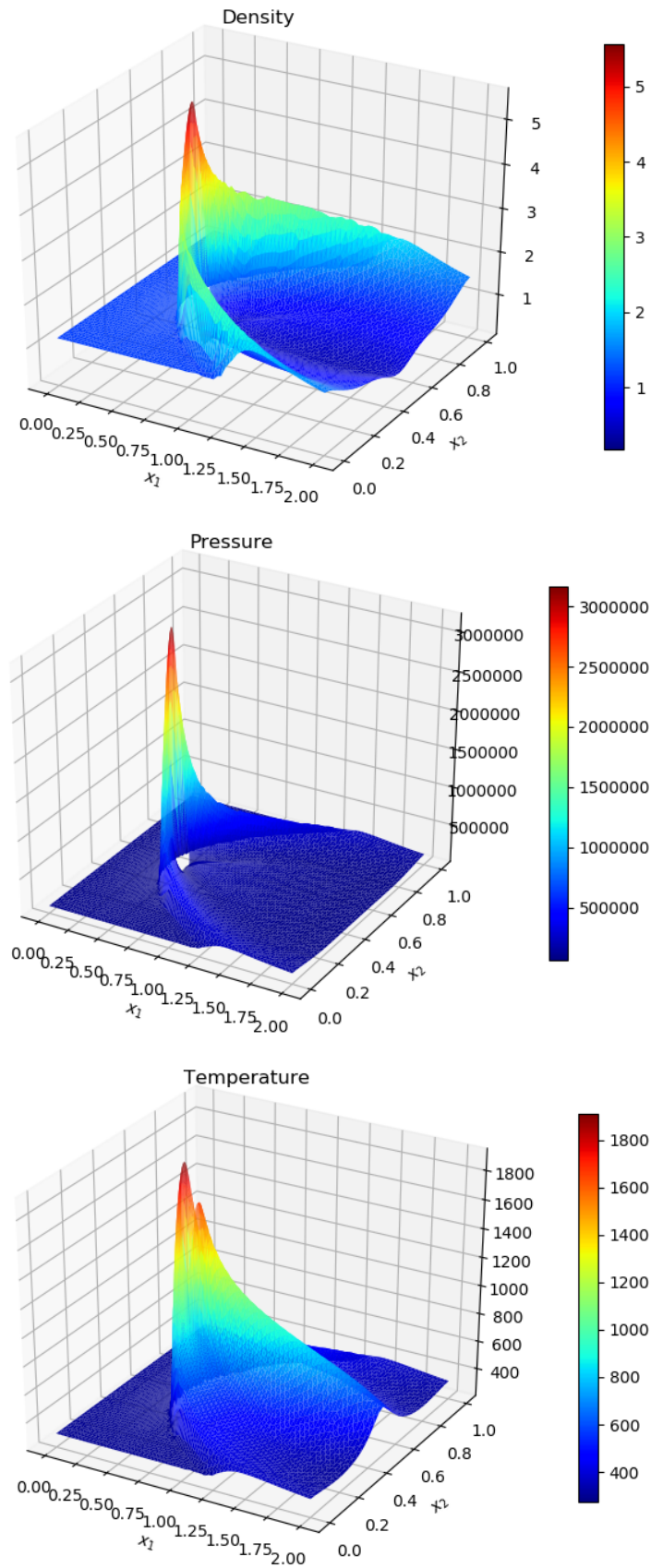
Figure 5.5: Density (kg/m³), pressure (Pa), and temperature (K) at $M = 5.0$.
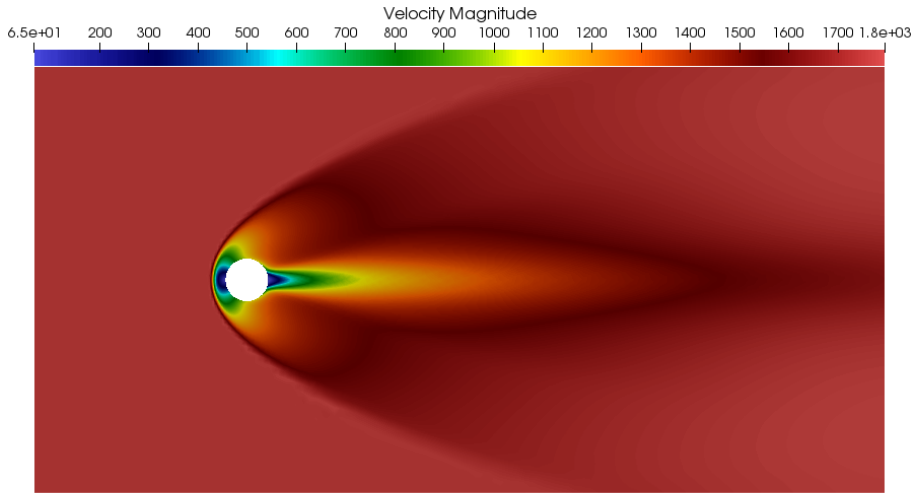
(a)
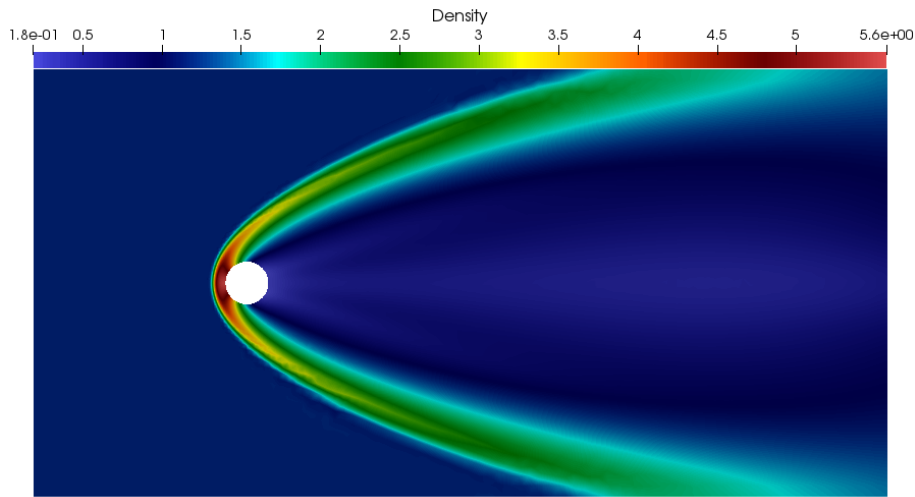
(b)
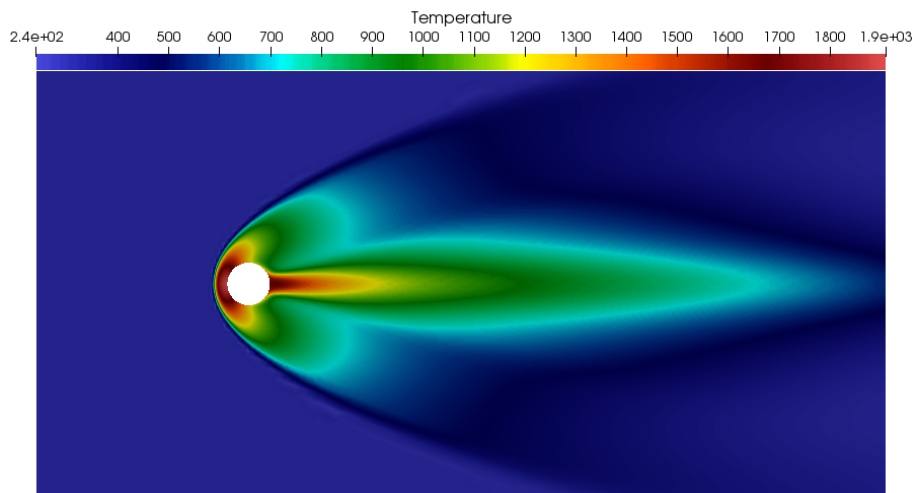
(c)

Figure 5.6: (a) Velocity (m/s), (b) density (kg/m^3), and (c) temperature (K) at $M = 5.0$.

Figure 5.7: Mach elevation and line plots in front of the cylinder (along the stagnation line) at $M = 5.0$ for various values of penetration parameter α.
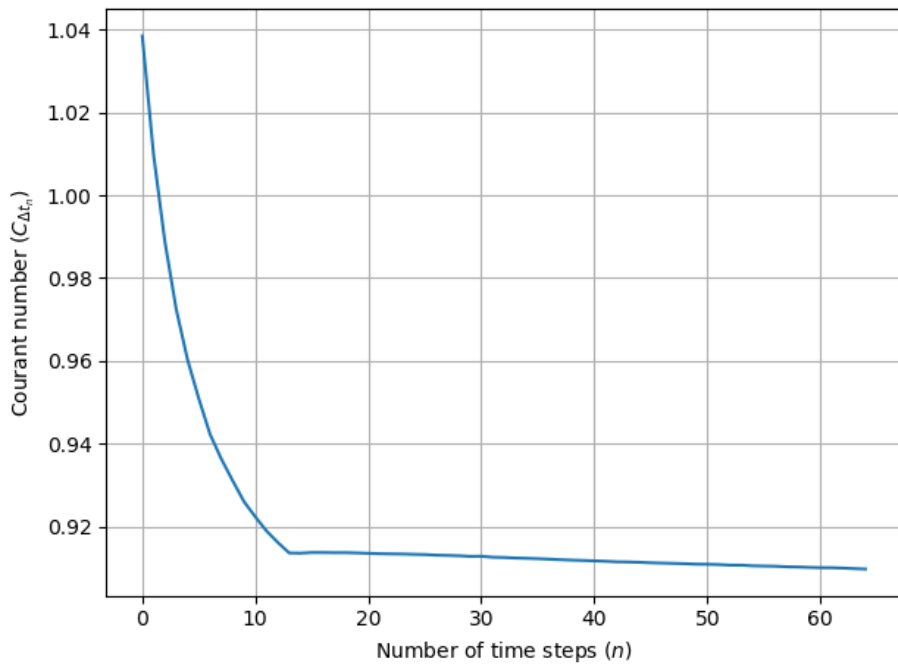
Figure 5.8: Courant number ($C_{\Delta t_n}$) versus number of time steps (n) for $M = 5.0$.

5.1.3 Mach 8.0

The inflow velocity vector is set as $\mathbf{u}_\infty = (2824.53, 0)$ m/s. Figure 5.9 shows the elevation plots for the pressure, density, and temperature. Figures 5.10a–5.10c show the velocity magnitude, density, and temperature around the cylinder.

The gas density rises to around 6.0 kg/m^3 at the stagnation point before falling under 1.0 kg/m^3 behind the cylinder. As expected, the temperature arises in regions where the velocity magnitude decreases, and this increasing temperature spreads to the cylinder's front and back. At the stagnation point, temperature and pressure, like density, reach their highest levels.

One can see in Figure 5.9 that the temperature around the cylinder is about $4,500$ K (it is about $4,600$ K at the stagnation point). At these temperatures, it is known that O_2 molecules are completely dissociated, and the same processes begin for N_2 molecules above temperatures $4,000$ K, as discussed in Section 1.2.2. It is worth noting that the ionization processes should also be taken into account for even higher temperatures.

The Mach elevation and line plots are shown in Figure 5.11, revealing that the penalty parameter of the impermeable boundary condition, which is set to $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg in computations, works well for $M = 8.0$. As in the $M = 2.0$ and $M = 5.0$ cases, it is seen in Figure 5.7 that the velocity vector completely vanishes at the stagnation point. Note that for $\alpha = 1.0 \times 10^{-1}$ m^2·s/kg, $\alpha = 1.0 \times 10^{-2}$ m^2·s/kg, and $\alpha = 1.0 \times 10^{-3}$ m^2·s/kg, the penalty approach results in unacceptable velocity profiles. It is observed that the flow penetrates the cylinder very slightly for $\alpha = 1.0 \times 10^{-4}$ m^2·s/kg. The velocity vector vanishes completely at the stagnation point for $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg and smaller values of the penalty parameter.

Figure 5.12 illustrates the behavior of the Courant number ($C_{\Delta t_n}$) as time evolves for $M = 8.0$. As in the $M = 5.0$ case, except for the first few time steps, Courant numbers stay below 1.0, indicating the choice of $C_{\Delta t} = 10.0$ works well at $M = 8.0$, as well.

The number of time steps to reach the final time is 98. The average number of Newton–Raphson iterations at each time-step is 6.
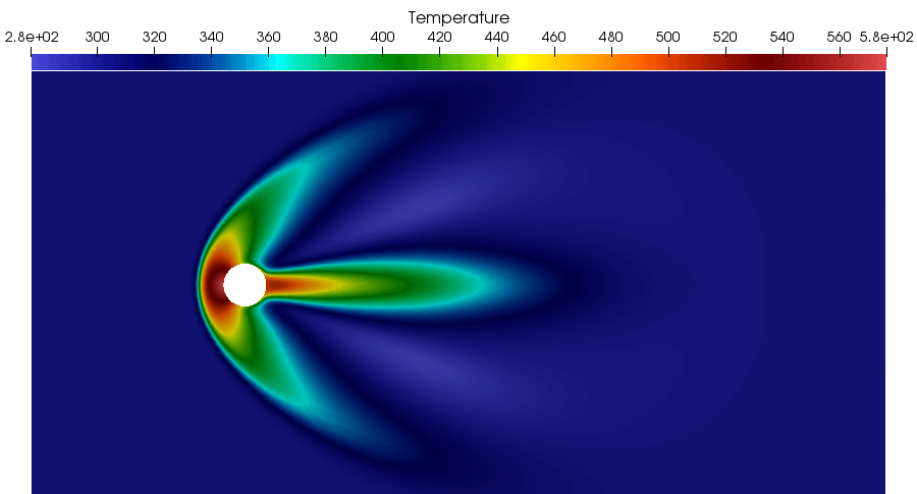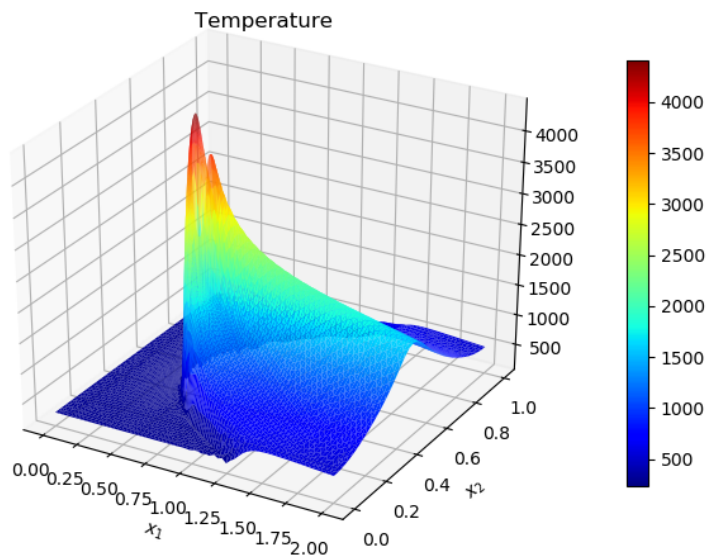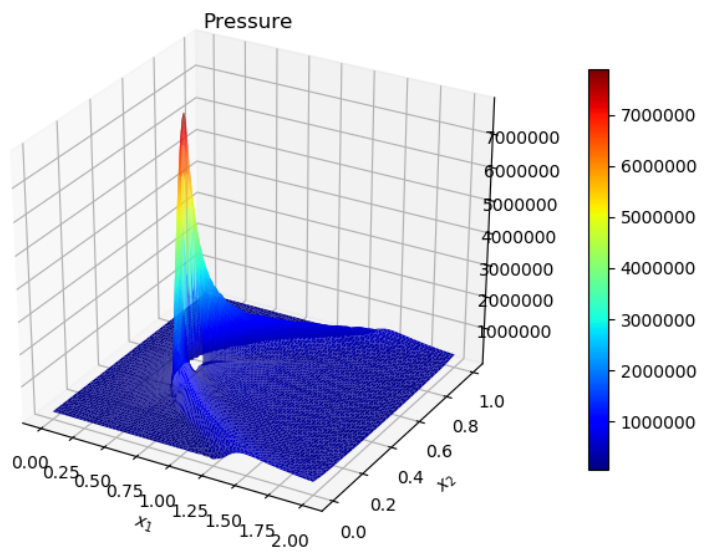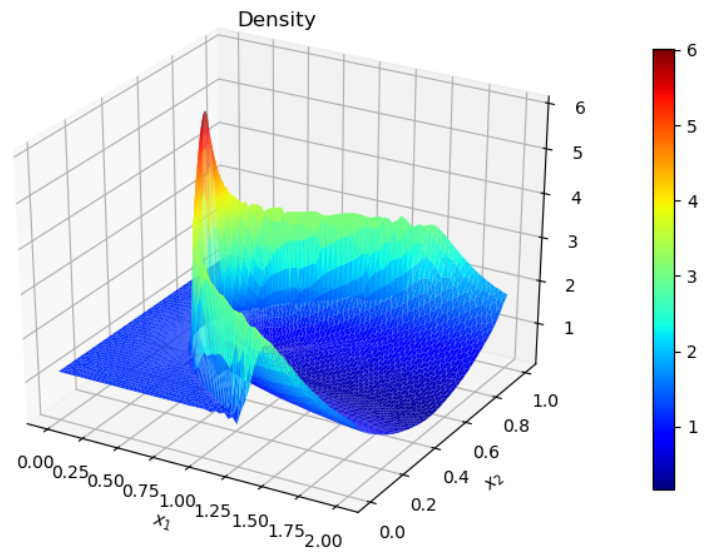
Figure 5.9: Density (kg/m^3), pressure (Pa), and temperature (K) at $M = 8.0$.
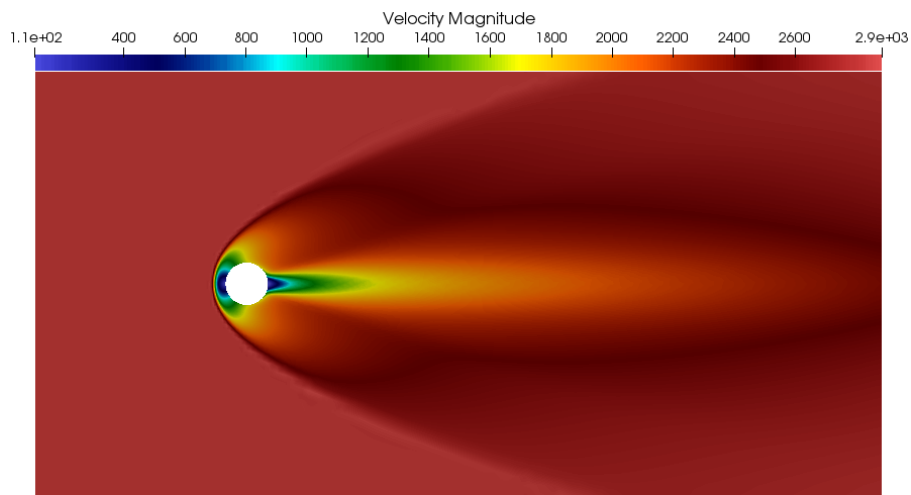
(a)
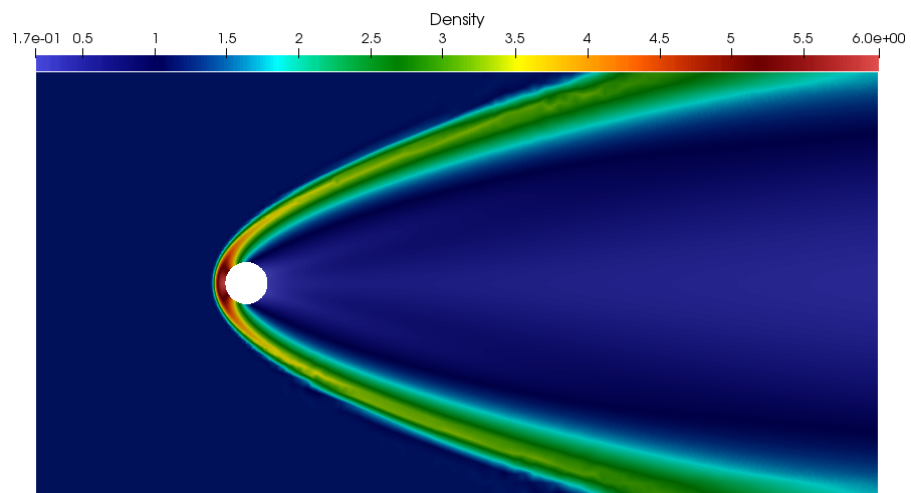
(b)
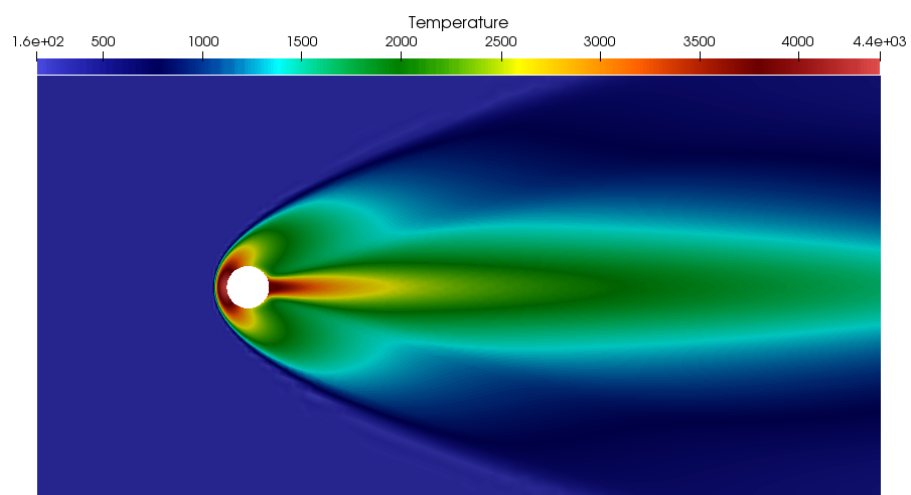
(c)

