Politecnico di Torino

PURDUE UNIVERSITY

Politecnico di Torino

Master's Degree Program in Aerospace Engineering
Academic Year 2023/2024
Graduation Session October 2024

HYQ1D
(Hypersonic Quasi-1-Dimensional): A Numerical Simulation Tool for Hypersonic Internal Flow Characterization

Advisors:
Prof. Domenic D'Ambrosio
Prof. Carlo Scalo

Candidate:
Luigi Renzulli

Abstract

HYQ1D (Hypersonic Quasi-1-Dimensional): A Numerical Simulation Tool for Hypersonic Internal Flow Characterization is a research code developed to investigate and catalog hypersonic wind tunnels, with a strong emphasis on modeling the free piston-driven mechanism and thermochemical effects.

It incorporates various physical models, ranging from ideal gas approximations to thermal and chemical non-equilibrium physics. Different numerical methods are presented for time integration and spatial discretization, spanning from first-order to high-order formulations, as well as several approximated Riemann solvers. The code can account for chemical and thermodynamic non-equilibrium through its internal formulation or by relying on PLATO thermochemical libraries [1].

The potential of the HYQ1D code lies in its modularity; the computational domain can be divided into different sections, with each section utilizing different numerical methods. All sections are linked through boundary conditions, advancing together in time, making this code an efficient tool for simulating wind tunnel facilities, which typically consist of various sections, each with distinct specific characteristics.

In this work, the HYQ1D code and the free-piston model are extensively tested to verify their capabilities in matching the physics of free-piston-driven hypersonic wind tunnels.

Acknowledgements

Friedrich Nietzsche: "He who has a why to live can bear almost any how."
I thank those who have been my "why" over the years.

Steve Jobs: "Stay hungry, stay foolish."
I thank myself for always having the necessary hunger and madness, hoping that this is just the starting point.

Friedrich Nietzsche: "Chi ha un perché per vivere può sopportare quasi ogni come."
Ringrazio chi in questi anni è stato il mio "perchè".

Steave Jobs: "Siate affamati, siate folli"
Ringrazio me stesso per aver sempre avuto la fame e la follia necessaria, augurandomi che questo sia solo un punto di partenza.

Table of Contents

List of Tables

List of Figures

Nomenclature

$\alpha_{s,r}$	Stoichiometric coefficient for reactants in the r-th reaction
$\beta_{s,r}$	Stoichiometric coefficient for products in the r-th reaction
Δx	Generic cell width
Δx_{P_L}	Left sub-cell width
Δx_{P_R}	Right sub-cell width
$\mathbf{I}$	Identity matrix
$\mathbf{v}$	Fluid's velocity vector $\frac{\mathrm{m}}{\mathrm{s}}$
μ	Friction coefficient
ν_s	Vibrational characteristic frequency of the s-th species
ρ	Fluid's density $\frac{\mathrm{Kg}}{\mathrm{m}^3}$
ρ_s	Partial density of the s-th species $\frac{\mathrm{Kg}}{\mathrm{m}^3}$
Θ_{vs}	Characteristic vibrational temperature of the s-th species (K), $\Theta_{vs} = \frac{h\nu_s}{k}$
A_P	Piston face area
a_p	Piston acceleration
c_v	Specific heat at constant volume of the mixture $\frac{\mathrm{J}}{\mathrm{KgK}}$
$C_{b,r}$	Backward pre-exponential parameter in Dunn and Kang model
$C_{f,r}$	Forward pre-exponential parameter in Dunn and Kang model
c_{v_s}	Specific heat at constant volume of the s-th species $\frac{\mathrm{J}}{\mathrm{KgK}}$
$E_{b,r}$	Backward activation energy in Dunn and Kang model

$E_{f,r}$ Forward activation energy in Dunn and Kang model

$F_{\perp}$ Force acting perpendicular to the surface

F_{bra} Brakes force

F_{fri} Friction force

F_{pres} Pressure force

fv Fluid-dynamic variables vector

k Boltzmann constant

$k_{b,r}$ Backward reaction rate coefficient in the r-th reaction (cgs)

$k_{f,r}$ Forward reaction rate coefficient in the r-th reaction (cgs)

m_p Piston mass

MM Mixture molar mass $\frac{\text{g}}{\text{mol}}$

MM_s Molar mass of the s-th species $\frac{\text{g}}{\text{mol}}$

N_s Number of chemical reacting species

N_v Number of species with vibrational degree of freedom

$n_{b,r}$ Backward exponential parameter in Dunn and Kang model

$n_{f,r}$ Forward exponential parameter in Dunn and Kang model

Nr Number of reactions

p Fluid's pressure Pa

p_{P_L} Pressure at the left face of the piston

p_{P_R} Pressure at the right face of the piston

R_s^* Specific gas constant for s-th species, $R_s^* = \frac{R}{MM_s}$

$R_{b,r}$ Backward reaction rate in the r-th reaction $\frac{\text{kg mol}}{\text{m}^3\text{s}}$