Figure 5.10: (a) Velocity (m/s), (b) density (kg/m^3), and (c) temperature (K) at $M = 8.0$.
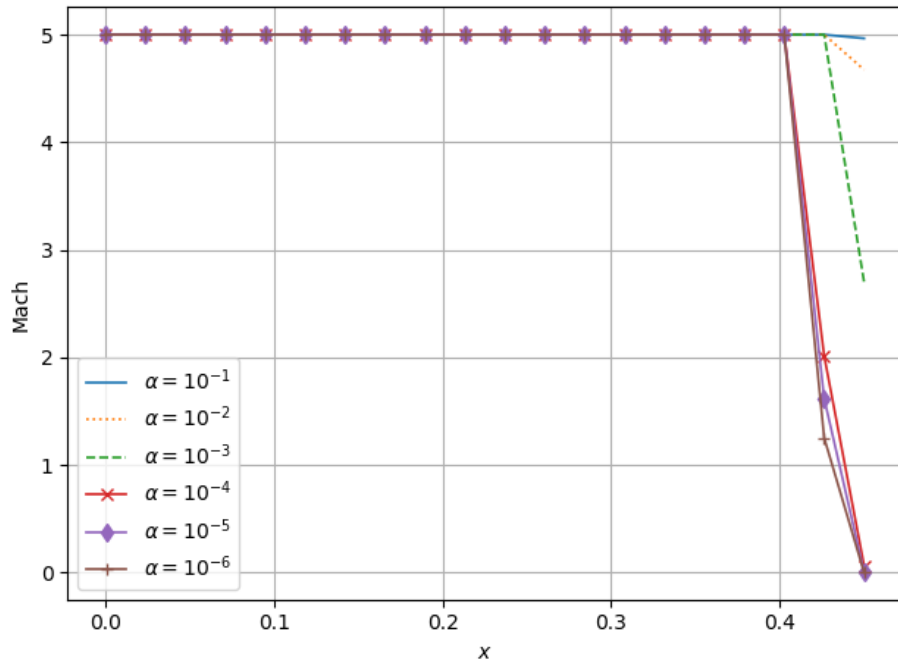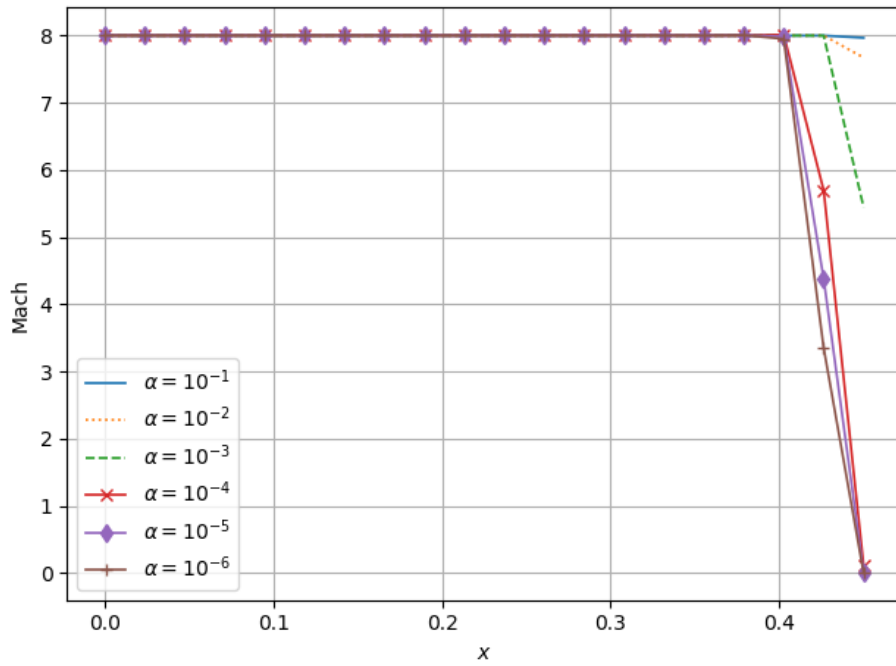
Figure 5.11: Mach elevation and line plots in front of the cylinder (along the stagnation line) at $M = 8.0$ for various values of penetration parameter α.
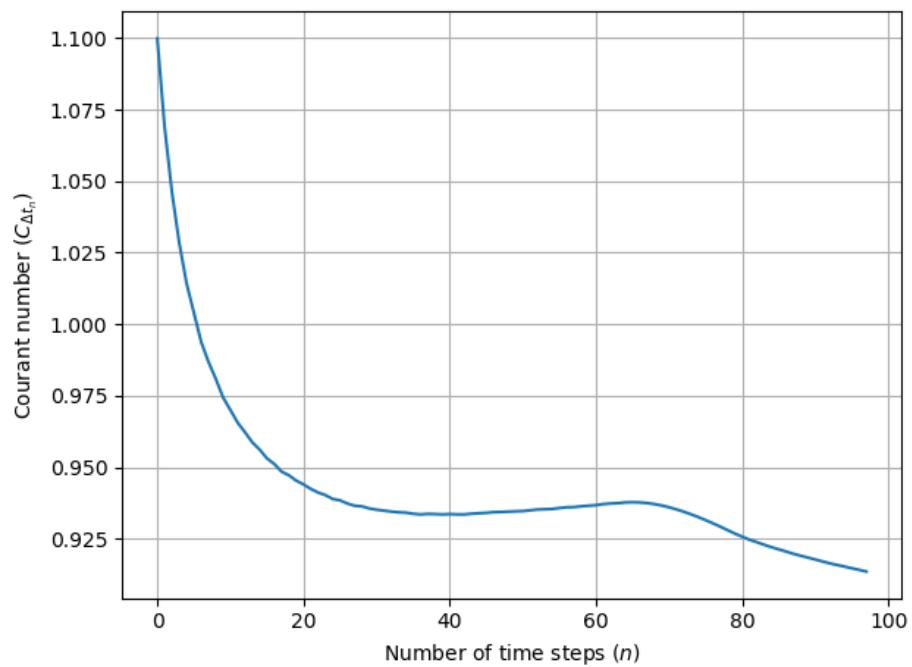
Figure 5.12: Courant number ($C_{\Delta t_n}$) versus number of time steps (n) for $M = 8.0$.

5.1.4 Weak Enforcement of the Zero-Normal-Boundary Condition

Figure 5.13 shows the solutions along the stagnation line, i.e., $x_2 = 0.5$, obtained by imposing the zero-normal-velocity condition with Eq. (4.45) for $M = 2.0$, $M = 5.0$, and $M = 8.0$. Although the weak enforcement yields similar velocity profiles as presented in [159], it is observed that as the Mach number increases, the flow slightly penetrates the cylinder.

On the other hand, it is observed that the penalty approach enforced with Eq. (4.44) apparently works well for various values of the Mach number (see Figures 5.3, 5.7, and 5.11). For values of the penalty parameter larger than $\alpha = 1.0 \times 10^{-4}$, the penalty approach fails to ensure the impermeability of the cylinder surface.

Figure 5.14 demonstrates the performance of the penalty approach for imposing the impermeability condition on the cylinder surface for various Mach numbers and $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg. Compared to the results obtained with the weak enforcement (see Figure 5.13), it is clear that the velocity vectors vanish totally and more sharply at the stagnation point.
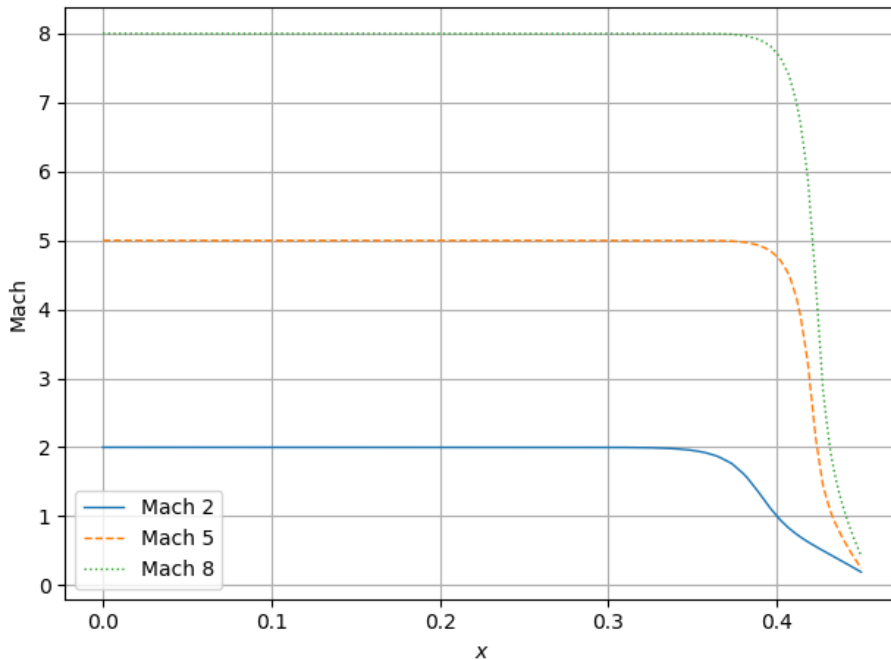
Figure 5.13: Mach elevation and line plots in front of the cylinder (along the stagnation line) at various values of M for weakly imposed zero-normal-velocity condition.
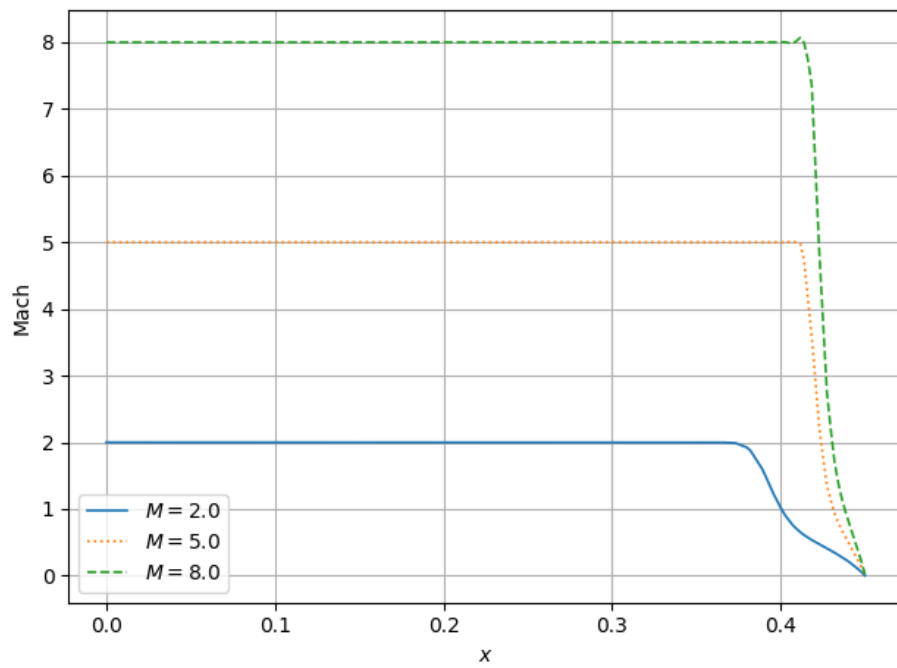
Figure 5.14: Mach elevation and line plots in front of the cylinder (along the stagnation line) at various values of M for penalized zero-normal-velocity condition, $\alpha = 1.0 \times 10^{-5}$ m$^2 \cdot$s/kg.

5.1.5   Stabilization Parameters

Figure 5.15 shows the distributions of the mesh-dependent stabilization parameters (see Section 4.1.1 and Section 4.1.2) for $M = 2.0$, $M = 5.0$, and $M = 8.0$. The parameters are only given for the equilibrium flows because the meshes used in computations are the same, and the flow characteristics (velocity profiles) are very similar to those associated with equilibrium flows.

The sharp gradients formed by the velocity vector vanishing due to the enforcement of the no-penetration boundary condition right front the cylinder surface result in higher values of the stabilization parameter in these regions, which is consistent with the formulation given by Eqs. (4.13)–(4.15).

(a)   (b)

(c)
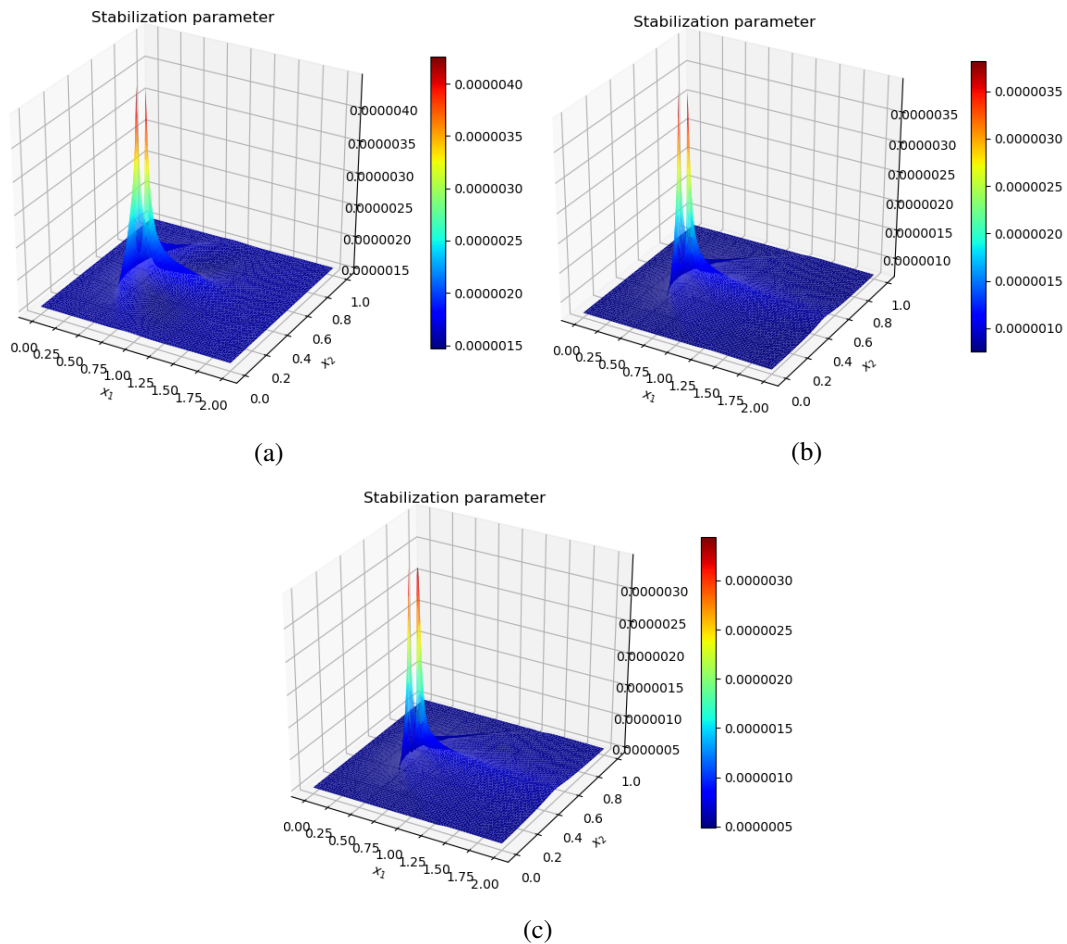
Figure 5.15: Stabilization parameters for equilibrium flows at (a) $M = 2.0$, (b) $M = 5.0$, and (c) $M = 8.0$.

5.1.6 Mesh Size Convergence

The mesh convergence process entails reducing the element size and assessing its effect on the accuracy of the solutions. This type of study is done in order to find a compromise between accuracy and computational cost.

Instead of a pre-adaptive mesh, i.e., shown in Figure 4.3, we employ an unstructured mesh generated in the FEniCS environment for computational simplicity since generating an adaptive mesh for each case would be very challenging. The CFL condition is applied by setting $C_{\Delta t} = 1.0$ because the nonlinear iteration process fails to convergence when such regular (non-adaptive) meshes are used. Besides that, the Mach number is set $M = 5.0$ in computations because, due to the regular meshes used, the nonlinear process fails in computing flows at $M = 8.0$, even with very small Courant numbers.

The mixture densities and pressures are compared along the stagnation line, i.e., $x_2 = 0.5$, for a series of six meshes in Figure 5.16 and Figure 5.17, respectively. It is seen that the solutions (density and pressure profiles along the stagnation line) obtained with the meshes constructed with $n_{el} = 3,061$ or more elements are almost indistinguishable.

Figure 5.18 and Figure 5.19 compare the solutions obtained on a reference mesh constructed with $n_{el} = 5,107$ elements and on the actual mesh (see Figure 4.2) for density and pressure at $M = 2.0$, respectively. It is observed that the solutions obtained with the reference (regular) mesh are prone to exhibit oscillatory behavior near the stagnation point where steep gradients occur. The solutions obtained with the pre-adaptive mesh reveal that the sharp gradients near the cylinder are resolved accurately. The layers of constant-thickness elements near the cylinder are what make the pre-adaptive mesh successful (see Figure 4.3).
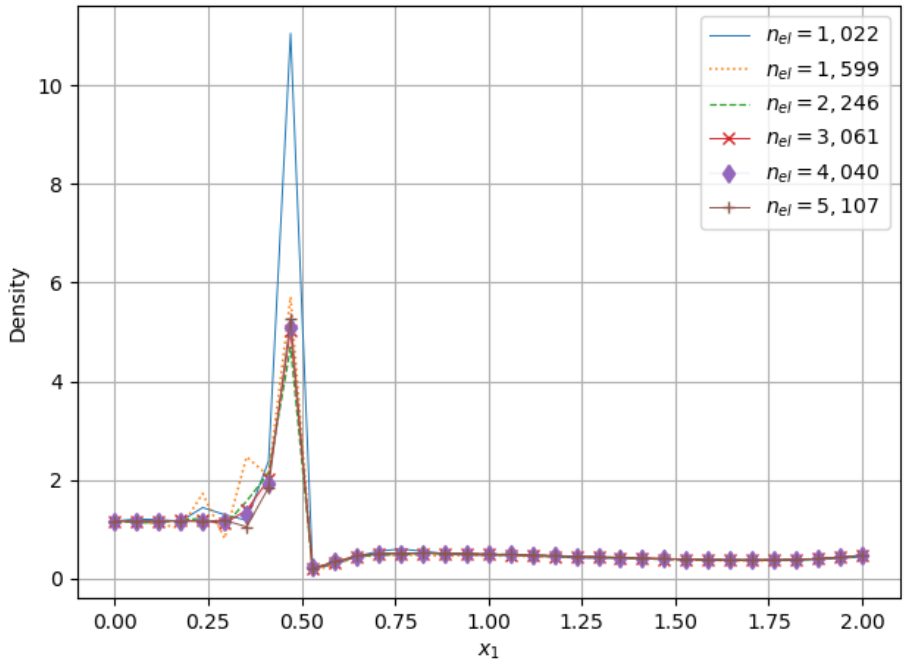
Figure 5.16: Mesh convergence. Densities (kg/m^3) along the stagnation line for equilibrium flow at $M = 5.0$.
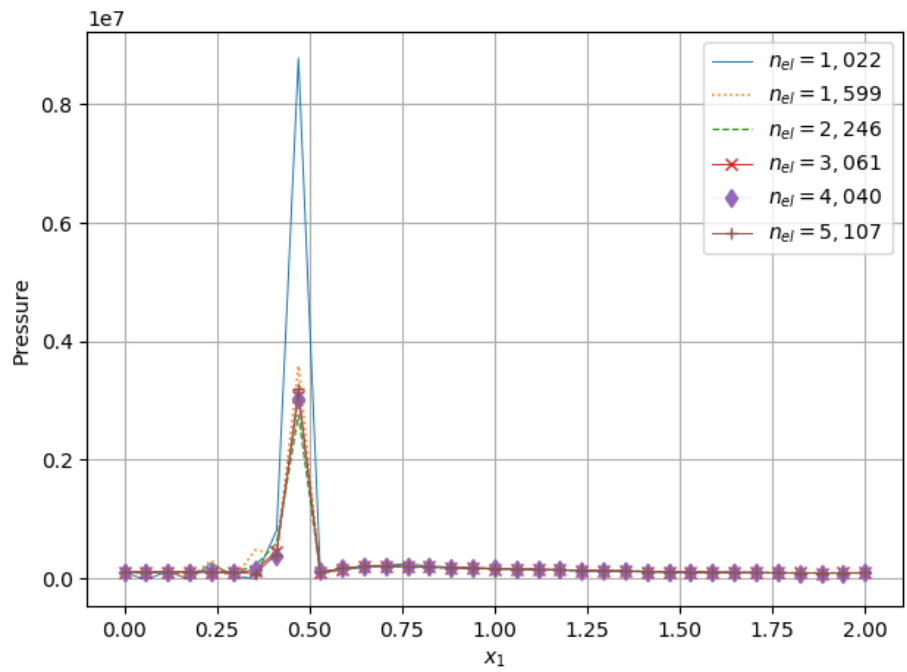
Figure 5.17: Mesh convergence. Pressures (Pa) along the stagnation line for equilibrium flow at $M = 5.0$.
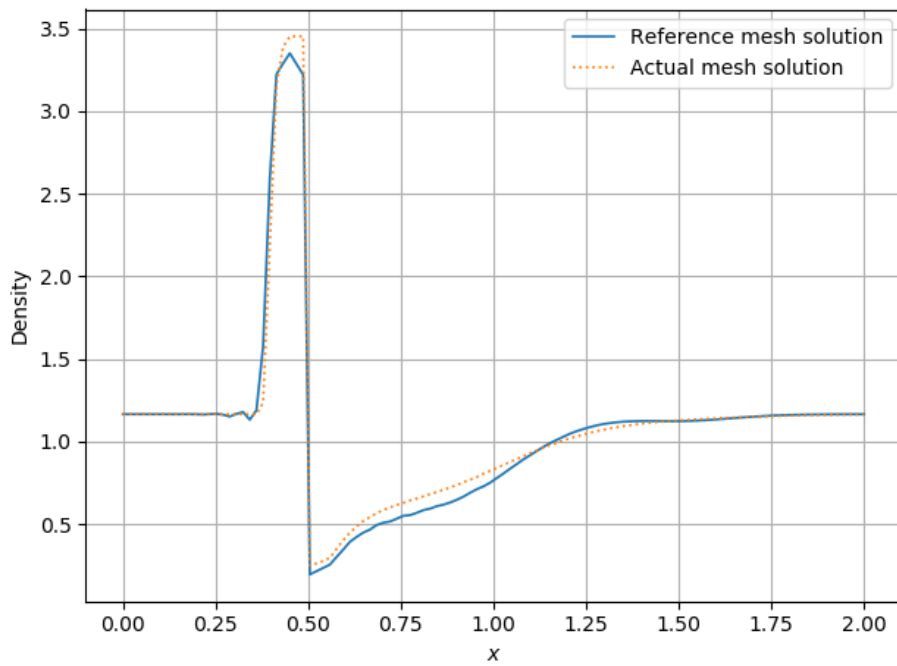
Figure 5.18: Comparison of densities (kg/m^3) obtained with a reference mesh ($n_{el} = 5,107$) and with actual mesh; $M = 2.0$.
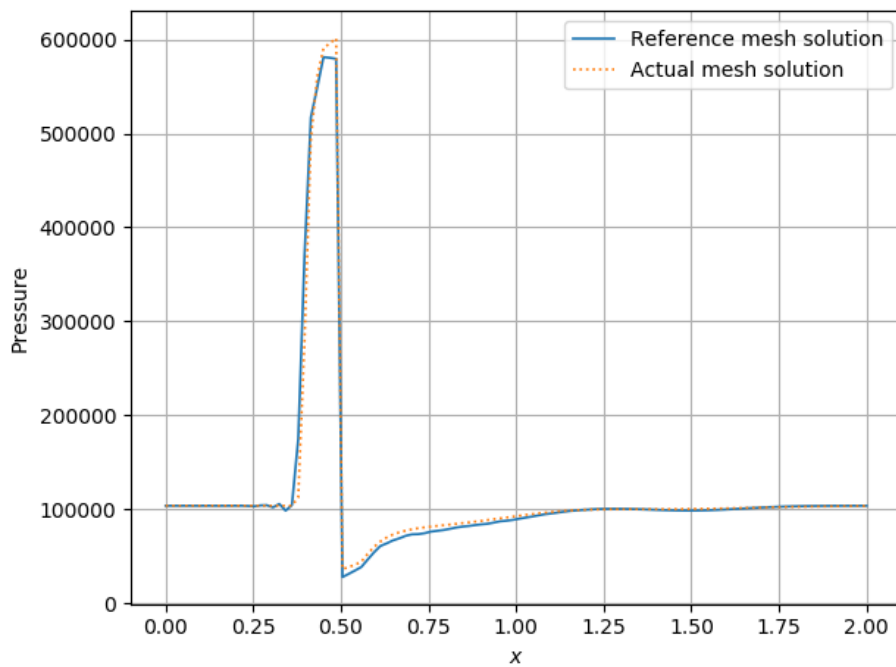
Figure 5.19: Comparison of pressures (Pa) obtained with a reference mesh ($n_{el} = 5,107$) and with actual mesh; $M = 2.0$.

5.2 Multispecies Hypersonic Flow in Nonequilibrium

Recall the governing equations of 2D five-species hypersonic nonequilibrium flows introduced in Section 2.4 by Eqs. (2.115)–(2.118):

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \mathbf{u}) = \omega_s, \tag{5.4}$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = \mathbf{0}, \tag{5.5}$$

$$\frac{\partial \rho e}{\partial t} + \nabla \cdot (\rho \mathbf{u} h) = 0, \tag{5.6}$$

$$\frac{\partial \rho e_V}{\partial t} + \nabla \cdot (\rho \mathbf{u} e_V) = \omega_V, \tag{5.7}$$

where the system is equipped with boundary conditions introduced in Section 4.2. Recall that the compressible-flow SUPG formulation of the problem supplemented with YZβ shock-capturing is presented by Eq. (4.39).

In all computations, the initial densities of species O, N, and NO are set to zero. The free-stream conditions used in computations are set $\rho_{O_2} = 0.29 \times 1.331$ kg/m^3, $\rho_{N_2} = 0.79 \times 1.165$ kg/m^3, and $T_\infty = 300.0$ K. Note that, compared to the equilibrium case, besides the additional initial conditions for the species densities, vibrational energy equation given by Eq. (5.7) also requires an additional initial condition. It is prescribed through the vibrational temperature, T_V, by specifying it as $T_{V,\infty} = T_\infty$, where $T_{V,\infty}$ represents the free-stream value of the vibrational temperature.

The penalty parameter of zero-normal-velocity condition is set as $\alpha = 1.0 \times 10^{-5}$ m^2·s/kg. In the YZβ term, we set $\beta = 2$. Test computations are performed until $t = 1.0 \times 10^{-3}$ s. The system of algebraic equations resulting from the space and time discretizations of Eqs. (5.4)–(5.7) has $54,531$ unknowns. The details on the linear and nonlinear solvers used in computations can be found in Section 4.6. Here, we apply the CFL condition by setting the Courant number as $C_{\Delta t} = 1.0$ (see Section 4.4). This selection has also effect on the Newton-Raphson iterations. It is observed that the number of N–R iterations decreases compared to those in the equilibrium computations.

A modified form of the exponential Arrhenius formula is used to calculate the forward rate coefficients in chemical reactions. The Landau–Teller model is combined with the Millikan–White model for the vibrational relaxation time to calculate the vibrational energy transfers. The computation of thermochemical source terms is explained in

Chapter 2. For the physical and chemical constants and curve-fitting parameters used in nonequilibrium computations, one can refer to Appendix A.

The huge thermochemical source terms, which make the solution process even more challenging, are also involved here, even though the same high velocities are used, as in the equilibrium simulations. However, it is observed that the simulations obtained with the combination of compressible-flow SUPG and YZβ shock-capturing do not contain any local or global spurious oscillations.

The 2D velocity and mixture density plots for nonequilibrium flows are not shown because the flow characteristics in equilibrium and nonequilibrium simulations show very similar trends (see Figures 5.2, 5.6, and 5.10). The simulations show that the mass fraction of oxygen atoms (O) is higher than that of nitrogen atoms (N), as expected; since the oxygen molecules (O_2) begin to dissociate at lower temperatures than nitrogen molecules (N_2) do (see Section 1.2.2). Notice that the mass fraction of nitrogen atoms is lower than that of nitric oxide (NO) molecules since the nitrogen atoms dissociated combine with oxygen atoms immediately and form NO molecules. The results are consistent with those of other published reports in the literature, see, for example [103, 107, 136, 208].

Although computed with different free-stream conditions and computational setups, the results for species mass fraction obtained here show similar solution profiles to those performed in [131]. The findings are also in line with those reported in [106, 107]. Again, the thermal equilibrium process follows a similar trend as in [32].

5.2.1   Mach 2.0

We set the free-stream velocity vector as $\mathbf{u}_\infty = (706.13, 0)$ m/s. The species densities are presented for the hypersonic thermochemical nonequilibrium flow at $M = 2.0$ in Figure 5.20. The species mass fractions are presented in Figure 5.21. Figure 5.22 shows the elevation plots for the pressure, translational-rotational temperature, and vibrational temperature around the cylinder.

The cylinder, as expected, generates bow shocks by causing moderate jumps in species densities, pressure, and temperatures. The high pressure and temperature distributions

predominantly accumulated near the stagnation point in a small area (see Figure 5.20 and Figure 5.22). It is seen that the densities of O_2 and N_2 increase up to 1.0 kg/m^3 and 2.0 kg/m^3, respectively, right in front of the cylinder. Right behind the cylinder, these values decrease to around 0.1 kg/m^3 and 0.25 kg/m^3, respectively. Since enough high temperatures are not reached at speeds of $M = 2.0$ for the production of species O, N, and NO, their densities remain at low levels.

The first molecule to decompose in the flow field as a result of high temperatures is O_2, as expected. Although only to a small extent, as can be seen in density plots, O, N, and NO productions begin in the flow field. Figure 5.21 indicates that the dissociation of N_2 molecules can be neglected at such speeds. Therefore, the calorically gas assumption can be made for flows at $M = 2.0$.

At the stagnation point, the translational-rotational temperature and vibrational temperature are about $T = 588$ K and $T_V = 584$ K, respectively. The controlling temperature is around $\overline{\overline{T}} = 587$ K. These results indicates that a state of thermal equilibrium has been reached.

Figure 5.23 indicates that thermal equilibrium cannot be achieved when the translational-vibrational energy exchange term, $Q_{\text{transfer}}^{t-v}$, is omitted from the computations (for comparison, see Figure 5.22).

The number of time steps to reach the final time is 270. The average number of Newton–Raphson iterations at each time-step is 3.

Figure 5.20: Densities (kg/m^3) of species (a) O, (b) N, (c) NO, (d) O_2, and (e) N_2 at $M = 2.0$.

Figure 5.21: Mass fractions of species (a) O, (b) N, (c) NO, (d) O_2, and (e) N_2 at $M = 2.0$.

(a)

(b)
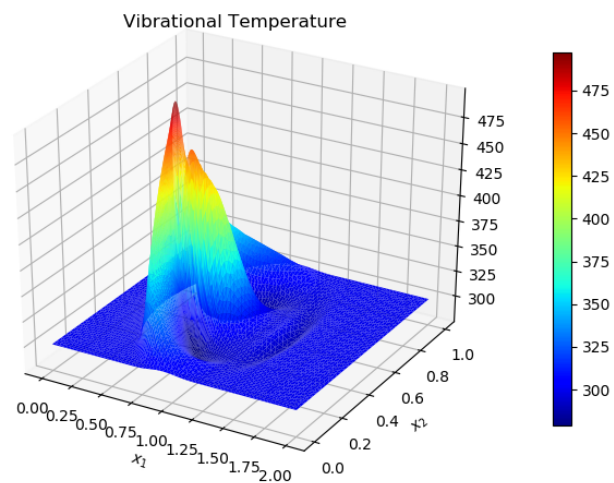
(c)

Figure 5.22: (a) Pressure (Pa), (b) translational-rotational temperature (K), and (c) vibrational temperature (K) at $M = 2.0$.

Translational-Rotational Temperature

(a)

Vibrational Temperature
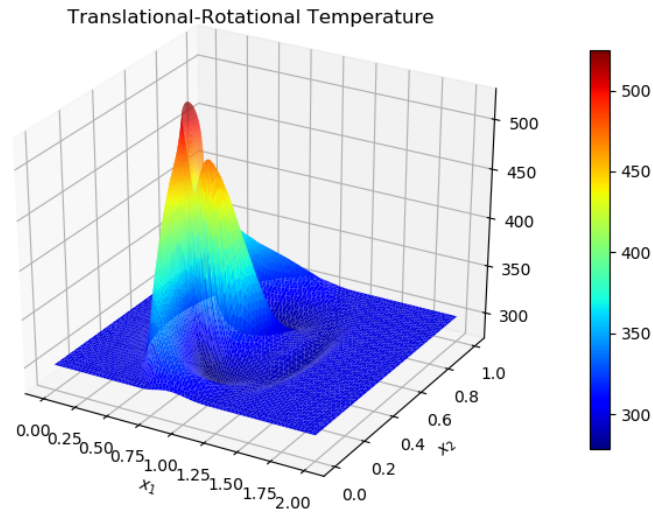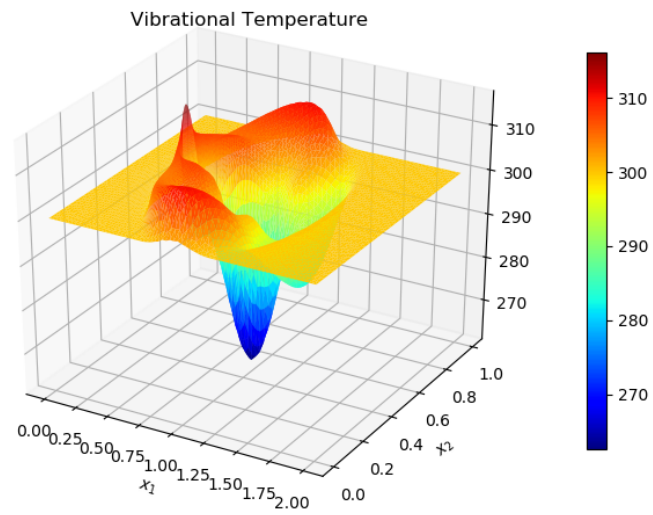
(b)

Figure 5.23: (a) Translational-rotational temperature (K), (b) vibrational temperature (K) computed ignoring the translational-vibrational energy exchange term $Q_{\text{transfer}}^{t-v}$ at $M = 2.0$.

5.2.2 Mach 5.0

The inflow velocity vector is taken as $\mathbf{u}_\infty = (1765.33, 0)$ m/s. The species densities are presented for the hypersonic thermochemical nonequilibrium flow at $M = 5.0$ in Figure 5.24. The species mass fractions are presented in Figure 5.25. Figure 5.26 shows the elevation plots for the pressure, translational-rotational temperature, and vibrational temperature around the cylinder.

The cylinder generates bow shocks by causing rapid changes in species densities, pressure, and temperatures, which are stronger than those observed in the supersonic ($M = 2.0$) case, as expected. The high pressure and temperature distributions predominantly accumulated near the stagnation point in a small region (see Figure 5.24 and Figure 5.26).

It is observed that the densities of O_2 and N_2 increase up to 2.0 kg/m^3 and 5.0 kg/m^3, respectively, around the stagnation point. Right behind the cylinder, these values decrease to around 0.2 kg/m^3 and 0.5 kg/m^3, respectively. Due to the increasing speed and resulting higher temperatures, the species densities have reached significant levels when compared to the $M = 2.0$ case. The species densities are around 0.3 kg/m^3 for O, 0.005 kg/m^3 for N, and 0.01 kg/m^3 for NO at the stagnation point.

As can be seen in density and mass fraction plots, nitrogen (N_2) and oxygen (O_2) molecules begin to dissociate, and O, N, and NO productions begin in the flow field. Figure 5.25 shows that N_2 dissociation reaches critical levels that must be taken into account. Therefore, the calorically gas assumption cannot be made for flows at $M = 5.0$.

The translational-rotational temperature and vibrational temperature are about $T = 782$ K and $T_V = 782$ K at the stagnation point, respectively. Therefore, the controlling temperature is also around $\overline{T} = 782$ K. These results indicates that a state of thermal equilibrium has been reached.

The number of time steps to reach the final time is 539. At each time-step, the average number of Newton–Raphson iterations is 3.
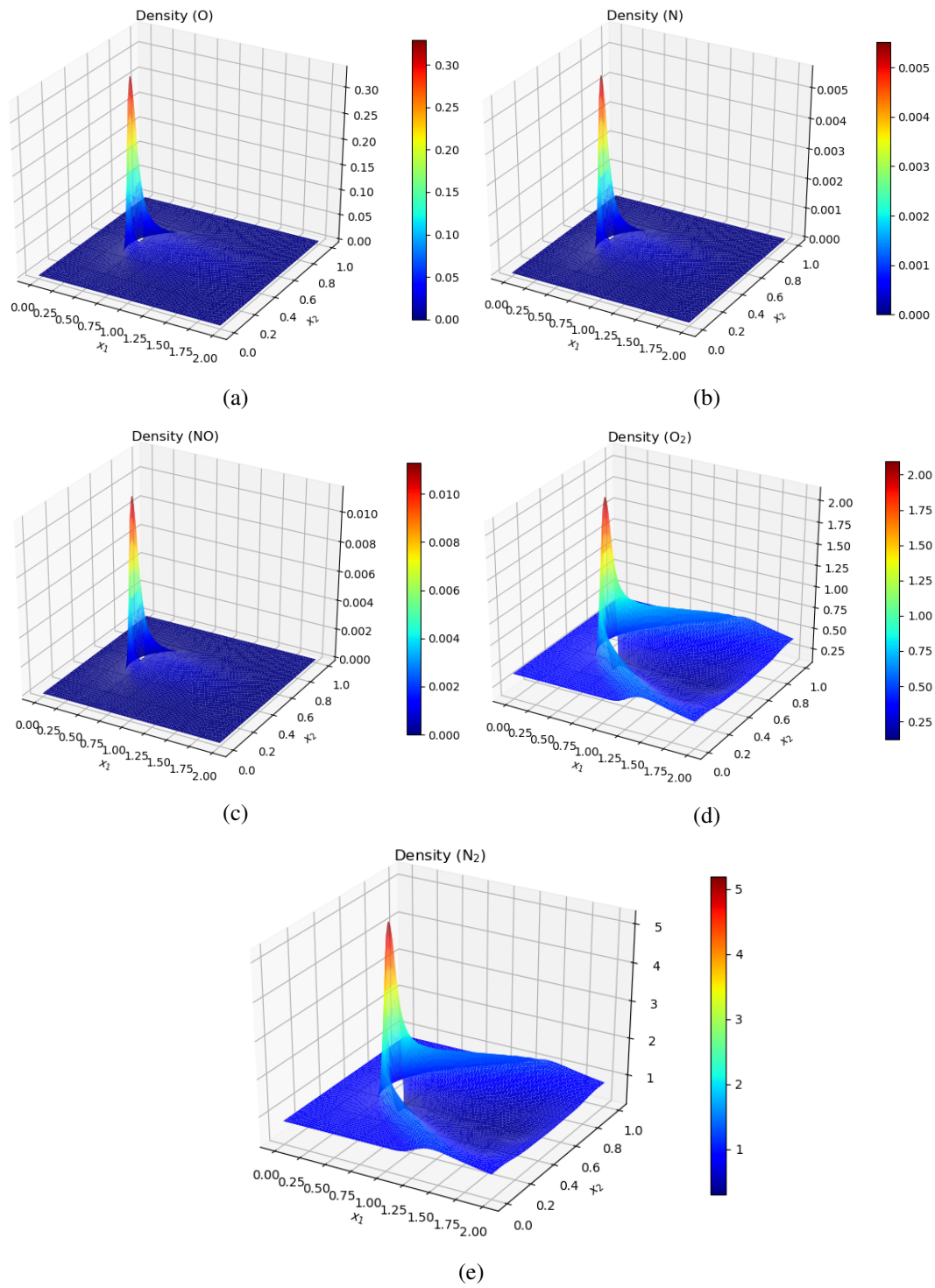
Figure 5.24: Densities (kg/m^3) of species (a) O, (b) N, (c) NO, (d) O_2, and (e) N_2 at $M = 5.0$.
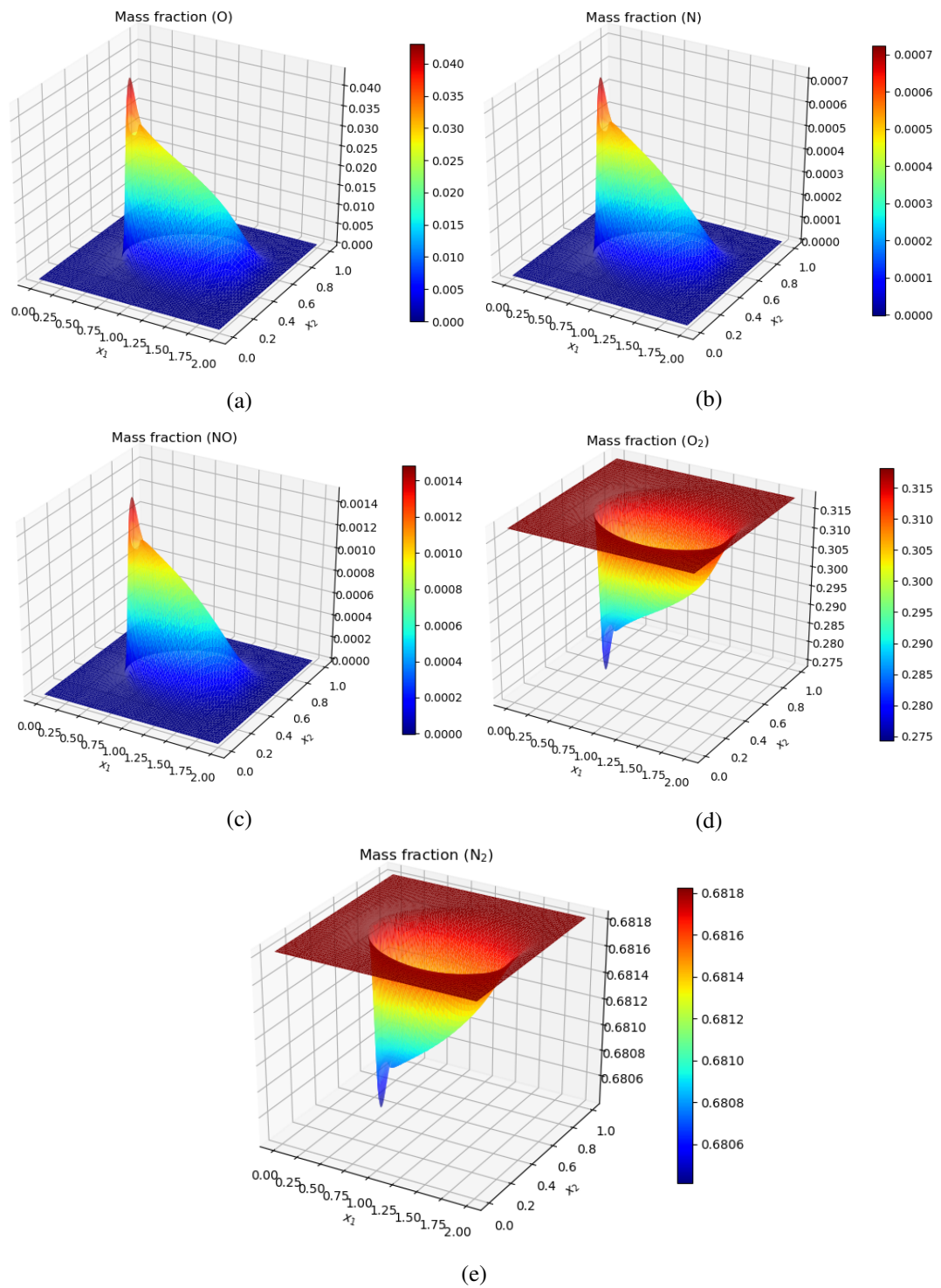
Figure 5.25: Mass fractions of species (a) O, (b) N, (c) NO, (d) O_2, and (e) N_2 at $M = 5.0$.
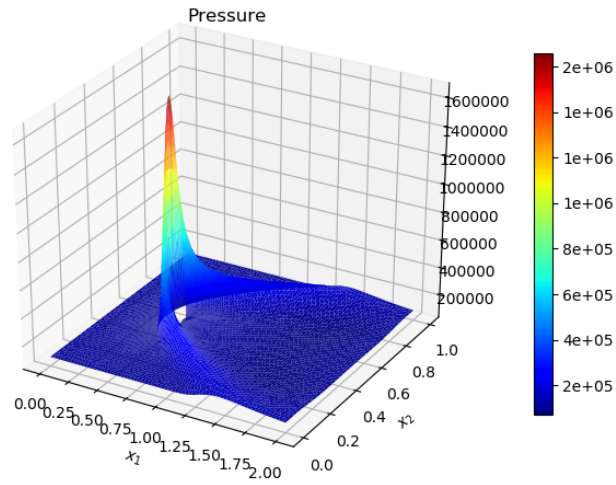
(a)
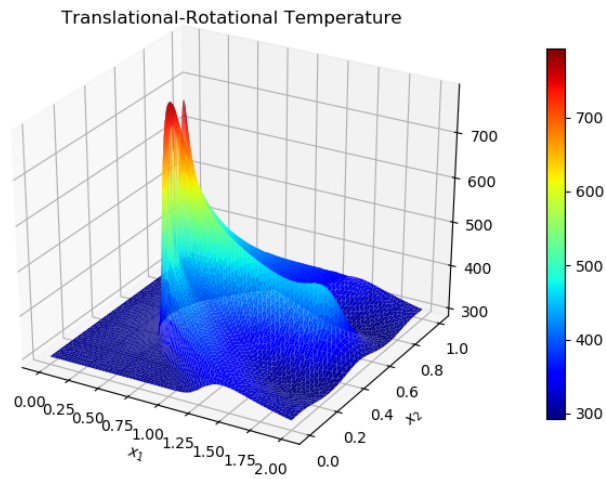
(b)
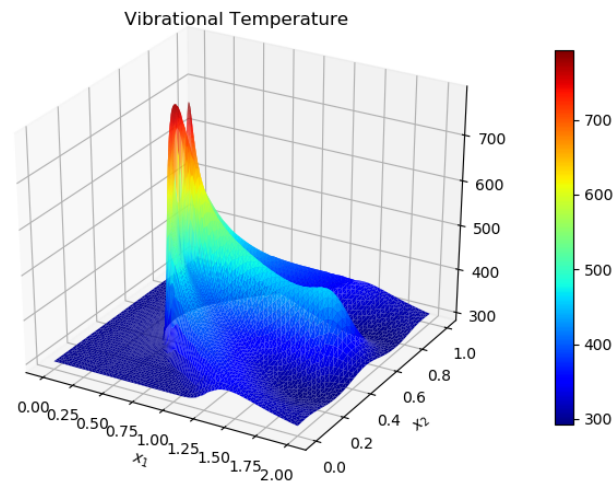
(c)

Figure 5.26: (a) Pressure (Pa), (b) translational-rotational temperature (K), and (c) vibrational temperature (K) at $M = 5.0$.

5.2.3 Mach 8.0

In computations, we set the free-stream velocity vector as $\mathbf{u}_\infty = (2824.53, 0)$ m/s. The species densities are presented for the hypersonic thermochemical nonequilibrium flow at $M = 8.0$ in Figure 5.27. The species mass fractions are presented in Figure 5.28. Figure 5.29 shows the elevation plots for the pressure, translational-rotational temperature, and vibrational temperature around the cylinder.

Compared to the previous nonequilibrium cases, i.e., $M = 2.0$ and $M = 5.0$, the cylinder generates stronger bow shocks with large jumps in species densities, pressure, and temperatures. The high pressure and temperature distributions predominantly accumulated near the stagnation point in a small area (see Figure 5.27 and Figure 5.29).

It is observed that the densities of O_2 and N_2 increase up to 1.8 kg/m^3 and 7.0 kg/m^3, respectively, around the stagnation point. Right behind the cylinder, these values decrease to around 0.2 kg/m^3 and 0.5 kg/m^3, respectively. Due to the even increasing speed and resulting even higher temperatures, the species densities have reached significant levels when compared to the $M = 2.0$ and $M = 5.0$ cases. The species densities are around 1.4 kg/m^3 for O, 0.05 kg/m^3 for N, and 0.08 kg/m^3 for NO at the stagnation point.

As can be seen in density and mass fraction plots, O, N, and NO productions are significant for such speeds. Figure 5.28 shows that, at higher temperatures, the dissociation of N_2, which was negligible at $M = 2.0$ but significant at $M = 5.0$, became even more important in the flow field at $M = 8.0$.

At the stagnation point, the translational-rotational temperature and vibrational temperature are about $T = 1,107$ K and $T_V = 1,107$ K, respectively. Then, naturally, the controlling temperature is also around $\overline{T} = 1,107$ K. These results indicates that a state of thermal equilibrium has been reached.

The number of time steps to reach the final time is 808. The average number of Newton–Raphson iterations at each time-step is 3.
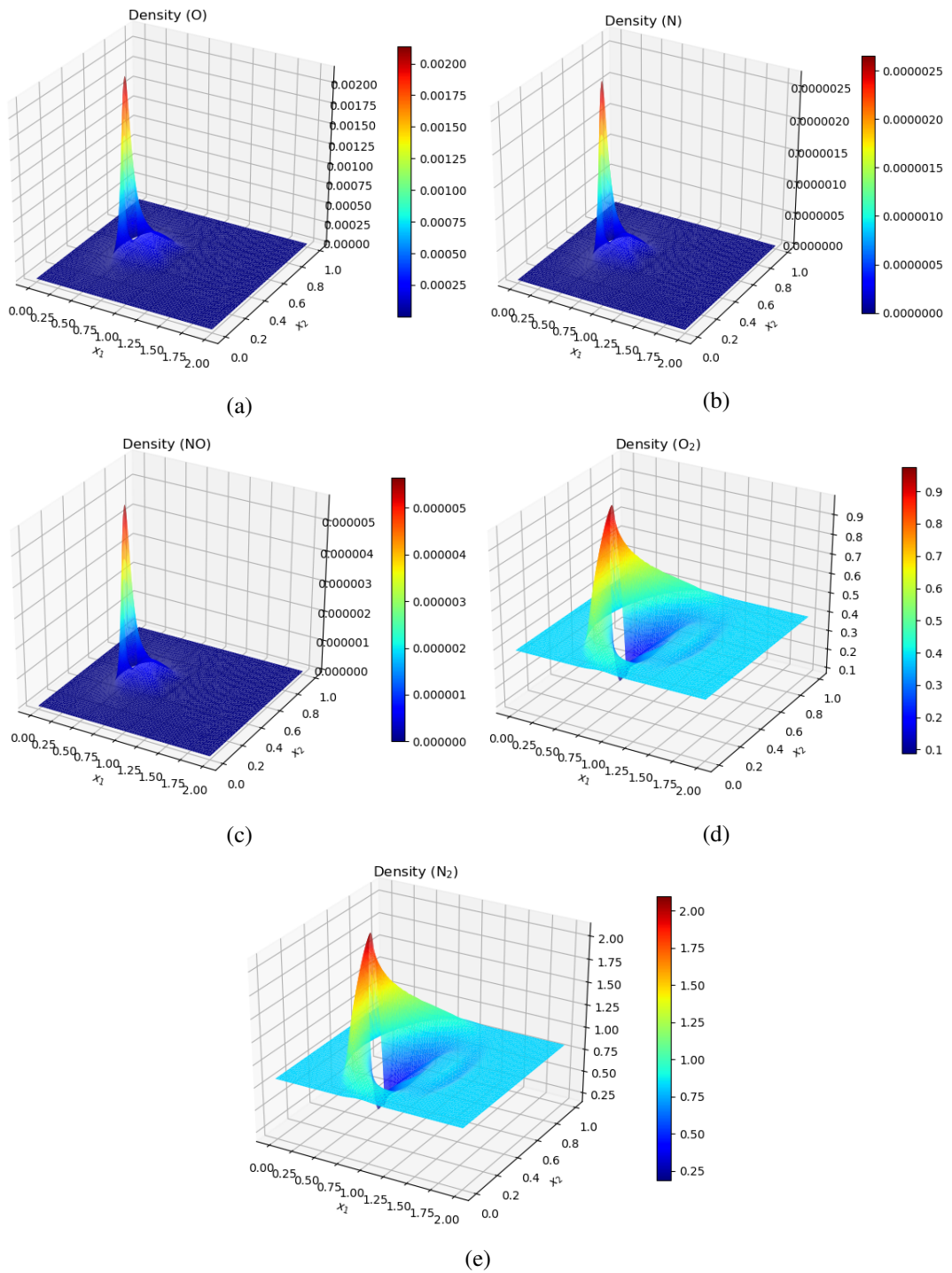
(a)

(b)

(c)

(d)

(e)
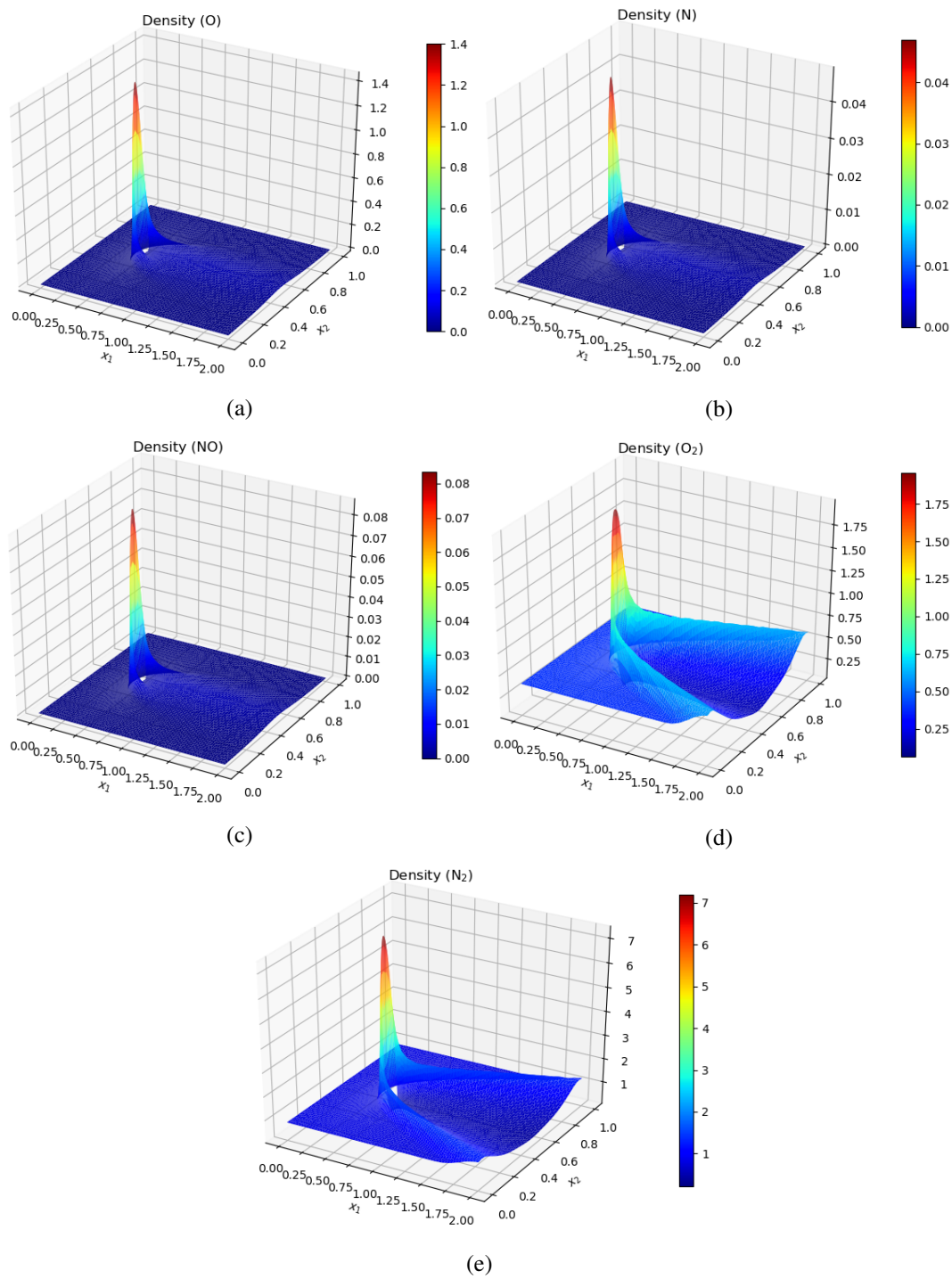
Figure 5.27: Densities (kg/m^3) of species (a) O, (b) N, (c) NO, (d) O_2, and (e) N_2 at $M = 8.0$.
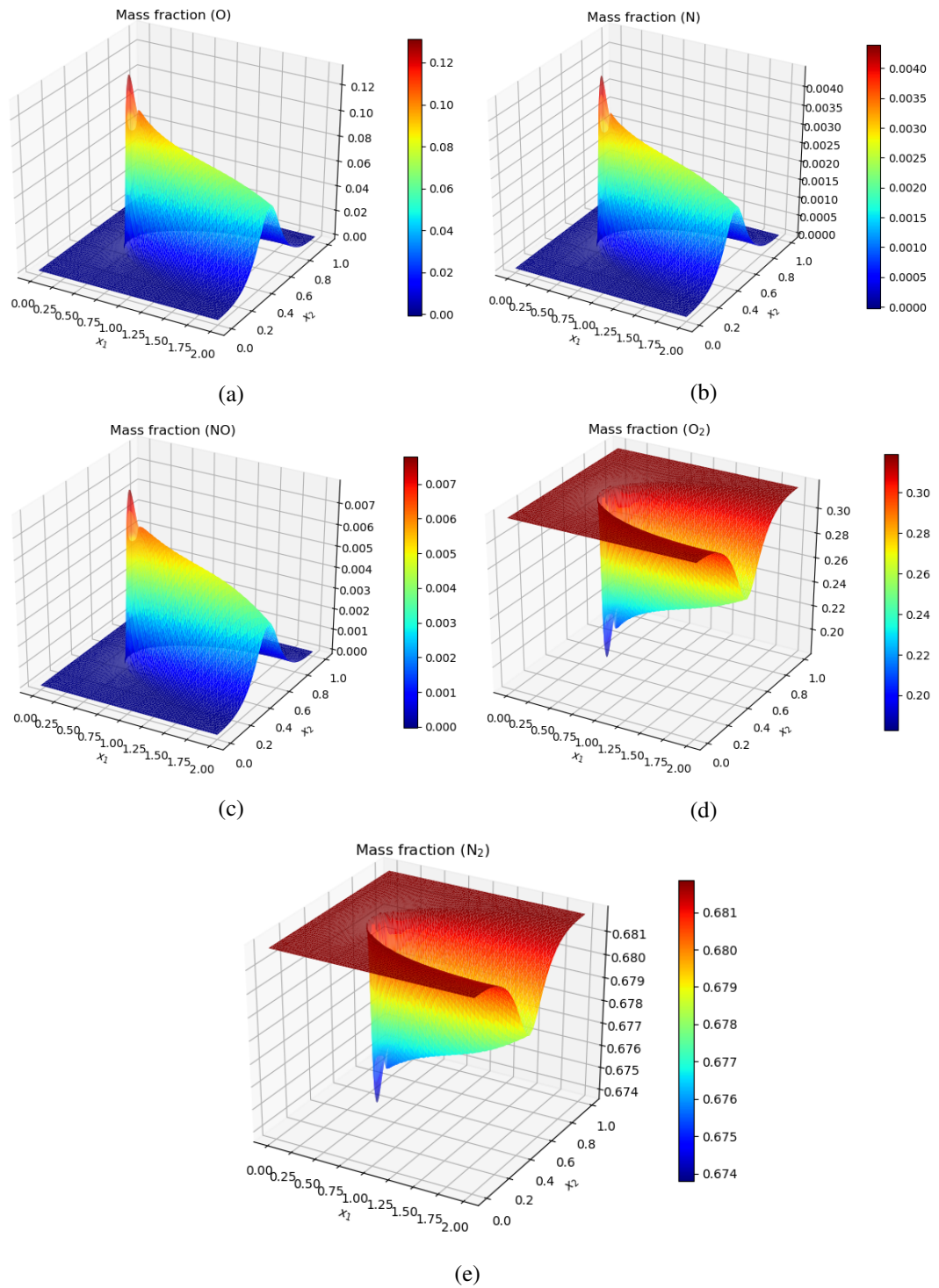
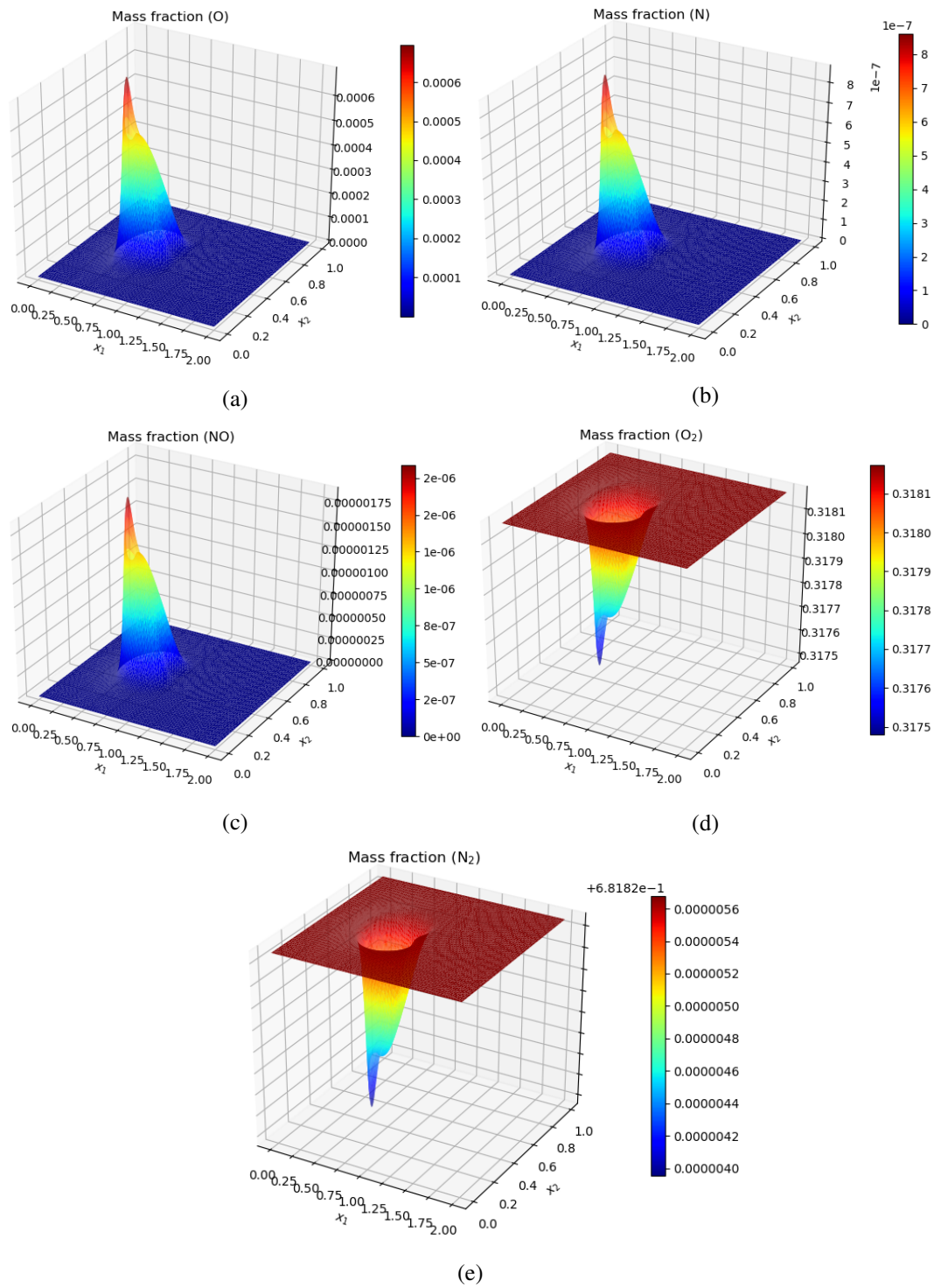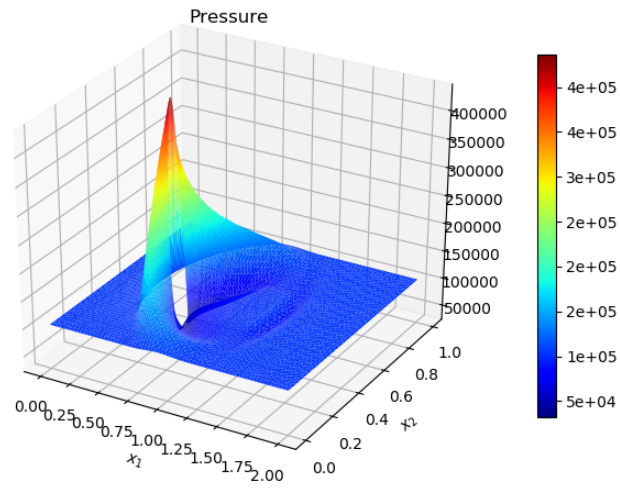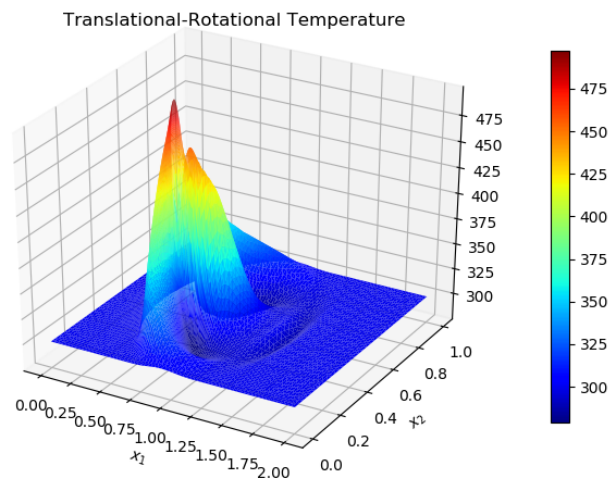Figure 5.28: Mass fractions of species (a) O, (b) N, (c) NO, (d) O_2, and (e) N_2 at $M = 8.0$.
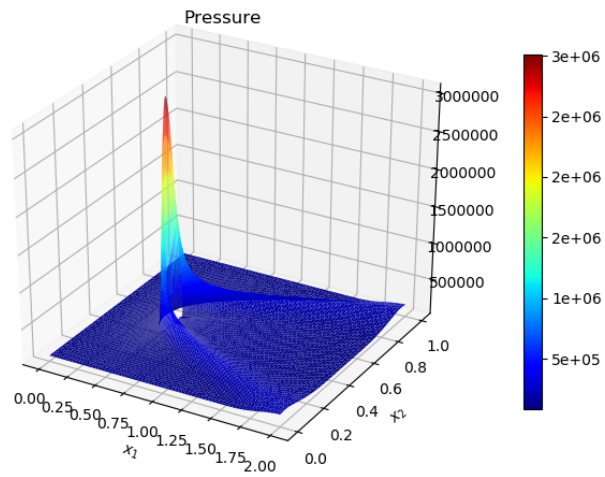
(a)
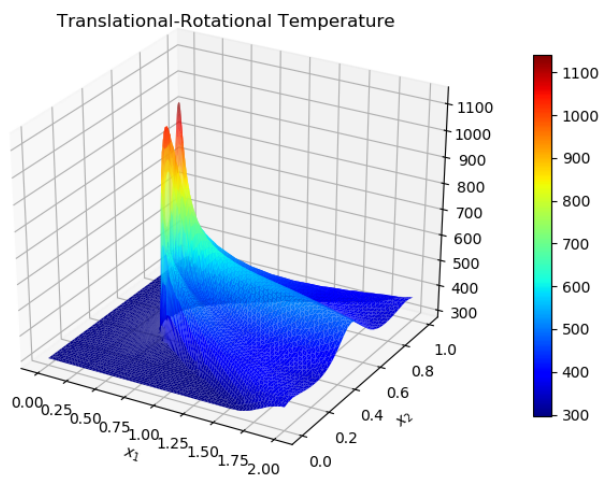
(b)
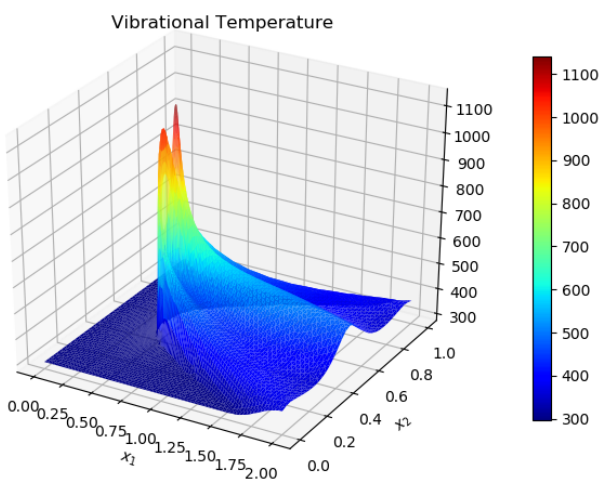

(c)

Figure 5.29: (a) Pressure (Pa), (b) translational-rotational temperature (K), and (c) vibrational temperature (K) at $M = 8.0$.

CHAPTER 6

CONCLUSIONS AND DISCUSSIONS

In this thesis, high-speed flows past a cylinder have been investigated. The simulations have been carried out for various Mach numbers from supersonic ($M = 2.0$) to hypersonic ($M = 5.0$ and $M = 8.0$). Both types of flows of thermocehmical equilibrium and nonequilibrium have been considered.

Prior to the high-speed flow computations, relatively simpler problems, i.e., 2D convection-dominated reaction-convection-diffusion equations, have been considered first to establish a basis for the stabilization and shock-capturing concepts. Towards this end, the Streamline-Upwind/Petrov–Galerkin (SUPG) formulation and YZβ shock-capturing technique have been proposed for solving a variety of convection-dominated problems, including the Burgers'-type equations. The numerical results have demonstrated that both local and global spurious oscillations have been successfully eliminated.

The compressible-flow SUPG method has been employed to overcome the instability issues encountered in simulating high-speed flows by using the standard Galerkin finite element method. The YZβ shock-capturing has also assisted the SUPG-stabilized formulation to obtain better shock representations around strong gradients. The CFL condition has been used to ensure stability in time with an implicit Euler temporal discretization.

The flow of nitrogen (N_2) gas has been considered for the equilibrium computations. For the nonequilibrium case, a five-species (O, N, NO, O_2, N_2) air mixture with a 17-reaction kinetics model has been used. Chemical source terms have been computed

141

using Park's two-temperature (controlling-temperature) approach.

The enforcement of the zero-normal-velocity (impermeable, slip) boundary condition on the cylinder surface has been accomplished by adding a penalizing term to the momentum equation in variational formulations. Besides that, for applying the slip condition on the cylinder surface, a weak enforcement approach has also been proposed, and the results have been compared to those obtained using the penalty technique. The system of nonlinear equations resulting from space and time discretizations has been solved with the Newton–Raphson method and the ILU-preconditioned GMRES search technique.

The followings provide a summary of the main findings and outcomes of the computations performed in this thesis.

- On a large set of 2D problems, the need for stabilized formulations and the necessity of the stabilized formulations should also be augmented with shock-capturing operators for solving convection-dominated problems has been demonstrated. Towards this end, the SUPG formulation supplemented with YZβ shock-capturing, SUPG-YZβ, has been applied to five test problems, including the Burgers'-type equations at high Reynolds numbers.

  Although the SUPG formulation has been quite successful in preventing global oscillations, the results have demonstrated that the method was insufficient in resolving local oscillations where steep gradients occurred. When the SUPG-stabilized formulation has been also supplemented with the YZβ shock-capturing, it has been observed that spurious oscillations have been eliminated almost completely.

- Supersonic and hypersonic flows past a cylinder in thermochemical equilibrium/nonequilibrium have been successfully simulated in the FEniCS environment. The simulations obtained do not contain any local or global spurious oscillations, although they have very sharp gradients near the cylinder where shocks occur. This situation indicates that the SUPG-YZβ formulation works well for simulating high-speed flows in thermochemical equilibrium/nonequilibrium.

- Even if the GFEM formulations have been stabilized with the SUPG method, the

iterative nonlinear solution process (N–R method) could not achieve convergence for solving the governing equations in the case of $M = 5.0$ and $M = 8.0$. This situation has demonstrated that the use of a shock-capturing technique has become compulsory for hypersonic flow conditions.

- High-speed equilibrium flow simulations have shown that for the speeds above $M = 2.0$, the calorically perfect gas assumption is not appropriate and chemical reactions that may take place in the flow field should also be taken into account.

- It has been observed that the weak implementation of the zero-normal-velocity condition with the penalty approach (see Eq. (4.44)) on the cylinder surface works well, and the cylinder surface stands fully impermeable for $\alpha = 1.0 \times 10^{-5}$ m$^2 \cdot$s/kg. It has been observed that the penalty approach failed to ensure impermeability of the cylinder surface for values of α larger than $\alpha = 1.0 \times 10^{-4}$ m$^2 \cdot$s/kg. For the values of the parameter $\alpha = 1.0 \times 10^{-5}$ m$^2 \cdot$s/kg and less, the cylinder surface acts completely impermeable.

  We have also enforced the slip boundary condition weakly (see Eq. (4.45)) and observed that the penalty approach worked better since the weakly enforced formulation allowed the fluid (gas) to penetrate the cylinder, even if to a small extent.

- The nonequilibrium simulations have revealed that oxygen O$_2$ is the primary species dissociating in the five-species air mixture. The reason for that is the energy required for the dissociating of N$_2$ is higher than that for O$_2$. It has also been observed that a significant proportion of the N atoms released due to the dissociation of N$_2$ molecules combined with free O atoms to form NO molecules.

- During the computational experiments, it has been seen that the main contribution to the vibrational energy source term, ω_V, is from the translational-vibrational energy exchange term, $Q_{\text{transfer}}^{t-v}$, rather than the source term associated with the vibrational energy component, Q_v^{chem}, emerging from chemical reactions. The thermal equilibrium could not have been reached if the term $Q_{\text{transfer}}^{t-v}$ had been omitted from the computations.

- Huge chemical and vibrational source terms, i.e., ω_s's and ω_V's, emerging in

143

thermochemical nonequilibrium flows have required the Courant number to be set much smaller, e.g., $C_{\Delta t} = 1.0$, and the computational time and memory use increased considerably. Note that it has been set $C_{\Delta t} = 10.0$ in equilibrium computations.

- The process of obtaining the vibrational temperature with a nonlinear solver during the computations is one of the many challenging steps in performing nonequilibrium simulations, such as the handling of shocks and discontinuities and treatment of huge thermochemical source terms. This challenge has been easily handled in the FEniCS with an appropriate implementation. Consequently, it has been noticed that even if they have highly complex mathematical formulations, when the compressible-flow SUPG formulation is supplemented with the YZβ shock-capturing, hypersonic flows in thermochemical equilibrium/nonequilibrium can be effectively simulated, and the FEniCS is a suitable environment for performing such simulations. However, it should be noted that direct implementation of the zero-normal-velocity boundary condition on nonrectangular walls is quite challenging in FEniCS.

- Even if the problem addressed is linear, the SUPG-YZβ formulation results in nonlinear algebraic equation systems. If the problem under consideration is already nonlinear, the nonlinearity of the system is further enhanced. Therefore, naturally, it has been observed that the number of nonlinear iterations increased slightly in the computations performed with the SUPG-YZβ formulation.

Some possible extensions of the numerical methods, computational techniques, and thermochemical models used in this study may be listed as follows:

- The governing equations used in simulating high-speed equilibrium/nonequilibrium laminar flows can be coupled with various turbulence models, e.g., the Spalart–Allmaras model.

- The simulations performed for inviscid flows, obtained by neglecting the diffusive and heat conduction terms, can be extended to multispecies viscous flows, i.e., Navier–Stokes equations (see Eq. (2.98)).

- Instead of a five-species air mixture model, a seven-species or an eleven-species model can be used. Besides, different temperature models (e.g., a three-temperature model) can also be used and compared.

- The high-speed flow simulations carried out in this study, assuming that they occur in Earth's atmosphere, can be performed for other atmospheres with different chemical structures, e.g., the atmosphere of the red planet Mars or Saturn's satellite, Titan.

- The development of novel stabilized formulations (particularly techniques for calculating element length scales) for the finite element methods is still an active area of research. The compressible-flow solvers developed in this present study can be modified for novel stabilized formulations and equipped with different shock-capturing terms.

- Instead of the semi-discrete SUPG formulation that has been solved with the implicit Euler time-integration algorithm, the space-time SUPG formulations can be preferred.

- The isogeometric analysis (IGA) formulations introduced by Hughes et al. in [85] (see also [14, 15, 17–20, 45, 99, 177]) with the goal of bridging the gap between computer-aided design (CAD) and finite element analysis (FEA) can be also used in conjunction with the compressible-flow SUPG formulation and various shock-capturing techniques.

- For simulating flows faster than Mach 8, ionization processes are quite common, and the effects of ionization should be also taken into account in computations. Future research may incorporate computational models for such high speeds as well.

REFERENCES

[1] B. E. Abali, *Computational Reality: Solving Nonlinear and Coupled Problems in Continuum Mechanics*, Advanced Structured Materials, Springer, Singapore, 2016.

[2] R. A. Adams and J. J. F. Fournier, *Sobolev Spaces*, volume 140 of *Pure and Applied Mathematics*, Academic Press, Amsterdam, Boston, second edition, 2003.

[3] M. S. Alnæs, A. Logg, and K.-A. Mardal, UFC: A finite element code generation interface, in *Automated Solution of Differential Equations by the Finite Element Method*, chapter 16, pp. 283–302, Springer-Verlag Berlin Heidelberg, 2012.

[4] M. S. Alnæs, A. Logg, K. B. Ølgaard, M. E. Rognes, and G. N. Wells, Unified Form Language: A domain-specific language for weak formulations of partial differential equations, ACM Transactions on Mathematical Software (TOMS), 40(2), pp. 1–37, 2014.

[5] M. S. Alnæs and K.-A. Mardal, SyFi and SFC: Symbolic finite elements and form compilation, in *Automated Solution of Differential Equations by the Finite Element Method*, chapter 15, pp. 273–282, Springer-Verlag Berlin Heidelberg, 2012.

[6] J. D. Anderson, *Fundamentals of Aerodynamics*, McGraw-Hill Education, New York, sixth edition, 2017.

[7] J. D. Anderson, *Hypersonic and High-Temperature Gas Dynamics*, AIAA, Washington, DC, third edition, 2019.

[8] W. K. Anderson, J. C. Newman, and S. L. Karman, Stabilized finite elements in FUN3D, Journal of Aircraft, 55(2), pp. 696–714, 2018.

[9] D. N. Arnold and A. Logg, Periodic table of the finite elements, SIAM News, 47(9), p. 212, 2014.

[10] G. Arora and V. Joshi, A computational approach using modified trigonometric cubic B-spline for numerical solution of Burgers' equation in one and two dimensions, Alexandria Engineering Journal, 57(2), pp. 1087–1098, 2018.

[11] E. Bänsch, Finite element discretization of the Navier–Stokes equations with a free capillary surface, Numerische Mathematik, 88(2), pp. 203–235, 2001.

[12] A. J. Bard, Calculation of equilibrium constants from free energy data, in S. A. Rice, editor, *Chemical Equilibrium*, pp. 182–185, Harper & Row, 1966.

[13] M. Bause, Stabilized finite element methods with shock-capturing for nonlinear convection-diffusion-reaction models, in *Numerical Mathematics and Advanced Applications 2009*, pp. 125–133, Springer, Berlin Heidelberg, 2010.

[14] Y. Bazilevs, I. Akkerman, D. J. Benson, G. Scovazzi, and M. J. Shashkov, Isogeometric analysis of Lagrangian hydrodynamics, Journal of Computational Physics, 243, pp. 224–243, 2013.

[15] Y. Bazilevs, V. M. Calo, T. J. R. Hughes, and Y. Zhang, Isogeometric fluid–structure interaction: theory, algorithms, and computations, Computational Mechanics, 43, pp. 3–37, 2008.

[16] Y. Bazilevs, V. M. Calo, T. E. Tezduyar, and T. J. R. Hughes, $YZ\beta$ discontinuity capturing for advection-dominated processes with application to arterial drug delivery, International Journal for Numerical Methods in Fluids, 54(6-8), pp. 593–608, 2007.

[17] Y. Bazilevs, V. M. Calo, Y. Zhang, and T. J. R. Hughes, Isogeometric fluid–structure interaction analysis with applications to arterial blood flow, Computational Mechanics, 38, pp. 310–322, 2006.

[18] Y. Bazilevs, M.-C. Hsu, and M. A. Scott, Isogeometric fluid–structure interaction analysis with emphasis on non-matching discretizations, and with application to wind turbines, Computer Methods in Applied Mechanics and Engineering, 249-252, pp. 28–41, 2012.

[19] Y. Bazilevs, C. Michler, V. M. Calo, and T. J. R. Hughes, Isogeometric variational multiscale modeling of wall-bounded turbulent flows with weakly enforced boundary conditions on unstretched meshes, Computer Methods in Applied Mechanics and Engineering, 199, pp. 780–790, 2010.

[20] Y. Bazilevs, K. Takizawa, T. E. Tezduyar, M.-C. Hsu, Y. Otoguro, H. Mochizuki, and M. C. H. Wu, Wind turbine and turbomachinery computational analysis with the ALE and space–time variational multiscale methods and isogeometric discretization, Journal of Advanced Engineering and Computation, 4, pp. 1–32, 2020.

[21] S. Bianchini and A. Bressan, Vanishing viscosity solutions of nonlinear hyperbolic systems, Annals of Mathematics, 161(1), pp. 223–342, 2005.

[22] G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford University Press, Oxford, 1994.

[23] J. Blazek, *Computational Fluid Dynamics: Principles and Applications*, Butterworth-Heinemann, Oxford, third edition, 2015.

[24] F. G. Blottner, M. Johnson, and M. Ellis, Chemically reacting viscous flow program for multi-component gas mixtures, NASA Technical Report NSA-26-027700, Sandia Labs., Albuquerque, N. Mex., 1971.

[25] S. W. Bova and B. S. Kirk, On the preservation of total enthalpy in SUPG methods, in *20th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2011-3841, Honolulu, HI, 2012.

[26] R. D. Braun and R. M. Manning, Mars exploration entry, descent, and landing challenges, Journal of Spacecraft and Rockets, 44(2), pp. 310–323, 2007.

[27] F. Brezzi and A. Russo, Stabilization techniques for the finite element method, in R. Spigler, editor, *Applied and Industrial Mathematics, Venice–2, 1998*, pp. 47–58, Springer Netherlands, Venice, Italy, 2000.

[28] A. N. Brooks and T. J. R. Hughes, Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations, Computer Methods in Applied Mechanics and Engineering, 32, pp. 199–259, 1982.

[29] Z. Cai and J. Ku, A dual finite element method for a singularly perturbed reaction-diffusion problem, SIAM Journal on Numerical Analysis, 58(3), pp. 1654–1673, 2020.

[30] G. V. Candler and I. Nompelis, Computational fluid dynamics for atmospheric entry, Technical Report RTO-EN-AVT-162, The von Karman Institute for Fluid Dynamics, 2009.

[31] V. Carandente, R. Savino, A. Esposito, G. Zuppardi, and V. Caso, Experimental and numerical simulation, by an arc-jet facility, of hypersonic flow in Titan's atmosphere, Experimental Thermal and Fluid Science, 48, pp. 97–101, 2013.

[32] V. Casseau, D. Espinoza, T. Scanlon, and R. Brown, A two-temperature open-source CFD model for hypersonic reacting flows, part two: Multi-dimensional analysis, Aerospace, 3(4), p. 45, 2016.

[33] L. Catabriga, A. L. G. A. Coutinho, and T. E. Tezduyar, Compressible flow SUPG stabilization parameters computed from degree-of-freedom submatrices, Computational Mechanics, 38, pp. 334–343, 2006.

[34] F. Chalot, T. J. R. Hughes, and F. Shakib, Symmetrization of conservation laws with entropy for high-temperature hypersonic computations, Computing Systems in Engineering, 1(2), pp. 495–521, 1990.

[35] F. Chalot, Z. Johan, M. Mallet, M. Ravachol, and G. Roge, Development of a finite element Navier–Stokes solver with applications to turbulent and hypersonic flows, in *Proceedings of AIAA 30th Aerospace Sciences Meeting*, AIAA Paper 92-0670, Reno, Nevada, 1992.

[36] F. L. Chalot, Industrial aerodynamics, in E. Stein, R. de Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics*, volume 3, pp. 407–458, Wiley, New York, 2004.

[37] G.-Q. Chen and D. Wang, The Cauchy problem for the Euler equations for compressible fluids, in S. Friedlander and D. Serre, editors, *Handbook of Mathematical Fluid Dynamics*, volume 1, pp. 421–543, North-Holland, 2002.

[38] S. Chen, *Modeling of Material-Environment Interactions for Hypersonic Thermal Protection Systems*, Ph.D. thesis, Aerospace Engineering, University of Michigan, Ann Arbor, MI, 2020.

[39] Y. K. Cheung and O. C. Zinkiewicz, Plates and tanks on elastic foundations–an application of finite element method, International Journal of Solids and Structures, 1(4), pp. 451–461, 1965.

[40] T. J. Chung, *Computational Fluid Dynamics*, Cambridge University Press, New York, second edition, 2010.

[41] R. W. Clough, The finite element method in plane stress analysis, in *Proceedings Second ASCE Conference on Electronic Computation*, Pittsburgh, Pennsylvania, 1960.

[42] R. Codina, A discontinuity-capturing crosswind-dissipation for the finite element solution of the convection-diffusion equation, Computer Methods in Applied Mechanics and Engineering, 110(3-4), pp. 325–342, 1993.

[43] R. Codina, Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods, Computer Methods in Applied Mechanics and Engineering, 190(13-14), pp. 1579–1599, 2000.

[44] D. Codoni, G. Moutsanidis, M.-C. Hsu, Y. Bazilevs, C. Johansen, and A. Korobenko, Stabilized methods for high-speed compressible flows: toward hypersonic simulations, Computational Mechanics, 67, pp. 785–809, 2021.

[45] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric Analysis. Toward Integration of CAD and FEA*, Wiley, Chichester, UK; NJ, USA, 2009.

[46] R. Courant, Variational methods for the solution of problems of equilibrium and vibrations, Bulletin of the American Mathematical Society, 49, pp. 1–23, 1943.

[47] R. Courant, K. Friedrichs, and H. Lewy, Über die partiellen differenzengleichungen der mathematischen physik, Mathematische Annalen, 100(1), pp. 32–74, 1928.

[48] R. Courant and D. Hilbert, *Methoden der mathematischen Physik*, Springer-Verlag Berlin Heidelberg, fourth edition, 1993.

[49] M. G. Crandall and P. E. Souganidis, Convergence of difference approximations of quasilinear evolution equations, Nonlinear Analysis: Theory, Methods & Applications, 10(5), pp. 425–445, 1986.

[50] C. M. Dafermos, *Hyperbolic Conservation Laws in Continuum Physics*, Grundlehren der mathematischen Wissenschaften, Springer Berlin Heidelberg, third edition, 2010.

[51] J. M. Desantes, J. M. García-Oliver, J. M. Pastor, A. Pandal, E. Baldwin, and D. P. Schmidt, Coupled/decoupled spray simulation comparison of the ECN spray a condition with the Σ-Y Eulerian atomization model, International Journal of Multiphase Flow, 80, pp. 89–99, 2016.

[52] E. Dick, Introduction to finite element methods in computational fluid dynamics, in J. F. Wendt, editor, *Computational Fluid Dynamics*, pp. 235–274, Springer Berlin Heidelberg, 2009.

[53] S. Dietrich and I. D. Boyd, Scalar and parallel optimized implementation of the direct simulation Monte Carlo method, Journal of Computational Physics, 126(2), pp. 328–342, 1996.

[54] J. Donea, A Taylor–Galerkin method for convective transport problems, International Journal for Numerical Methods in Engineering, 20(1), pp. 101–119, 1984.

[55] T. A. Driscoll and R. J. Braun, *Fundamentals of Numerical Computation*, SIAM, New York, 2017.

[56] V. Emelyanov, A. Karpenko, and K. Volkov, Simulation of hypersonic flows with equilibrium chemical reactions on graphics processor units, Acta Astronautica, 163, pp. 259–271, 2019.

[57] A. Eucken, Über das wärmeleitvermögen, die spezifische wärme und die innere reibung der gase, Physik. Z, 14(8), pp. 324–332, 1913.

[58] L. C. Evans, *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*, American Mathematical Society, Providence, RI, second edition, 2010.

[59] P. Farrell, A. Hegarty, J. M. Miller, E. O'Riordan, and G. I. Shishkin, *Robust Computational Techniques for Boundary Layers*, Chapman and Hall/CRC, New York, 2000.

[60] C. A. Felippa, A historical outline of matrix structural analysis: A play in three acts, Computers & Structures, 79(14), pp. 1313–1324, 2001.

[61] J. H. Ferziger, M. Perić, and R. L. Street, *Computational Methods for Fluid Dynamics*, Springer, Cham, Switzerland, fourth edition, 2020.

[62] C. A. J. Fletcher, Generating exact solutions of the two-dimensional Burgers' equations, International Journal for Numerical Methods in Fluids, 3(3), pp. 213–216, 1983.

[63] C. A. J. Fletcher, *Computational Techniques for Fluid Dynamics II: Specific Techniques for Different Flow Categories*, Springer-Verlag Berlin Heidelberg, New York, second edition, 1991.

[64] T.-P. Fries and H. G. Matthies, Review of Petrov–Galerkin stabilization approaches and an extension to meshfree methods, Technical Report Informatikbericht-Nr. 2004-01, Technische Universität Braunschweig, 2004.

[65] G. P. Galdi and W. J. Layton, Approximation of the larger eddies in fluid motion II: A model for space-filtered flow, Mathematical Models and Methods in Applied Sciences, 10(03), pp. 343–350, 2000.

[66] J. Garicano-Mena, *On the Computation of Heat Flux in Hypersonic Flows Using Residual Distribution Schemes*, Ph.D. thesis, Université libre de Bruxelles, von Karman Institute for Fluid Dynamics, Bruxelles, 2014.

[67] M. B. Gerdroodbary and S. M. Hosseinalipour, Numerical simulation of hypersonic flow over highly blunted cones with spike, Acta Astronautica, 67(1-2), pp. 180–193, 2010.

[68] C. Geuzaine and J.-F. Remacle, Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities, International Journal for Numerical Methods in Engineering, 79(11), pp. 1309–1331, 2009.

[69] E. D. Gillman, J. E. Foster, and I. M. Blankson, Review of leading approaches for mitigating hypersonic vehicle communications blackout and a method of ceramic particulate injection via cathode spot arcs for blackout mitigation, NASA Technical Report NASA/TM 2010-216220, NASA, 2010.

[70] S. F. Gimelshein and I. J. Wysong, Applicability of 5, 7, and 11 species air models in nonequilibrium hypersonic reacting flows, in *AIAA Scitech 2020 Forum*, AIAA Paper 2020-2190, Orlando, Florida, 2020.

[71] J. Glimm, Solutions in the large for nonlinear hyperbolic systems of equations, Communications on Pure and Applied Mathematics, 18(4), pp. 697–715, 1965.

[72] P. A. Gnoffo, R. N. Gupta, and J. L. Shinn, Conservation equations and physical models for hypersonic air flows in thermal and chemical nonequilibrium, NASA Technical Paper NASA-TP-2867, NASA, 1989.

[73] S. Godunov and I. Bohachevsky, Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics, Matematičeskij Sbornik, 47(3), pp. 271–306, 1959.

[74] R. Gosse and G. V. Candler, Diffusion flux modeling: Application to direct entry problems, in *43rd AIAA Aerospace Sciences Meeting and Exhibit*, AIAA Paper 2005-389, Reno, Nevada, 2005.

[75] J.-L. Guermond and B. Popov, Viscous regularization of the Euler equations and entropy principles, SIAM Journal on Applied Mathematics, 74(2), pp. 284–305, 2014.

[76] R. N. Gupta and R. A. Thompson, A review of reaction rates and thermodynamic and transport properties for an 11-species air model for chemical and thermal nonequilibrium calculations to 30,000 K, NASA Reference Publication NASA-RP-1232, NASA, 1990.

[77] J. Hao, J. Wang, Z. Gao, C. Jiang, and C. Lee, Comparison of transport properties models for numerical simulations of Mars entry vehicles, Acta Astronautica, 130, pp. 24–33, 2017.

[78] J. Hao, J. Wang, and C. Lee, Numerical study of hypersonic flows over reentry configurations with different chemical nonequilibrium models, Acta Astronautica, 126, pp. 1–10, 2016.

[79] G. Hauke, Simple stabilizing matrices for the computation of compressible flows in primitive variables, Computer Methods in Applied Mechanics and Engineering, 190, pp. 6881–6893, 1994.

[80] W. D. Hayes and R. F. Probstein, *Hypersonic Flow Theory, Vol. 1: Inviscid Flows*, Academic Press, New York, second edition, 1966.

[81] T. D. Holman and I. D. Boyd, Effects of continuum breakdown on hypersonic aerothermodynamics for reacting flow, Physics of Fluids, 23(2), p. 027101, 2011.

[82] M. A. Howard, *Finite Element Modeling and Optimization of High-Speed Aerothermoelastic Systems*, Ph.D. thesis, Department of Aerospace Engineering Sciences, University of Colorado, Boulder, CO, 2010.

[83] T. J. R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles, and the origins of stabilized methods, Computer Methods in Applied Mechanics and Engineering, 127, pp. 387–401, 1995.

[84] T. J. R. Hughes and A. N. Brooks, A multi-dimensional upwind scheme with no crosswind diffusion, in T. J. R. Hughes, editor, *Finite Element Methods for Convection Dominated Flows*, AMD-Vol.34, pp. 19–35, ASME, New York, 1979.

[85] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement, Computer Methods in Applied Mechanics and Engineering, 194, pp. 4135–4195, 2005.

[86] T. J. R. Hughes, L. P. Franca, and M. Mallet, A new finite element formulation for computational fluid dynamics: VI. Convergence analysis of the generalized SUPG formulation for linear time-dependent multi-dimensional advective-diffusive systems, Computer Methods in Applied Mechanics and Engineering, 63, pp. 97–112, 1987.

[87] T. J. R. Hughes and M. Mallet, A new finite element formulation for computational fluid dynamics: IV. A discontinuity-capturing operator for multidimensional advective-diffusive systems, Computer Methods in Applied Mechanics and Engineering, 58, pp. 329–336, 1986.

[88] T. J. R. Hughes, M. Mallet, and A. Mizukami, A new finite element formulation for computational fluid dynamics: II. Beyond SUPG, Computer Methods in Applied Mechanics and Engineering, 54, pp. 341–355, 1986.

[89] T. J. R. Hughes, G. Scovazzi, and T. E. Tezduyar, Stabilized methods for compressible flows, Journal of Scientific Computing, 43, pp. 343–368, 2010.

[90] T. J. R. Hughes and T. E. Tezduyar, Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible Euler equations, Computer Methods in Applied Mechanics and Engineering, 45, pp. 217–284, 1984.

[91] V. John, Slip with friction and penetration with resistance boundary conditions for the Navier–Stokes equations—numerical tests and aspects of the implementation, Journal of Computational and Applied Mathematics, 147(2), pp. 287–300, 2002.

[92] V. John and P. Knobloch, On spurious oscillations at layers diminishing (SOLD) methods for convection–diffusion equations: Part I–A review, Computer Methods in Applied Mechanics and Engineering, 196(17), pp. 2197–2215, 2007.

[93] V. John and P. Knobloch, On spurious oscillations at layers diminishing (SOLD) methods for convection–diffusion equations: Part II–Analysis for P_1 and Q_1 finite elements, Computer Methods in Applied Mechanics and Engineering, 197(21-24), pp. 1997–2014, 2008.

[94] V. John and J. Novo, Error analysis of the SUPG finite element discretization of evolutionary convection-diffusion-reaction equations, SIAM Journal on Numerical Analysis, 49(3), pp. 1149–1176, 2011.

[95] V. John and E. Schmeyer, Finite element methods for time-dependent convection-diffusion-reaction equations with small diffusion, Computer Methods in Applied Mechanics and Engineering, 198(3), pp. 475–494, 2008.

[96] C. Johnson, U. Navert, and J. Pitkäranta, Finite element methods for linear hyperbolic problems, Computer Methods in Applied Mechanics and Engineering, 45, pp. 285–312, 1984.

[97] M. Juntunen and R. Stenberg, Nitsche's method for general boundary conditions, Mathematics of Computation, 78(267), pp. 1353–1374, 2009.

[98] C. G. Justus and R. D. Braun, Atmospheric environments for entry, descent and landing (EDL), in *Proceedings of the 5th International Planetary Probe Workshop*, Bordeaux, France, 2007.

[99] T. Kanai, K. Takizawa, T. E. Tezduyar, T. Tanaka, and A. Hartmann, Compressible-flow geometric-porosity modeling and spacecraft parachute computation with isogeometric discretization, Computational Mechanics, 63, pp. 301–321, 2019.

[100] S. D. Kasen, *Thermal Management at Hypersonic Leading Edges*, Ph.D. thesis, School of Engineering and Applied Science, University of Virginia, Charlottesville, VA, 2013.

[101] T. Kato, The Cauchy problem for quasi-linear symmetric hyperbolic systems, Archive for Rational Mechanics and Analysis, 58(3), pp. 181–205, 1975.

[102] T. Kato, Quasi-linear equations of evolution, with applications to partial differential equations, in W. N. Everitt, editor, *Spectral Theory and Differential Equations*, pp. 25–70, Springer-Verlag Berlin Heidelberg, Dundee, Scotland, 1975.

[103] M. P. Kessler and A. M. Awruch, Analysis of hypersonic flows using finite elements with Taylor–Galerkin scheme, International Journal for Numerical Methods in Fluids, 44(12), pp. 1355–1376, 2004.

[104] R. A. Khurram and A. Masud, A multiscale/stabilized formulation of the incompressible Navier–Stokes equations for moving boundary flows and fluid–structure interaction, Computational Mechanics, 38, pp. 403–416, 2006.

[105] R. C. Kirby, Algorithm 839: FIAT, a new paradigm for computing finite element basis functions, ACM Transactions on Mathematical Software (TOMS), 30(4), pp. 502–516, 2004.

[106] B. Kirk, S. Bova, and R. Bond, The influence of stabilization parameters in the SUPG finite element method for hypersonic flows, in *48th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, AIAA Paper 2010-11, Orlando, Florida, 2010.

[107] B. Kirk, S. Bova, and R. Bond, A streamline-upwind Petrov–Galerkin finite element scheme for non-ionized hypersonic flows in thermochemical nonequilibrium, in *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, AIAA Paper 2011-134, Orlando, Florida, 2011.

[108] B. S. Kirk and T. A. Oliver, Validation of SUPG finite element simulations of shockwave/turbulent boundary layer interactions in hypersonic flows, in *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, AIAA Paper 2013-0306, Grapevine (Dallas/Ft. Worth Region), Texas, 2013.

[109] B. S. Kirk, R. H. Stogner, P. T. Bauman, and T. A. Oliver, Modeling hypersonic entry with the fully-implicit Navier–Stokes (FIN-S) stabilized finite element flow solver, Computers & Fluids, 92, pp. 281–292, 2014.

[110] B. Knorr, Y. Xie, C. Stumpp, P. Maloszewski, and C. T. Simmons, Representativeness of 2D models to simulate 3D unstable variable density flow in porous media, Journal of Hydrology, 542, pp. 541–551, 2016.

[111] O. Kolditz, *Computational Methods in Environmental Fluid Mechanics*, Springer-Verlag Berlin Heidelberg, New York, 2002.

[112] S. Kumar, *Numerical Simulation of Chemically Reactive Hypersonic Flows*, Ph.D. thesis, Faculty of Mechanical Engineering, RWTH Aachen University, Aachen, Germany, 2006.

[113] T. Kuraishi, K. Takizawa, and T. E. Tezduyar, Tire aerodynamics with actual tire geometry, road contact and tire deformation, Computational Mechanics, 63, pp. 1165–1185, 2019.

[114] C. B. Laney, *Computational Gasdynamics*, Cambridge University Press, New York, 1998.

[115] H. P. Langtangen and K.-A. Mardal, *Introduction to Numerical Methods for Variational Problems*, Springer, Cham, Switzerland, 2019.

[116] M. G. Larson and F. Bengzon, *The Finite Element Method: Theory, Implementation, and Applications*, volume 10 of *Texts in Computational Science and Engineering*, Springer-Verlag Berlin Heidelberg, 2013.

[117] P. D. Lax, Hyperbolic systems of conservation laws II, Communications on Pure and Applied Mathematics, 10(4), pp. 537–566, 1957.

[118] W. Layton, Weak imposition of "no-slip" conditions in finite element methods, Computers & Mathematics with Applications, 38(5-6), pp. 129–142, 1999.

[119] N. T. P. Le, H. Xiao, and R. S. Myong, A triangular discontinuous Galerkin method for non-Newtonian implicit constitutive models of rarefied and microscale gases, Journal of Computational Physics, 273, pp. 160–184, 2014.

[120] G. J. Le Beau, S. E. Ray, S. K. Aliabadi, and T. E. Tezduyar, SUPG finite element computation of compressible flows with the entropy and conservation variables formulations, Computer Methods in Applied Mechanics and Engineering, 104, pp. 397–422, 1993.

[121] G. J. Le Beau and T. E. Tezduyar, Finite element computation of compressible flows with the SUPG formulation, in *Advances in Finite Element Analysis in Fluid Dynamics*, FED-Vol.123, pp. 21–27, ASME, New York, 1991.

[122] J.-H. Lee, Basic governing equations for the flight regimes of aeroassisted orbital transfer vehicles, in *19th Thermophysics Conference*, AIAA Paper 1984-1729, Snowmass, Colorado, 1985.

[123] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-state and Time-dependent Problems*, volume 98, SIAM, Philadelphia, PA, 2007.

[124] S. Li and X. Jiang, Review and prospect of guidance and control for Mars atmospheric entry, Progress in Aerospace Sciences, 69, pp. 40–57, 2014.

[125] T. Linß and M. Stynes, The SDFEM on Shishkin meshes for linear convection-diffusion problems, Numerische Mathematik, 87(3), pp. 457–484, 2001.

[126] A. Logg, K.-A. Mardal, and G. Wells, *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS book*, volume 84 of *Lecture Notes in Computational Science and Engineering*, Springer-Verlag Berlin Heidelberg, 2012.

[127] A. Logg, K. B. Ølgaard, M. E. Rognes, and G. N. Wells, FFC: The FEniCS form compiler, in *Automated Solution of Differential Equations by the Finite Element Method*, chapter 11, pp. 227–238, Springer-Verlag Berlin Heidelberg, 2012.

[128] A. Logg and G. N. Wells, DOLFIN: Automated finite element computing, ACM Transactions on Mathematical Software (TOMS), 37(2), p. 20, 2010.

[129] J. A. Lordi and R. E. Mates, Rotational relaxation in nonpolar diatomic gases, Physics of Fluids, 13, pp. 291–308, 1970.

[130] R. W. MacCormack and A. J. Paullay, Computational efficiency achieved by time splitting of finite difference operators, in *10th Aerospace Sciences Meeting*, AIAA Paper 1972-154, San Diego, California, 1972.

[131] W. T. Maier, J. T. Needels, C. Garbacz, F. Morgado, J. J. Alonso, and M. Fossati, SU2-NEMO: An open-source framework for high-Mach nonequilibrium multi-species flows, Aerospace, 8(7), p. 193, 2021.

[132] A. Majda, *Compressible Fluid Flow and Systems of Conservation Laws in Several Space Variables*, volume 53 of *Applied Mathematical Sciences*, Springer-Verlag New York, 1984.

[133] T. Makino, S. Ukai, and S. Kawashima, Sur la solution à support compact de l'equation d'Euler compressible, Japan Journal of Applied Mathematics, 3(2), pp. 249–257, 1986.

[134] M. Mallet, J. Periaux, P. Perrier, and B. Stoufflet, Flow modelization and computational methodologies for the aerothermal design of hypersonic vehicles: Application to the European Hermes, in *AlAA Thermophysics, Plasma dynamics and Lasers Conference*, AIAA Paper 88-2628, San Antonio, Texas, 1988.

[135] O. Marxen, T. E. Magin, E. S. Shaqfeh, and G. Iaccarino, A method for the direct numerical simulation of hypersonic boundary-layer instability with finite-rate chemistry, Journal of Computational Physics, 255, pp. 572–589, 2013.

[136] H. S. Massimi, H. Shen, C.-Y. Wen, Y.-S. Chen, and S.-M. Liang, Numerical analysis of hypersonic flows around blunt-nosed models and a space vehicle, Aerospace Science and Technology, 43, pp. 360–371, 2015.

[137] P. W. McDonald, The computation of transonic flow through two-dimensional gas turbine cascades, in *ASME 1971 International Gas Turbine Conference and Products Show*, ASME, Houston, Texas, 1971.

[138] R. C. Millikan and D. R. White, Systematics of vibrational relaxation, The Journal of Chemical Physics, 39(12), pp. 3209–3213, 1963.

[139] R. C. Mittal and A. Tripathi, Numerical solutions of two-dimensional Burgers' equations using modified Bi-cubic B-spline finite elements, Engineering Computations, 32(5), pp. 1275–1306, 2015.

[140] S. Mittal, On the performance of high aspect ratio elements for incompressible flows, Computer Methods in Applied Mechanics and Engineering, 188(1-3), pp. 269–287, 2000.

[141] G. Moretti and M. Abbett, A time-dependent computational method for blunt-body flows, AIAA Journal, 4(12), pp. 2136–2141, 1966.

[142] T. K. S. Murthy, *Computational Methods in Hypersonic Aerodynamics*, volume 9 of *Fluid Mechanics and Its Applications*, Springer Netherlands, 2010.

[143] National Aeronautics and Space Administration, Apollo program summary report, NASA B. Johnson Space Center Report JSC-09423, NASA, 1975.

[144] M. Nazarov, *Adaptive Algorithms and High Order Stabilization for Finite Element Computation of Turbulent Compressible Flow*, Ph.D. thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2011.

[145] M. Nazarov and A. Larcher, Numerical investigation of a viscous regularization of the Euler equations by entropy viscosity, Computer Methods in Applied Mechanics and Engineering, 317, pp. 128–152, 2017.

[146] I. Nompelis, *Computational Study of Hypersonic Double-cone Experiments for Code Validation*, Ph.D. thesis, Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, MN, 2004.

[147] A. Novotny and I. Straškraba, *Introduction to the Mathematical Theory of Compressible Flow*, volume 27 of *Oxford Lecture Series in Mathematics and its Applications*, Oxford University Press, 2004.

[148] Y. Otoguro, K. Takizawa, and T. E. Tezduyar, Element length calculation in b-spline meshes for complex geometries, Computational Mechanics, 65(4), pp. 1085–1103, 2020.

[149] C. C. Paige and M. A. Saunders, Solution of sparse indefinite systems of linear equations, SIAM Journal on Numerical Analysis, 12(4), pp. 617–629, 1975.

[150] G. E. Palmer and M. J. Wright, Comparison of methods to compute high-temperature gas viscosity, Journal of Thermophysics and Heat Transfer, 17(2), pp. 232–239, 2003.

[151] C. Park, On convergence of computation of chemically reacting flows, in *23rd AIAA Aerospace Sciences Meeting*, AIAA Paper 85-0247, Reno, Nevada, 1985.

[152] C. Park, *Nonequilibrium Hypersonic Aerothermodynamics*, Wiley, New York, 1990.

[153] C. Park, R. L. Jaffe, and H. Partridge, Chemical-kinetic parameters of hyperbolic Earth entry, Journal of Thermophysics and Heat Transfer, 15(1), pp. 76–90, 2001.

[154] R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow*, Springer Series in Computational Physics, Springer-Verlag Berlin Heidelberg, New York, 1983.

[155] W. Qin, S. Datta, and W. Zhang, 3D simulation models simplified to 2D planar/axisymmetric problems in automotive structures, Technical Report SAE Technical Paper 2016-01-0397, FCA US LLC, 2016.

[156] A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations*, Springer Series in Computational Mathematics, Springer Berlin Heidelberg, 1994.

[157] M. R. Rajanna, E. L. Johnson, D. Codoni, A. Korobenko, Y. Bazilevs, N. Liu, J. Lua, N. D. Phan, and M.-C. Hsu, Finite element simulation and validation for aerospace applications: Stabilized methods, weak Dirichlet boundary conditions, and discontinuity capturing for compressible flows, in *AIAA Scitech 2022 Forum*, AIAA Paper 2022-1077, San Diego, California, 2022.

[158] F. Rispoli, A. Corsini, and T. E. Tezduyar, Finite element computation of turbulent flows with the discontinuity-capturing directional dissipation (DCDD), Computers & Fluids, 36, pp. 121–126, 2007.

[159] F. Rispoli, R. Saavedra, F. Menichini, and T. E. Tezduyar, Computation of inviscid supersonic flows around cylinders and spheres with the V-SGS stabilization and YZβ shock-capturing, Journal of Applied Mechanics, 76, p. 021209, 2009.

[160] H.-G. Roos and M. Schopf, Convergence and stability in balanced norms of finite element methods on shishkin meshes for reaction-diffusion problems, ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik, 95(6), pp. 551–565, 2014.

[161] S. Russell and M. Stynes, Balanced-norm error estimates for sparse grid finite element methods applied to singularly perturbed reaction-diffusion problems, Journal of Numerical Mathematics, 27(1), pp. 37–55, 2019.

[162] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, Philadelphia, PA, second edition, 2003.

[163] Y. Saad and M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing, 7(3), pp. 856–869, 1986.

[164] G. S. R. Sarma, Physico-chemical modelling in hypersonic flow simulation, Progress in Aerospace Sciences, 36(3-4), pp. 281–349, 2000.

[165] L. C. Scalabrin, *Numerical Simulation of Weakly Ionized Hypersonic Flow Over Reentry Capsules*, Ph.D. thesis, Aerospace Engineering, University of Michigan, Ann Arbor, MI, 2007.

[166] J. Seguin, S. Gao, W. G. Habashi, D. Isola, and G. Baruzzi, A finite element solver for hypersonic flows in thermo-chemical non-equilibrium, Part I, International Journal of Numerical Methods for Heat & Fluid Flow, 29(7), pp. 2352–2388, 2019.

[167] F. Shakib, *Finite Element Analysis of the Compressible Euler and Navier–Stokes Equations*, Ph.D. thesis, Mechanical Engineering, Stanford University, Stanford, CA, 1988.

[168] F. Shakib, T. J. Hughes, and Z. Johan, A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier–Stokes equations, Computer Methods in Applied Mechanics and Engineering, 89(1-3), pp. 141–219, 1991.

[169] V. K. Srivastava, M. K. Awasthi, and S. Singh, An implicit logarithmic finite-difference technique for two dimensional coupled viscous Burgers' equation, AIP Advances, 3(12), p. 122105, 2013.

[170] R. P. Starkey, Hypersonic vehicle telemetry blackout analysis, Journal of Spacecraft and Rockets, 52(2), pp. 426–438, 2015.

[171] P. Subrahmanyam and D. Rasky, Entry, descent, and landing technological barriers and crewed Mars vehicle performance analysis, Progress in Aerospace Sciences, 91, pp. 1–26, 2017.

[172] W. Sutherland, LII. The viscosity of gases and molecular force, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 36(223), pp. 507–531, 1893.

[173] D. Sziroczak and H. Smith, A review of design issues specific to hypersonic flight vehicles, Progress in Aerospace Sciences, 84, pp. 1–28, 2016.

[174] K. Takizawa, T. E. Tezduyar, and T. Kanai, Porosity models and computational methods for compressible-flow aerodynamics of parachutes with geometric porosity, Mathematical Models and Methods in Applied Sciences, 27, pp. 771–806, 2017.

[175] K. Takizawa, T. E. Tezduyar, and T. Kuraishi, Multiscale ST methods for thermo-fluid analysis of a ground vehicle and its tires, Mathematical Models and Methods in Applied Sciences, 25, pp. 2227–2255, 2015.

[176] K. Takizawa, T. E. Tezduyar, S. McIntyre, N. Kostov, R. Kolesar, and C. Habluetzel, Space–time VMS computation of wind-turbine rotor and tower aerodynamics, Computational Mechanics, 53, pp. 1–15, 2014.

[177] K. Takizawa, T. E. Tezduyar, and Y. Otoguro, Stabilization and discontinuity-capturing parameters for space–time flow computations with finite element and isogeometric discretizations, Computational Mechanics, 62, pp. 1169–1186, 2018.

[178] K. Takizawa, Y. Ueda, and T. E. Tezduyar, A node-numbering-invariant directional length scale for simplex elements, Mathematical Models and Methods in Applied Sciences, 29, pp. 2719–2753, 2019.

[179] M. Tamsir, V. K. Srivastava, and R. Jiwari, An algorithm based on exponential modified cubic B-spline differential quadrature method for nonlinear Burgers' equation, Applied Mathematics and Computation, 290, pp. 111–124, 2016.

[180] T. Tezduyar, S. Aliabadi, M. Behr, A. Johnson, and S. Mittal, Parallel finite-element computation of 3D flows, Computer, 26(10), pp. 27–36, 1993.

[181] T. E. Tezduyar, Stabilized finite element formulations for incompressible flow computations, Advances in Applied Mechanics, 28, pp. 1–44, 1991.

[182] T. E. Tezduyar, Computation of moving boundaries and interfaces and stabilization parameters, International Journal for Numerical Methods in Fluids, 43, pp. 555–575, 2003.

161

[183] T. E. Tezduyar, Determination of the stabilization and shock-capturing parameters in SUPG formulation of compressible flows, in *Proceedings of the European Congress on Computational Methods in Applied Sciences and Engineering, EC-COMAS 2004 (CD-ROM)*, Jyvaskyla, Finland, 2004.

[184] T. E. Tezduyar, Finite element methods for fluid dynamics with moving boundaries and interfaces, in E. Stein, R. D. Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics*, Volume 3: Fluids, chapter 17, Wiley, 2004.

[185] T. E. Tezduyar, Finite elements in fluids: Stabilized formulations and moving boundaries and interfaces, Computers & Fluids, 36, pp. 191–206, 2007.

[186] T. E. Tezduyar, S. K. Aliabadi, M. Behr, and S. Mittal, Massively parallel finite element simulation of compressible and incompressible flows, Computer Methods in Applied Mechanics and Engineering, 119, pp. 157–177, 1994.

[187] T. E. Tezduyar and T. J. R. Hughes, Development of time-accurate finite element techniques for first-order hyperbolic systems with particular emphasis on the compressible Euler equations, NASA Technical Report NASA-CR-204772, NASA, 1982.

[188] T. E. Tezduyar and T. J. R. Hughes, Finite element formulations for convection dominated flows with particular emphasis on the compressible Euler equations, in *Proceedings of AIAA 21st Aerospace Sciences Meeting*, AIAA Paper 83-0125, Reno, Nevada, 1983.

[189] T. E. Tezduyar and Y. Osawa, Finite element stabilization parameters computed from element matrices and vectors, Computer Methods in Applied Mechanics and Engineering, 190, pp. 411–430, 2000.

[190] T. E. Tezduyar and Y. J. Park, Discontinuity capturing finite element formulations for nonlinear convection-diffusion-reaction equations, Computer Methods in Applied Mechanics and Engineering, 59, pp. 307–325, 1986.

[191] T. E. Tezduyar and M. Senga, Stabilization and shock-capturing parameters in SUPG formulation of compressible flows, Computer Methods in Applied Mechanics and Engineering, 195, pp. 1621–1632, 2006.

[192] T. E. Tezduyar and M. Senga, SUPG finite element computation of inviscid supersonic flows with YZβ shock-capturing, Computers & Fluids, 36, pp. 147–159, 2007.

[193] T. E. Tezduyar, M. Senga, and D. Vicker, Computation of inviscid supersonic flows around cylinders and spheres with the SUPG formulation and YZβ shock-capturing, Computational Mechanics, 38, pp. 469–481, 2006.

[194] S. Tissera, *Assessment of High-Resolution Methods in Hypersonic Real-Gas Flows*, Ph.D. thesis, School of Engineering, Cranfield University, Cranfield, UK, 2010.

[195] J. Tu, G.-H. Yeoh, and C. Liu, *Computational Fluid Dynamics: A Practical Approach*, Butterworth-Heinemann, Oxford, UK; Cambridge, MA, third edition, 2019.

[196] Y. Ueda, Y. Otoguro, K. Takizawa, and T. E. Tezduyar, Element-splitting-invariant local-length-scale calculation in b-spline meshes for complex geometries, Mathematical Models and Methods in Applied Sciences, 30(11), pp. 2139–2174, 2020.

[197] U.S. Government Printing Office, *U.S. Standard Atmosphere, 1976*, National Oceanic and Atmospheric Administration, Washington, DC, 1976.

[198] M. Uzunca, B. Karasözen, and M. Manguoğlu, Adaptive discontinuous Galerkin methods for non-linear diffusion–convection–reaction equations, Computers & Chemical Engineering, 68, pp. 24–37, 2014.

[199] H. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Pearson Education, UK, second edition, 2007.

[200] W. G. Vincenti and C. H. Kruger, *Introduction to physical gas dynamics*, Wiley, New York, 1965.

[201] X. Wang, C. Yan, W. Zheng, K. Zhong, and Y. Geng, Laminar and turbulent heating predictions for Mars entry vehicles, Acta Astronautica, 128, pp. 217–228, 2016.

[202] C. Y. Wen, H. S. Massimi, and H. Shen, Extension of CE/SE method to non-equilibrium dissociating flows, Journal of Computational Physics, 356, pp. 240–260, 2018.

[203] W. Wienken, J. Stiller, and U. Fladrich, A finite-element based Navier–Stokes solver for LES, in K. Matsuno, A. Ecer, N. Satofuka, J. Periaux, and P. Fox, editors, *Parallel Computational Fluid Dynamics 2002*, pp. 361–368, North-Holland, Amsterdam, 2003.

[204] C. R. Wilke, A viscosity equation for gas mixtures, The Journal of Chemical Physics, 18(4), pp. 517–519, 1950.

[205] G. S. K. Wong, Speed of sound in standard air, The Journal of the Acoustical Society of America, 79(5), pp. 1359–1366, 1986.

[206] M. Wright, K. Edquist, C. Tang, B. Hollis, P. Krasa, and C. Campbell, A review of aerothermal modeling for Mars entry missions, in *48th AIAA Aerospace*

Sciences Meeting including the New Horizons Forum and Aerospace Exposition, AIAA Paper 2010-443, Orlando, Florida, 2010.

[207] M. J. Wright, *A Family of Data-parallel Relaxation Methods for the Navier–Stokes Equations*, Ph.D. thesis, Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, MN, 1997.

[208] J. Yang and M. Liu, Numerical analysis of hypersonic thermochemical non-equilibrium environment for an entry configuration in ionized flow, Chinese Journal of Aeronautics, 32(12), pp. 2641–2654, 2019.

[209] H. Yücel, M. Stoll, and P. Benner, Discontinuous Galerkin finite element methods with shock-capturing for nonlinear convection dominated models, Computers & Chemical Engineering, 58, pp. 278–287, 2013.

[210] Z. Zhang, Finite element superconvergence on Shishkin mesh for 2-D convection-diffusion problems, Mathematics of Computation, 72(243), pp. 1147–1178, 2003.

APPENDIX A

PHYSICAL AND CHEMICAL CONSTANTS

This appendix presents the physical, chemical, and curve-fitting constants used in simulations carried out in this thesis. For further details, the interested reader may also refer to [72, 151–153]

In Table A.1, the species molar masses, M_s, heats of formation, h_s^0, and characteristic vibrational energies, θ_{vs}, are given.

Table A.1: Species chemical data [107].

Species	M_s (kg/kmol)	h_s^0 (J/kg) $\times 10^{-6}$	θ_{vs} (K)
O	16.000	15.420	-
N	14.008	33.622	-
NO	30.008	2.996	2,817
O$_2$	32.000	0.000	2,239
N$_2$	28.016	0.000	3,395

Table A.2 presents the Blottner's curve-fitting parameters used in Eq. (2.37) for determining the species viscosities.

Table A.2: Blottner's curve-fitting coefficients [24].

Species	A_s	B_s	C_s
O	0.0203144	0.4294404	-11.6031403
N	0.0115572	0.6031679	-12.4327495
NO	0.0436378	-0.0335511	-9.5767430
O$_2$	0.0449290	-0.0826158	-9.2019475
N$_2$	0.0268142	0.3177838	-11.31555130

Table A.3 shows the constants to calculate the forward reaction rate coefficients used in Eq. (2.63).

Table A.3: Coefficients for calculating the forward reaction rates in Arrhenius formula [107].

Reaction	Π	C_{fr} (m³/kmol·s)	Υ_r	E_{ar} (cal/mol)
$N_2 + \Pi \rightleftharpoons 2N + \Pi$	O	3×10^{19}	-1.6	$224,815.2$
	N	3×10^{19}	-1.6	$224,815.2$
	NO	7×10^{18}	-1.6	$224,815.2$
	O_2	7×10^{18}	-1.6	$224,815.2$
	N_2	7×10^{18}	-1.6	$224,815.2$
$O_2 + \Pi \rightleftharpoons 2O + \Pi$	O	2×10^{18}	-1.5	$118,167.0$
	N	2×10^{18}	-1.5	$118,167.0$
	NO	2×10^{18}	-1.5	$118,167.0$
	O_2	1×10^{19}	-1.5	$118,167.0$
	N_2	1×10^{19}	-1.5	$118,167.0$
$NO + \Pi \rightleftharpoons N + O + \Pi$	O	5×10^{12}	0.0	$149,943.0$
	N	5×10^{12}	0.0	$149,943.0$
	NO	5×10^{12}	0.0	$149,943.0$
	O_2	5×10^{12}	0.0	$149,943.0$
	N_2	5×10^{12}	0.0	$149,943.0$
$N_2 + O \rightleftharpoons NO + N$	-	6.4×10^{14}	-1.0	$76,262.0$
$NO + O \rightleftharpoons O_2 + N$	-	8.4×10^{9}	0.0	$38,628.0$

In Table A.4, the curve-fitting parameters used in Eq. (2.65) for determining the species equilibrium constants, $K_{eq,r}$, are given.

Table A.4: Curve-fitting coefficients for calculating the equilibrium constants [152].

Reaction	A_1	A_2	A_3	A_4	A_5
$N_2 + \Pi \rightleftharpoons 2N + \Pi$	3.898	-12.611	0.683	-0.118	0.006
$O_2 + \Pi \rightleftharpoons 2O + \Pi$	1.335	-4.127	-0.616	0.093	-0.005
$NO + \Pi \rightleftharpoons N + O + \Pi$	1.549	-7.784	0.228	-0.043	0.002
$N_2 + O \rightleftharpoons NO + N$	2.349	-4.828	0.455	-0.075	0.004
$NO + O \rightleftharpoons O_2 + N$	0.215	-3.652	0.843	-0.136	0.007

APPENDIX B

CHARACTERISTIC PROPERTIES OF THE COMPRESSIBLE EULER EQUATIONS

From the mathematical point of view, the Euler equations of compressible flows are nonlinear, coupled, and hyperbolic system of equations. Some featuring properties of hyperbolic equations can be given as follows:

- They accept discontinuous solutions, i.e., shocks and contact discontinuities can be observed in their solutions,

- Their solutions can be expressed as a combination of eigenvectors called the *characteristics* or *characteristic speeds*.

There is a close relationship between the characteristics of the Euler equations and shocks. The effect of an explosion at any point in the flow field propagates at characteristic speeds, which are the speeds of shock waves. Studying these characteristics has two advantages: it guides in the understanding of equations with a much more complex nonlinear nature, and it provides a framework for numerical methods to be used or developed to solve them.

The advective Jacobians, $\mathbf{A}_i$, can be factorized as follows:

$$\mathbf{A}_i = \frac{\partial \mathbf{F}_i}{\partial \mathbf{U}} = \mathbf{L}_i \mathbf{\Lambda}_i \mathbf{R}_i, \tag{B.1}$$

where $\mathbf{\Lambda}_i$ is a diagonal matrix composed of the eigenvalues of the advective Jacobian $\mathbf{A}_i$, $\mathbf{R}_i$ is the matrix of row eigenvectors, and $\mathbf{L}_i$ is the matrix of column eigenvectors of $\mathbf{A}_i$, with

$$\mathbf{L}_i \mathbf{R}_i = \mathbf{I}. \tag{B.2}$$

B.1 Characteristic Analysis of Compressible Euler Equations in 1D

Consider the 1D Euler equations of compressible flows given as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} = \mathbf{S}(x, t, \mathbf{U}), \tag{B.3}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(e + p) \end{bmatrix}. \tag{B.4}$$

The equation of state is given as

$$p = p(\rho, e) = (\gamma - 1)\left(E - \frac{1}{2}\rho u^2\right) = (\gamma - 1)\,\rho e, \tag{B.5}$$

where the term E denotes the total energy per unit volume and is defined as

$$E = \rho e \frac{1}{2} + \rho u^2. \tag{B.6}$$

If we rewrite the vectors $\mathbf{U}$ and $\mathbf{F}$ in terms of new variables u_1, u_2, and u_3, we obtain

$$\mathbf{U} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} u_2 \\ \frac{u_2^2}{u_1} + (\gamma - 1)\left(u_3 + \frac{1}{2}\frac{u_2^2}{u_1}\right) \\ \frac{u_2}{u_1}\left[u_3 + (\gamma - 1)\left(u_3 + \frac{1}{2}\frac{u_2^2}{u_1}\right)\right] \end{bmatrix}. \tag{B.7}$$

Then, the advective Jacobian $\mathbf{A}$ is given as

$$\mathbf{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \frac{\partial f_1}{\partial u_3} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \frac{\partial f_2}{\partial u_3} \\ \frac{\partial f_3}{\partial u_1} & \frac{\partial f_3}{\partial u_2} & \frac{\partial f_3}{\partial u_3} \end{bmatrix} \tag{B.8}$$

$$= \begin{bmatrix} 0 & 1 & 0 \\ \frac{\gamma-3}{2}u^2 & (3 - \gamma)u & \gamma - 1 \\ \frac{-\gamma E u}{\rho}(\gamma - 1)u^3 & \frac{\gamma E}{\rho} - \frac{3(\gamma-1)u^2}{2} & \gamma u \end{bmatrix}. \tag{B.9}$$

Now, if we freeze the Jacobian $\mathbf{A}$ at a reference state, e.g., at $\mathbf{A}_0$, then we have

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_0 \frac{\partial \mathbf{U}}{\partial x} = \mathbf{S}. \tag{B.10}$$

By using Eq. (B.2), we find

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_0 \underbrace{\mathbf{R}_0 \mathbf{L}_0}_{\mathbf{I}} \frac{\partial \mathbf{U}}{\partial x} = \mathbf{S}. \tag{B.11}$$

Multiplying both sides of Eq. (B.11) by $\mathbf{L}_0$, one finds

$$\mathbf{L}_0 \left[\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_0 \mathbf{R}_0 \mathbf{L}_0 \frac{\partial \mathbf{U}}{\partial x} = \mathbf{S} \right]. \tag{B.12}$$

Since the matrix $\mathbf{A}_0$ is a constant matrix, its factors are also constant matrices:

$$\frac{\partial \mathbf{L}_0 \mathbf{U}}{\partial t} + \underbrace{\mathbf{L}_0 \mathbf{A}_0 \mathbf{R}_0}_{\mathbf{\Lambda}_0} \frac{\partial \mathbf{L}_0 \mathbf{U}}{\partial x} = \mathbf{L}_0 \mathbf{S}, \tag{B.13}$$

where the diagonal matrix consisting of the eigenvalues of $\mathbf{A}_0$ is given as follows:

$$\mathbf{\Lambda}_0 = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} = \begin{bmatrix} u - a & 0 & 0 \\ 0 & u & 0 \\ 0 & 0 & u + a \end{bmatrix}, \tag{B.14}$$

with the corresponding eigenvectors

$$\begin{bmatrix} 1 \\ u - a \\ \frac{u^2}{2} - au + \frac{a^2}{\gamma - 1} \end{bmatrix}, \quad \begin{bmatrix} 1 \\ u \\ \frac{u^2}{2} \end{bmatrix}, \quad \begin{bmatrix} 1 \\ u + a \\ \frac{u^2}{2} + au + \frac{a^2}{\gamma - 1} \end{bmatrix}. \tag{B.15}$$

Here, the term a denotes the speed of the sound, i.e., $a = c_{\text{sound}} = \sqrt{\frac{\gamma p}{\rho}}$. Note that all the eigenvalues are real in Eq. (B.14). Thus, by definition [58], Eq. (B.3) represent a *hyperbolic system*.

Letting $\mathbf{L}_0 \mathbf{U} = \mathbf{W}$ and $\mathbf{L}_0 \mathbf{S} = \mathbf{G}$, one finds

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{\Lambda}_0 \frac{\partial \mathbf{W}}{\partial x} = \mathbf{G}, \tag{B.16}$$

where the vector $\mathbf{W}$ is called the *Riemann invariant* or *characteristic variable*. Eq. (B.16) can be recast in a fully decoupled form as follows:

$$\frac{\partial w_i}{\partial t} + \lambda_{0,i} \frac{\partial w_i}{\partial x} = g_i, \tag{B.17}$$

where $i = 1, 2, 3$.

It is obvious that we have a wave equation for each i, where $a = \lambda_{0,i}$ and $u = w_i$. Therefore, any process or analysis applied to wave equations holds also for each characteristic equation of w_i. Notice that the eigenvalue $\lambda_{0,i}$ denotes the speed of the wave convected for the ith equation. As a consequence, the wave equation is an appropriate model problem for analyzing the Euler equations.

B.2 Extension to 2D Single-Species Euler Equations

The factorization matrices for the advective Jacobians introduced in Eqs. (2.113)–(2.114) can be given as follows:

$$\mathbf{L}_1 = \begin{bmatrix} \frac{(\gamma-1)q+au_1}{2a^2} & \frac{(1-\gamma)u_1-a}{2a^2} & \frac{(1-\gamma)u_2}{2a^2} & \frac{(\gamma-1)}{2a^2} \\ \frac{a^2-(\gamma-1)q}{a^2} & \frac{(\gamma-1)u_1}{a^2} & \frac{(\gamma-1)u_2}{a^2} & \frac{(1-\gamma)}{a^2} \\ \frac{(\gamma-1)q-au_1}{2a^2} & \frac{(1-\gamma)u_1+a}{2a^2} & \frac{(1-\gamma)u_2}{2a^2} & \frac{(\gamma-1)}{2a^2} \\ u_2 & 0 & -1 & 0 \end{bmatrix}, \tag{B.18}$$

$$\mathbf{R}_1 = \begin{bmatrix} 1 & 1 & 1 & 0 \\ u_1-a & u_1 & u_1+a & 0 \\ u_2 & u_2 & u_2 & -1 \\ h-au_2 & q & h+au_1 & -u_2 \end{bmatrix}, \tag{B.19}$$

$$\mathbf{L}_2 = \begin{bmatrix} \frac{(\gamma-1)q+au_2}{2a^2} & \frac{(1-\gamma)u_1}{2a^2} & \frac{(1-\gamma)u_2-a}{2a^2} & \frac{(\gamma-1)}{2a^2} \\ \frac{a^2-(\gamma-1)q}{a^2} & \frac{(\gamma-1)u_1}{a^2} & \frac{(\gamma-1)u_2}{a^2} & \frac{(1-\gamma)}{a^2} \\ \frac{(\gamma-1)q-au_2}{2a^2} & \frac{(1-\gamma)u_1}{2a^2} & \frac{(1-\gamma)u_2+a}{2a^2} & \frac{(\gamma-1)}{2a^2} \\ -u_1 & 1 & 0 & 0 \end{bmatrix}, \tag{B.20}$$

$$\mathbf{R}_2 = \begin{bmatrix} 1 & 1 & 1 & 0 \\ u_1 & u_1 & u_1 & 1 \\ u_2-a & u_2 & u_2+a & 0 \\ h-au_2 & q & h+au_2 & u_1 \end{bmatrix}, \tag{B.21}$$

with the eigenvalues located on the main diagonal of the following matrices:

$$\mathbf{\Lambda}_1 = \begin{bmatrix} u_1-a & 0 & 0 & 0 \\ 0 & u_1 & 0 & 0 \\ 0 & 0 & u_1+a & 0 \\ 0 & 0 & 0 & u_1 \end{bmatrix} \tag{B.22}$$

and

$$\mathbf{\Lambda}_2 = \begin{bmatrix} u_2-a & 0 & 0 & 0 \\ 0 & u_2 & 0 & 0 \\ 0 & 0 & u_2+a & 0 \\ 0 & 0 & 0 & u_2 \end{bmatrix}. \tag{B.23}$$

B.3 Extension to 2D Multispecies Euler Equations

The factorization matrices for the advective Jacobians introduced in Eqs. (2.99)–(2.100) are given by matrices (B.26)–(B.29) [72]. The norms appearing in the matrices given by Eqs. (B.28)–(B.29) stand for the standard Euclidean norm. The other terms have been explained in Section 2.2.

The eigenvalues located on the main diagonal of the following matrices [72]:

$$
\mathbf{\Lambda}_1 = \begin{bmatrix}
u_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & u_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & u_1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & u_1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & u_1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & u_1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & u_1 + a & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & u_1 - a & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u_1
\end{bmatrix} , \tag{B.24}
$$

$$
\mathbf{\Lambda}_2 = \begin{bmatrix}
u_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & u_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & u_2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & u_2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & u_2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & u_2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & u_2 + a & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & u_2 - a & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u_2
\end{bmatrix} . \tag{B.25}
$$

$$\mathbf{L}_1 = \begin{bmatrix}
a^2 - Y_O\gamma_O & -Y_O\gamma_N & -Y_O\gamma_{NO} & -Y_O\gamma_{O_2} & -Y_O\gamma_{N_2} & \beta u_1 Y_O & \beta u_2 Y_O & -\beta Y_O & \beta Y_O \\
-Y_N\gamma_O & a^2 - Y_N\gamma_N & -Y_N\gamma_{NO} & -Y_N\gamma_{O_2} & -Y_N\gamma_{N_2} & \beta u_1 \gamma_N & \beta u_2 \gamma_N & -\beta Y_N & \beta Y_N \\
-Y_{NO}\gamma_O & -Y_{NO}\gamma_N & a^2 - Y_{NO}\gamma_{NO} & -Y_{NO}\gamma_{O_2} & -Y_{NO}\gamma_{N_2} & \beta u_1 Y_{NO} & \beta u_2 Y_{NO} & -\beta Y_{NO} & \beta Y_{NO} \\
-Y_{O_2}\gamma_O & -Y_{O_2}\gamma_N & -Y_{O_2}\gamma_{NO} & a^2 - Y_{O_2}\gamma_{O_2} & -Y_{O_2}\gamma_{N_2} & \beta u_1 Y_{O_2} & \beta u_2 Y_{O_2} & -\beta Y_{O_2} & \beta Y_{O_2} \\
-Y_{N_2}\gamma_O & -Y_{N_2}\gamma_N & -Y_{N_2}\gamma_{NO} & -Y_{N_2}\gamma_{O_2} & a^2 - Y_{N_2}\gamma_{N_2} & \beta u_1 Y_{N_2} & \beta u_2 Y_{N_2} & -\beta Y_{N_2} & \beta Y_{N_2} \\
-u_2 & -u_2 & -u_2 & -u_2 & -u_2 & 0 & 1 & 0 & 0 \\
\gamma_O - u_1 a & \gamma_N - u_1 a & \gamma_{NO} - u_1 a & \gamma_{O_2} - u_1 a & \gamma_{N_2} - u_1 a & a - \beta u_1 & -\beta u_2 & \beta & -\beta \\
\gamma_O + u_1 a & \gamma_N + u_1 a & \gamma_{NO} + u_1 a & \gamma_{O_2} + u_1 a & \gamma_{N_2} + u_1 a & -a - \beta u_1 & -\beta u_2 & \beta & -\beta \\
-e_V\gamma_O & -e_V\gamma_N & -e_V\gamma_{NO} & -e_V\gamma_{O_2} & -e_V\gamma_{N_2} & \beta u_1 e_V & \beta u_2 e_V & -\beta e_V & a^2 + \beta e_V
\end{bmatrix} \tag{B.26}$$

172

$$\mathbf{L}_2 = \begin{bmatrix}
a^2 - Y_O\gamma_O & -Y_O\gamma_N & -Y_O\gamma_{NO} & -Y_O\gamma_{O_2} & -Y_O\gamma_{N_2} & \beta u_1 Y_O & \beta u_2 Y_O & -\beta Y_O & \beta Y_O \\
-Y_N\gamma_O & a^2 - Y_N\gamma_N & -Y_N\gamma_{NO} & -Y_N\gamma_{O_2} & -Y_N\gamma_{N_2} & \beta u_1 \gamma_N & \beta u_2 \gamma_N & -\beta Y_N & \beta Y_N \\
-Y_{NO}\gamma_O & -Y_{NO}\gamma_N & a^2 - Y_{NO}\gamma_{NO} & -Y_{NO}\gamma_{O_2} & -Y_{NO}\gamma_{N_2} & \beta u_1 Y_{NO} & \beta u_2 Y_{NO} & -\beta Y_{NO} & \beta Y_{NO} \\
-Y_{O_2}\gamma_O & -Y_{O_2}\gamma_N & -Y_{O_2}\gamma_{NO} & a^2 - Y_{O_2}\gamma_{O_2} & -Y_{O_2}\gamma_{N_2} & \beta u_1 Y_{O_2} & \beta u_2 Y_{O_2} & -\beta Y_{O_2} & \beta Y_{O_2} \\
-Y_{N_2}\gamma_O & -Y_{N_2}\gamma_N & -Y_{N_2}\gamma_{NO} & -Y_{N_2}\gamma_{O_2} & a^2 - Y_{N_2}\gamma_{N_2} & \beta u_1 Y_{N_2} & \beta u_2 Y_{N_2} & -\beta Y_{N_2} & \beta Y_{N_2} \\
u_1 & u_1 & u_1 & u_1 & u_1 & -1 & 0 & 0 & 0 \\
\gamma_O - u_1 a & \gamma_N - u_1 a & \gamma_{NO} - u_1 a & \gamma_{O_2} - u_1 a & \gamma_{N_2} - u_1 a & -\beta u_1 & A - \beta u_2 & \beta & -\beta \\
\gamma_O + u_1 a & \gamma_N + u_1 a & \gamma_{NO} + u_1 a & \gamma_{O_2} + u_1 a & \gamma_{N_2} + u_1 a & -\beta u_1 & -a - \beta u_2 & \beta & -\beta \\
-e_V\gamma_O & -e_V\gamma_N & -e_V\gamma_{NO} & -e_V\gamma_{O_2} & -e_V\gamma_{N_2} & \beta u_1 e_V & \beta u_2 e_V & -\beta e_V & a^2 + \beta e_V
\end{bmatrix} \tag{B.27}$$

$$\mathbf{R}_1 = \begin{bmatrix}
\frac{1}{a^2} & 0 & 0 & 0 & 0 & 0 & \frac{Y_\text{O}}{2a^2} & \frac{Y_\text{O}}{2a^2} & 0 \\
0 & \frac{1}{a^2} & 0 & 0 & 0 & 0 & \frac{Y_\text{N}}{2a^2} & \frac{Y_\text{N}}{2a^2} & 0 \\
0 & 0 & \frac{1}{a^2} & 0 & 0 & 0 & \frac{Y_\text{NO}}{2a^2} & \frac{Y_\text{NO}}{2a^2} & 0 \\
0 & 0 & 0 & \frac{1}{a^2} & 0 & 0 & \frac{Y_{\text{O}_2}}{2a^2} & \frac{Y_{\text{O}_2}}{2a^2} & 0 \\
0 & 0 & 0 & 0 & \frac{1}{a^2} & 0 & \frac{Y_{\text{N}_2}}{2a^2} & \frac{Y_{\text{N}_2}}{2a^2} & 0 \\
\frac{u_1}{a^2} & \frac{u_1}{a^2} & \frac{u_1}{a^2} & \frac{u_1}{a^2} & \frac{u_1}{a^2} & 0 & \frac{u_1+a}{2a^2} & \frac{u_1-a}{2a^2} & 0 \\
\frac{u_2}{a^2} & \frac{u_2}{a^2} & \frac{u_2}{a^2} & \frac{u_2}{a^2} & \frac{u_2}{a^2} & \frac{-1}{a^2} & \frac{u_2}{2a^2} & \frac{u_2}{2a^2} & 0 \\
\frac{\beta\|\mathbf{u}\|^2-\gamma_\text{O}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_\text{N}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_\text{NO}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_{\text{O}_2}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_{\text{N}_2}}{\beta a^2} & \frac{u_2}{a^2} & \frac{h+au_1}{2a^2} & \frac{h-au_1}{2a^2} & \frac{1}{a^2} \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{e_v}{2a^2} & \frac{e_v}{2a^2} & \frac{1}{a^2}
\end{bmatrix}$$
(B.28)

$$\mathbf{R}_2 = \begin{bmatrix}
\frac{1}{a^2} & 0 & 0 & 0 & 0 & 0 & \frac{Y_\text{O}}{2a^2} & \frac{Y_\text{O}}{2a^2} & 0 \\
0 & \frac{1}{a^2} & 0 & 0 & 0 & 0 & \frac{Y_\text{N}}{2a^2} & \frac{Y_\text{N}}{2a^2} & 0 \\
0 & 0 & \frac{1}{a^2} & 0 & 0 & 0 & \frac{Y_\text{NO}}{2a^2} & \frac{Y_\text{NO}}{2a^2} & 0 \\
0 & 0 & 0 & \frac{1}{a^2} & 0 & 0 & \frac{Y_{\text{O}_2}}{2a^2} & \frac{Y_{\text{O}_2}}{2a^2} & 0 \\
0 & 0 & 0 & 0 & \frac{1}{a^2} & 0 & \frac{Y_{\text{N}_2}}{2a^2} & \frac{Y_{\text{N}_2}}{2a^2} & 0 \\
\frac{u_1}{a^2} & \frac{u_1}{a^2} & \frac{u_1}{a^2} & \frac{u_1}{a^2} & \frac{u_1}{a^2} & 0 & \frac{u_1}{2a^2} & \frac{u_1}{2a^2} & 0 \\
\frac{u_2}{a^2} & \frac{u_2}{a^2} & \frac{u_2}{a^2} & \frac{u_2}{a^2} & \frac{u_2}{2a^2} & \frac{-1}{2a^2} & \frac{u_2+a}{2a^2} & \frac{u_2-a}{2a^2} & 0 \\
\frac{\beta\|\mathbf{u}\|^2-\gamma_\text{O}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_\text{N}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_\text{NO}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_{\text{O}_2}}{\beta a^2} & \frac{\beta\|\mathbf{u}\|^2-\gamma_{\text{N}_2}}{\beta a^2} & \frac{u_1}{a^2} & \frac{h+au_2}{2a^2} & \frac{h-au_2}{2a^2} & \frac{1}{a^2} \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{e_v}{2a^2} & \frac{e_v}{2a^2} & \frac{1}{a^2}
\end{bmatrix}$$
(B.29)

APPENDIX C

ON THE EXISTENCE AND UNIQUENESS OF SOLUTIONS OF EULER EQUATIONS

Many desirable characteristics of solving a partial differential equation (PDE) are captured by the concept of a "well-posed problem." If a PDE has a solution, the solution is unique, and it is continuously dependent on the initial (given) data, then the problem governed by this PDE is called *well-posed* [58]. If the uniqueness (of the solution) requirement does not hold, the primary mathematical concern is to classify the solutions. It is preferable to solve a problem in such a way that the well-posedness criteria are satisfied; however, when studying conservation laws, the solutions that are not continuously differentiable or even not continuous must be taken into account as well.

In general, conservation equations do not have classical solutions, but they are well-posed if properly defined generalized or weak solutions are allowed [58]. Consequently, the nature of the problem may force one to abandon searching for smooth solutions, encouraging one to look for broader classes of solution candidates, e.g., weak solutions, instead of achieving well-posedness conditions for classical solutions. To this end, in this section, it is necessary to use the tools of general topology, real analysis, and functional analysis for constructing appropriate function spaces in the search for solutions to conservation equations.

Even today, a solid mathematical understanding of the theory on the existence and uniqueness of solutions of multidimensional nonlinear conservation equations is still very limited [58]. For this reason, the studies on conservation equations are generally carried out in one-dimensional spaces or for spaces equipped with severe constraints.

175

One can refer to [50, 58, 147] for further details.

The nonlinear system of differential equations governing inviscid compressible flows without source terms, studied in Section 2.3, can be given in the following general form [58, 147]:

$$\frac{\partial \mathbf{U}}{\partial t} + \sum_{j=1}^{n_{\text{sd}}} \frac{\partial}{\partial x_j} \mathbf{F}_j\left(\mathbf{U}\right) = \mathbf{0}, \tag{C.1}$$

where the flux vectors $\mathbf{F}_j = \left[F_1^{(j)}, \ldots, F_m^{(j)}\right]^T : \Omega \to \mathbb{R}^m$ consist of continuously differentiable functions, m represents the number of equations involving in system (C.1), and $\Omega \subset \mathbb{R}^m$ is the domain (open set) occupied by a fluid (gas). We consider Eq. (C.1) in a space-time cylinder $Q_T = \Omega \times (0, T)$, where $T > 0$ denotes the final time.

Even when the initial data is sufficiently smooth, certain physical phenomena such as wave breaking, shock waves, and vortices cause the solution of problem (C.1) to develop singularities in finite time [37]. Therefore, the classical analytic techniques used in the theory of other types of PDEs (e.g., parabolic, elliptic, or mixed) cannot be applied to hyperbolic PDEs directly. The nonstrict hyperbolicity of the governing equations is another major issue. For the Euler equations, this degeneracy occurs at the vacuum states or due to the multiple eigenvalues of the system [37].

System (C.1) is said to be a *system of conservation laws,* and can be written in a quasi-linear form as follows:

$$\mathbf{A}_0\left(\mathbf{U}\right) \frac{\partial \mathbf{U}}{\partial t} + \sum_{j=1}^{n_{\text{sd}}} \mathbf{A}_j\left(\mathbf{U}\right) \frac{\partial \mathbf{U}}{\partial x_j} = \mathbf{0}, \tag{C.2}$$

where the Jacobian matrix of flux $\mathbf{F}_j$ is

$$\mathbf{A}_j = \frac{\partial \mathbf{F}_j\left(\mathbf{U}\right)}{\partial \mathbf{U}}, \quad j = 1, 2, \ldots, n_{\text{sd}}, \tag{C.3}$$

and $\mathbf{A}_0\left(\mathbf{U}\right) = \mathbf{I}$ is the unit matrix of order m.

Even the simplest equations having the form of Eq. (C.2) are known to exhibit nonlinear phenomena such as the nonexistence of global smooth solutions on a large set of initial/boundary data [147]. The 1D inviscid Burgers' equations, for example, are one of the best examples of this phenomenon:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0, \quad x \in \Omega = \mathbb{R}, \quad t > 0, \tag{C.4}$$

with initial conditions $u(x, 0) = sx$, $s = \pm 1$. If $s = 1$, the exact solution to Eq. (C.4) is given as $u = \frac{x}{1-t}$, and it is clear that the solution blows up at $t = 1$.

The 3D Euler equations are, of course, the only realistic perspective on space dimension. In many cases, however, a 2D or even a 1D model is sufficient to describe significant aspects of a physical configuration. Besides, in understanding the main features of the Euler equations, some facts from the theory of quasi-linear system of PDEs should be presented. The following lines provide a brief review of the theory on the existence and uniqueness of solutions to conservation equations.

Theorem [147]: Let $\Omega \in \mathbb{R}^m$ be a open set and be the domain of the matrices $\mathbf{A}_j(\mathbf{U})$, $j = 1, \ldots, n_{\text{sd}}$. System (C.2) is symmetrizable in Ω if for any $\mathbf{U} \in \Omega$, there is a positive definite matrix $\widetilde{\mathbf{A}}_0(\mathbf{U})$ such that

(i) for any $D_1 \subset \overline{D}_1 \subset \Omega$,

$$c^{-1}\|\mathbf{z}\|^2 \leq \mathbf{z} \cdot \left(\widetilde{\mathbf{A}}_0(\mathbf{U})\, \mathbf{z}\right) \leq c\|\mathbf{z}\|^2, \quad \widetilde{\mathbf{A}}_0(\mathbf{U}) = \widetilde{\mathbf{A}}_0(\mathbf{U})^T,$$

with a positive constant c independent of $\mathbf{U} \in D_1$, and $\mathbf{z} \in \mathbb{R}^{n_{\text{sd}}}$, where $\|\cdot\|$ represents the standard Euclidean norm, and "$\cdot$" denotes the standard Euclidean inner product;

(ii) for each $j = 1, \ldots, n_{\text{sd}}$, the following statement holds:

$$\widetilde{\mathbf{A}}_0(\mathbf{U})\,\mathbf{A}_j(\mathbf{U}) = \widetilde{\mathbf{A}}_j(\mathbf{U}), \quad \text{where} \quad \widetilde{\mathbf{A}}_j(\mathbf{U}) = \widetilde{\mathbf{A}}_j(\mathbf{U})^T.$$

If a set of conservation laws can be symmetrized, then the theory of linear symmetric hyperbolic systems can be used to study the solutions to equations in the form of Eq. (C.2).

Definition: System (C.2) is said to be *hyperbolic* in a region $\Omega \subset \mathbb{R}^m$ if all solutions (so-called the generalized eigenvalues), $\lambda_j = \lambda_j(\mathbf{U}, \widetilde{\mathbf{n}})$, $j = 1, \ldots, n_{\text{sd}}$, of equation

$$\det\left(\lambda \mathbf{A}_0(\mathbf{U}) - \sum_{j=1}^{n_{\text{sd}}} \widetilde{n}_j \mathbf{A}_j(\mathbf{U})\right) = 0 \tag{C.5}$$

are real for any $\widetilde{\mathbf{n}} \in \mathbb{R}^{n_{\text{sd}}}$ and $\mathbf{U} \in \Omega$. If these eigenvalues are all distinct, then system (C.2) is said to be *strictly hyperbolic*. System (C.2) is called *diagonally and*

strictly hyperbolic if the matrix

$$\mathbb{P} = \sum_{j=1}^{n_{sd}} \widetilde{n}_j \mathbf{A}_j (\mathbf{U}) \tag{C.6}$$

is diagonalizable. That is, there exists a nonsingular matrix $\mathbb{T} (\mathbf{U}, \widetilde{\mathbf{n}})$ such that

$$\mathbb{T}^{-1}\mathbb{P}\mathbb{T} = \text{diag} (\lambda_1, \lambda_2, \ldots, \lambda_m) = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_m \end{bmatrix}.$$

System (C.1), and correspondingly system (C.2), should naturally be equipped with certain initial conditions:

$$\mathbf{U} (\mathbf{x}, 0) = \mathbf{U}_0 (\mathbf{x}), \tag{C.7}$$

where $\mathbf{U}_0$ is a given function. Then, the initial-value (Cauchy) problem can be expressed as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \sum_{j=1}^{n_{sd}} \frac{\partial}{\partial t}\mathbf{F}_j (\mathbf{U}) = \mathbf{0}, \text{ in } Q_T = \Omega \times (0, T), \tag{C.8}$$

$$\mathbf{U} (\mathbf{x}, 0) = \mathbf{U}_0 (\mathbf{x}), \quad \mathbf{x} \in \Omega. \tag{C.9}$$

Definition [147]: Let $\Omega \subset \mathbb{R}^{n_{sd}}$ be the domain set of functions $\mathbf{F}_j$, $j = 1, \ldots, n_{sd}$. The vector $\mathbf{U}$ is said to be the *classical solution* of (the Cauchy problem) Eqs. (C.8)–(C.9) if the followings hold:

(i) $\mathbf{U} \in \mathcal{C}^1 (\Omega \times (0, T))^m \cap \mathcal{C} (\Omega \times [0, T))^m$,

(ii) $\mathbf{U} (\mathbf{x}, t) \in \Omega, \forall (\mathbf{x}, t) \in Q_T$,

(iii) $\mathbf{U} (\mathbf{x}, t)$ satisfies Eqs. (C.8)–(C.9) for all $(\mathbf{x}, t) \in Q_T$ and $\mathbf{x} \in \Omega$, and for $\mathbf{x} \in \Omega$ and $t = 0$.

Here, the space of continuous functions defined on domain Ω is represented by $\mathcal{C} (\Omega)$, or equivalently, by $\mathcal{C}^0 (\Omega)$, and the space of functions whose first derivatives and themselves are continuous is represented by $\mathcal{C}^1 (\Omega)$. It can be generalized as follows: the space $\mathcal{C}^k (\Omega)$ represents the space of functions with continuous zeroth through kth-order derivatives. Then, the following statement holds:

$$\mathcal{C}^\infty (\Omega) = \bigcap_{m=0}^{\infty} \mathcal{C}^m (\Omega). \tag{C.10}$$

Definition: Let $\mathbb{X}$ be a normed space (see, for example, [2]), and A be a subset of $\mathbb{X}$. If every sequence of points in A has a subsequence converging to an element in A, then A is called *compact*. Compact sets are closed and bounded sets, but closed and bounded sets need not be compact if $\mathbb{X}$ is not a finite-dimensional space.

Definition: Let $\Omega \subset \mathbb{R}^{n_{sd}}$ is an open set, and let $f : \Omega \to \mathbb{R}$ be a continuous function. The *support* of f is defined as

$$\mathrm{supp} f = \overline{\{\mathbf{x} \in \Omega \ : \ f(\mathbf{x}) \neq 0\}}. \tag{C.11}$$

The function f is called *compactly supported* if supp f is a compact set.

Note that the subspace $\mathcal{C}_0^\infty (\Omega) \subset \mathcal{C}^\infty (\Omega)$ consists of functions from $\mathcal{C}^\infty (\Omega)$ that have compact support.

Theorem [147]: Let the following linear and scalar Cauchy problem be given

$$\frac{\partial u}{\partial t} + a (x,t) \frac{\partial u}{\partial x} = 0, \ \ x \in \Omega = \mathbb{R}, \ \ t > 0, \tag{C.12}$$

where $u(x,0) = u_0(x) \in \mathcal{C}^1 (\mathbb{R})$, $a(x,t)$ and $\frac{da}{dx}$ belong to $\mathcal{C} \left(\overline{Q_T} \right)$. Then, for any $(x,t) \in Q_T$ there exists a unique classical solution. If $a (x,t)$ is a constant function, the solution (traveling wave) is given as

$$u (x,t) = u_0(x - at). \tag{C.13}$$

Theorem [147]: Let the following 1D linear system of conservation laws be given:

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{U}}{\partial x} = 0 \ \ \text{in} \ \ \mathbb{R} \times (0, \infty), \tag{C.14}$$

equipped with initial conditions $\mathbf{U} (x,0) = \mathbf{U}_0 (x)$. Let $\mathbf{A}$ be an $m \times m$ diagonalizable matrix and has real eigenvalues, i.e., $\lambda_1, \lambda_2, \ldots, \lambda_m \in \mathbb{R}$. Besides, let $\mathbf{U}_0$ can be expanded into the basis formed by $\{\mathbf{r}_j\}$:

$$\mathbf{U}_0 (x,t) = u_j^0(x)\mathbf{r}_j, \ \ \forall x \in \mathbb{R}, \ \ t \in [0, \infty). \tag{C.15}$$

Then, there exists a unique classical solution, $\mathbf{U} \in \mathcal{C}^1 (\mathbb{R} \times (0, \infty))^m$, given as follows:

$$\mathbf{U} (x,t) = \sum_{j=1}^m u_j^0 (x - \lambda_j t) \, \mathbf{r}_j, \tag{C.16}$$

where $\mathbf{r}_j$ is the eigenvector corresponding to eigenvalue λ_j.

Theorem [147]: Let the following nonlinear conservation equation be given

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \ \ x \in \Omega = \mathbb{R}, \ \ t > 0, \tag{C.17}$$

where $u(x, 0) = u_0(x)$. Let $f \in \mathcal{C}^2(\mathbb{R})$ and $u_0(x) \in \mathcal{C}^2(\mathbb{R}) \cup W^{1,\infty}(\mathbb{R})$. Then, there exists a positive number $z \in D$

$$T < \left(\left\| u_0'(x) \right\|_{L^\infty(\mathbb{R})} \sup_{z \in D} \left| f''(z) \right| \right)^{-1}, \tag{C.18}$$

such that there is a unique classical solution to Eq. (C.17), where

$$D = \overline{\{u_0(x) : x \in \mathbb{R}\}}. \tag{C.19}$$

The solution is given as

$$u(x, t) = u_0(x - f'(u)t). \tag{C.20}$$

For further details, one can refer to [21, 71, 117] and the material therein. The definition of the Sobolev space $W^{1,\infty}(\mathbb{R})$ can be deduced from Eq. (C.25).

Definition: A measurable function u is called *essentially bounded* on Ω if there is a constant K such that $|u(x)| \leq K, \ \forall x \in \Omega$. The space $L^\infty(\Omega)$ represents the space of Lebesgue measurable functions defined on Ω that are essentially bounded, where the norm $\| \cdot \|_{L^\infty(\Omega)}$ is defined as

$$\|u\|_{L^\infty(\Omega)} = \operatorname{ess\,sup}_{x \in \Omega} |u(x)|. \tag{C.21}$$

Definition: Let $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ be an n-tuple of nonnegative integers, then α is said to be a *multiindex* of order n. The length of this multiindex is given as follows:

$$|\alpha| = \sum_{j=1}^{n} \alpha_j. \tag{C.22}$$

Then, for a real function $u = u(x_1, x_2, \ldots, x_n)$ with n variables, the αth-order partial derivative of u can be given as follows:

$$D^\alpha u = \frac{\partial^{|\alpha|} u}{\partial x_1^{\alpha_1} \ldots \partial x_n^{\alpha_n}}. \tag{C.23}$$

Definition: A function $\varphi : \Omega \to \mathbb{R}$ belonging the space of infinitely differentiable functions with compact support, $\mathcal{C}_0^\infty(\Omega)$, is called a *test function*.

Definition: A function u defined almost everywhere on Ω is called *locally integrable* on Ω provided $u \in L^1(\Omega)$ for every open subset of Ω, and it is denoted by $u \in L_1^{\text{loc}}(\Omega)$. It can be generalized as follows: $u \in L_p^{\text{loc}}(\Omega)$ if $u|_{\Omega'} \in L^p(\Omega')$ for any bounded $\Omega' \subset \Omega$ such that $\overline{\Omega'} \subset \Omega$, where the term $u|_{\Omega'}$ represents the restriction of function u to the set Ω'.

Definition: Let the functions $u, v \in L_1^{\text{loc}}(\Omega)$, and α be a multiindex. The αth *weak (distributional) derivative* of u, $D^\alpha u$, is given by $D^\alpha u = v$, provided that

$$\int_\Omega u D^\alpha \varphi dx = (-1)^{|\alpha|} \int_\Omega v\varphi dx, \quad \forall \varphi \in \mathcal{C}_0^\infty(\Omega). \tag{C.24}$$

Definition: Let $k \geq 0$ be an integer and $1 \leq p \leq \infty$. Then, the Sobolev space $W^{k,p}(\Omega)$ is the space of all functions $u \in L^p(\Omega)$ such that their weak derivatives up to the order k are also elements of the space $L^p(\Omega)$:

$$W^{k,p}(\Omega) = \{u \in \Omega : D^\alpha u \in L^p(\Omega) \text{ for } 0 \leq |\alpha| \leq k\}. \tag{C.25}$$

The Sobolev space $W^{k,p}$ is equipped with the norm:

$$\|u\|_{W^{k,p}(\Omega)} = \left(\sum_{0 \leq |\alpha| \leq k} \|D^\alpha u\|_{L^p(\Omega)}^p \right)^{1/p}, \quad \text{if} \quad 1 \leq p < \infty, \tag{C.26}$$

$$\|u\|_{W^{k,\infty}(\Omega)} = \max_{0 \leq |\alpha| \leq k} \|D^\alpha u\|_{L^\infty(\Omega)}, \quad \text{if} \quad p = \infty. \tag{C.27}$$

In other words, Sobolev spaces are vector spaces consisting of functions whose weak (distributional) partial derivatives satisfy certain integrability conditions. Notice that if $p = 2$, then

$$W^{k,2}(\Omega) = H^k(\Omega). \tag{C.28}$$

The letter "H" in Eq. (C.28) indicates the space $H^k(\Omega)$ is a Hilbert space. Note that $H^0(\Omega) = L^2(\Omega)$.

Theorem [37]: Let $\Omega \subset \mathbb{R}^{n_{\text{sd}}}$, and $\mathbf{U}_0(\mathbf{x}, t) : \Omega \to G \subset \mathbb{R}^{n_{\text{sd}}+2}$ be in $H^s \cap L^\infty$, where G is the state space, and $s > \frac{n_{\text{sd}}}{2} + 1$. Then, there exists a finite time $T = T\left(\|\mathbf{U}_0\|_{H^s(\Omega)}, \|\mathbf{U}_0\|_{L^\infty(\Omega)}\right) \in (0, \infty)$ for the Cauchy problem given by Eqs. (C.8)–(C.9) such that there is a unique bounded classical solution $\mathbf{U} \in \mathcal{C}^1(\Omega \times [0, \infty])$ with $\mathbf{U}(\mathbf{x}, t) \in G$ for $(\mathbf{x}, t) \in (\Omega \times [0, \infty])$ and $\mathbf{U} \in \mathcal{C}([0, T], H^s) \cap \mathcal{C}^1([0, T], H^{s-1})$.

Note that the governing equations of hypersonic equilibrium flow introduced in Chapter 2 and numerically simulated in Section 5.1 can be recast in the form of Eqs. (C.8)–(C.9). Therefore, this theorem also applies to the governing equations of hypersonic flows in equilibrium.

Majda [132] proved this theorem, which is based solely on the elementary linear existence theory for symmetric hyperbolic systems with smooth coefficients (see [48] by Courant and Hilbert). Kato [101] provided a proof of the theorem, which treats appropriate linearized problems using the abstract semigroup theory of evolution equations. Kato also formulated and applied this basic idea in an abstract framework in [102], resulting in the local existence of smooth solutions for a variety of interesting problems of mathematical physics. Makino et al. [133] used the theory of quasi-linear symmetric hyperbolic systems to apply the local existence theory to the Cauchy problem with compactly supported initial data for multidimensional Euler equations. For further discussion, one can refer to [49] by Crandall and Souganidis.

The authors of [37] considered conservation equations having the form of Eqs. (C.8)–(C.9) with a source term (vector) $\mathbf{S}(\mathbf{U})$:

$$\frac{\partial \mathbf{U}}{\partial t} + \sum_{j=1}^{n_{\mathrm{sd}}} \frac{\partial}{\partial x_j} \mathbf{F}_j(\mathbf{U}) = \mathbf{S}(\mathbf{U}), \ \ \text{in} \ \ Q_T = \Omega \times (0, T), \tag{C.29}$$

$$\mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0(\mathbf{x}), \ \ \mathbf{x} \in \Omega. \tag{C.30}$$

The source term $\mathbf{S}(\mathbf{U})$ generally arises from the chemical reactions that occur in the flow field or due to the kinetic relaxation scheme employed for solving the system under consideration. For more on chemical reaction terms that might enter the system, one can refer to Section 2.1.4.

Notice that the governing equations of hypersonic nonequilibrium flow introduced in Chapter 2 and numerically simulated in Section 5.2 can be recast in the form of Eqs. C.29–C.30. Therefore, the theory also applies to multispecies nonequilibrium flow equations that are the focus of this thesis.

The fact that there are generally no globally defined smooth solutions, even for simple quasilinear scalar conservation equations, raises the need to define weaker formulations for systems having the form of Eqs. (C.8)–(C.9).

Definition: Let $\mathbf{U}_0 \in [L_\infty^{\mathrm{loc}}(\Omega)]^m$, where $\Omega \subset \mathbb{R}^{n_{\mathrm{sd}}}$. The vector $\mathbf{U}$ is called a *weak (integral) solution* of Eqs. (C.8)–(C.9) if $\mathbf{U} \in [L_\infty^{\mathrm{loc}}(\Omega \times [0,\infty))]^m$ and the following statement holds [147]:

$$
\int_0^\infty \int_\Omega \left(\mathbf{U} \cdot \frac{\partial \boldsymbol{\varphi}}{\partial t} + \sum_{j=1}^{n_{\mathrm{sd}}} \mathbf{F}_j(\mathbf{U}) \cdot \frac{\partial \boldsymbol{\varphi}}{\partial x_j} \right) \, d\mathbf{x} \, dt
$$
$$
+ \int_\Omega \mathbf{U}_0(\mathbf{x}) \cdot \boldsymbol{\varphi}(\mathbf{x}, 0) \, d\mathbf{x} = 0, \quad \text{(C.31)}
$$

$\forall \boldsymbol{\varphi} \in [\mathcal{C}_0^\infty(\Omega \times [0,\infty))]^m$. This definition can also be given for Eqs. (C.29)–(C.30) in a similar way.

Any classical solution of Eqs. (C.8)–(C.9) is also a weak solution (see [147]). However, since the weak solutions of conservation equations do not have to be unique, the concept of weak solutions is still insufficient alone; therefore, certain additional conditions, e.g., the entropy condition, are needed to select an appropriate weak solution. The entropy condition is based on the second law of thermodynamics, which implies that the entropy of individual fluid particles cannot decrease as time goes. One can find further discussion on entropy conditions in [50, 58, 147] and the material therein.

This appendix has aimed to present a general and concise discussion on the existence and uniqueness of the solutions of the conservation equations, which have been considered computationally in the thesis. For many more theoretical concepts on the properties of solutions, the interested reader can exploit the references provided in the above lines. Besides that, tor more on the special function spaces and norms briefly introduced in this section, the interested reader may refer to [2, 58].

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Cengizci, Süleyman

Nationality: Turkish

Date and Place of Birth: 04.07.1989, Antalya / Turkey

E-mail: suleymancengizci@gmail.com; suleyman.cengizci@antalya.edu.tr

ORCID: 0000-0002-4345-1253

EDUCATION

Degree	Institution	Year of Graduation
Ph.D.	Institute of Applied Mathematics, METU	2022
M.S.	Dept. of Mathematics, Nevşehir University	2014
B.S.	Dept. of Mathematics, Niğde University	2012

PROFESSIONAL EXPERIENCE

Year	Place	Enrollment
2017–Onwards	Comp. Prog., Antalya Bilim University	Lecturer
2019–2021	Antalya Y. Z. Ö. High School of Science	IB Math. Teacher
2014–2017	Dept. of Econ., Antalya Bilim University	Research Assistant

PUBLICATIONS

Papers Submitted to International Journals

1. S. Cengizci, Ö. Uğur, S. Natesan, *SUPG-YZβ computation of chemically reactive convection-dominated nonlinear models*, Mathematics and Computers in Simulation, 2021.

2. S. Cengizci, Ö. Uğur, S. Natesan, *SUPG-YZβ formulation for solving convection-dominated steady linear reaction-convection-diffusion equations*, Mathematical Methods in the Applied Sciences, 2021.

3. S. Cengizci, Ö. Uğur, *A stabilized FEM formulation with discontinuity-capturing for solving Burgers'-type equations at high Reynolds numbers*, Applied Mathematics and Computation, 2021.

Papers in Progress

1. S. Cengizci, Ö. Uğur, K. Takizawa, T. E. Tezduyar, *Stabilized finite element computation of inviscid hypersonic flow around a cylinder*, Journal of Mechanics, Oxford, 2022.

2. S. Cengizci, Ö. Uğur, T. E. Tezduyar, K. Takizawa, *SUPG computation of hypersonic flows in thermochemical nonequilibrium with YZβ shock-capturing*, 2022.

3. S. Cengizci, Ö. Uğur, T. E. Tezduyar, *SUPG finite element formulation of shallow-water equations with shock-capturing*, 2022.

Abstracts in International Conference Publications

1. S. Cengizci, Ö. Uğur, S. Natesan, *A SUPG formulation for solving a class of singularly perturbed steady problems in 2D*, The 20th Biennial Computational Techniques and Applications Conference (CTAC2020), 30th Aug–2nd Sep 2020, Sydney, NSW, Australia.

186

2. S. Cengizci, Ö. Uğur, K. Takizawa, T. E. Tezduyar, *A Streamline-Upwind/P etrov−Galerkin formulation for supersonic and hypersonic flow simulations*, The 20th Biennial Computational Techniques and Applications Conference (CTAC2020), 30th Aug–2nd Sep 2020, Sydney, NSW, Australia.

3. S. Cengizci, Ö. Uğur, S. Natesan, *A stabilized finite element formulation for numerical simulation of convection-dominated reactive models*, International Conference on Advances in Differential Equations and Numerical Analysis (ADENA2020), October 12–15, 2020, Indian Institute of Technology Guwahati, Guwahati, India.

4. S. Cengizci, Ö. Uğur, T. E. Tezduyar, *Stabilized numerical simulations of hypersonic flows in thermochemical nonequilibrium with FEniCS*, FEniCS 2021, 21–26 March 2021, University of Cambridge, UK.

5. S. Cengizci, Ö. Uğur, S. Natesan, *Stabilized finite element simulations for Burgers'-type equations*, International Conference on Analysis and Its Applications (ICAA 2021), 9–11 April 2021, Kathmandu University, Dhulikhel, Nepal.

6. S. Cengizci, Ö. Uğur, T. E. Tezduyar, *SUPG-stabilized finite element formulation of shallow-water equations*, International Conference of Young Mathematicians, 3–5 June 2021, Institute of Mathematics of NAS of Ukraine, Kyiv, Ukraine.

VISITING INSTITUTIONS

1. Erasmus+ Visiting Staff, The Interdisciplinary Center for Scientific Computing (IWR), Ruprecht-Karls-Universität Heidelberg, Germany, 7–12 May, 2017, hosted by Prof. Dr. Anna Marciniak-Czochra.