$R_{f,r}$ Forward reaction rate in the r-th reaction $\frac{\text{kg mol}}{\text{m}^3\text{s}}$

T_v Vibrational temperature of the mixture K

T_{rot} Rotational temperature K

T_{tr}	Translational temperature K
T_{v_s}	Vibrational temperature linked to the s-th species K
Tq	Rate-controlling temperature
U	Conservative variables vector
u	X-coordinate of the fluid's velocity vector $\frac{\mathrm{m}}{\mathrm{s}}$
X_s	Molar fraction of the s-th species
Y_s	Mass fraction of the s-th species
${x_P}^*$	Piston x-coordinate at * instant

Introduction

This work aims to develop a tool, HYQ1D (Hypersonic Quasi-1-Dimensional), for performing fluid dynamic simulations in the hypersonic flow regime. Unlike subsonic and supersonic regimes, hypersonic flow lacks a strict definition in terms of Mach number. However, we can consider a flow to be hypersonic when certain distinctive physical phenomena begin to occur with significant frequency.

In hypersonic flows, the kinetic energy of the fluid is substantially higher than in other regimes due to the elevated Mach numbers. When the fluid encounters an obstacle, this kinetic energy is converted into internal energy, leading to extremely high temperatures. These temperatures, in turn, trigger significant chemical reaction rates between the fluid species. Another notable aspect is the excitation of additional energetic degrees of freedom, which, at normal temperatures, would contain negligible amounts of energy. These two aspects are closely correlated, each influencing the other [2, 3].

Given the exotic phenomena described in the general hypersonic regime, great attention must be paid to the physical and computational models. Since this flow regime is not very common, commercial codes typically do not account effectively for all these phenomena, leading to the development of research codes for specific scenarios like this one.

This type of flow is typical, for example, in hypersonic wind tunnels. Therefore, the quasi-one-dimensional code was designed specifically to further investigate and analyze these facilities. A strong emphasis was placed on free-driven piston wind tunnels, for which a specific free-piston model was developed. Various research codes exist to simulate these facilities partially or completely in Lagrangian form [4, 5, 6], but this work focuses on a tool capable of capturing all hypersonic phenomena, along with the free-piston effects, within an Eulerian framework that can simulate the entire wind tunnel.

The wind tunnels primarily used for designing and testing the code were the T4 wind tunnel at Queensland University [7] and the DLR-HEG tunnel [8, 9]. However, numerous other facilities use free-piston drivers to compress gas, such as JAXA-HIEST [10], Caltech-T5 [11], Purdue-HYPULSE [12], and others.

Chapter 1

Hypersonic flow regime

To start the description of the HYQ1D development, an initial overview of the main physical aspects of hypersonic flow, such as chemical and thermal non-equilibrium, is provided.

1.1 Chemical reacting flow

1.1.1 Damköhler number

When dealing with chemically reacting flows, the Damköhler number [13, 14] is a valuable tool for quantifying how quickly a gas reaches its equilibrium composition at a given temperature. It is defined as:

$$Dm = \frac{\tau f}{\tau c}$$

where τ_f represents the characteristic time of the fluid, which can be approximated by $\frac{L}{V_\infty}$, with L as the reference length and V_∞ the free-stream velocity.

τ_c represents the characteristic time for chemistry and is taken as the fastest characteristic time among all reactions in the chosen chemical model.

This definition of the Damköhler number describes whether we are closer to a situation of *chemical equilibrium* where reactions occur instantaneously (i.e., when $\tau_c \ll \tau_f$) or a *frozen equilibrium* case, where chemical reactions are essentially halted because they don't have time to occur (i.e., when $\tau_c \gg \tau_f$). In between lies the non-equilibrium case, where the rates of chemical reactions and fluid convection dynamics are of the same order of magnitude, meaning no approximations can be made.

In this work, the focus will be on non-equilibrium chemistry, as the other cases are special instances of this more general condition. Thus, the implementation

will follow the most appropriate approach for hypersonic simulations under various conditions.

1.1.2 Chemical non-equilibrium model

As previously mentioned, in a fluid mixture, when the temperature becomes sufficiently high, as in hypersonic applications, the chemical reaction rates become non-negligible. To understand this, we need to consider a single chemical reaction, which tends toward equilibrium between the forward and backward rates at a specific temperature. This temperature acts as the key factor in determining whether the reactants or products are more stable under given conditions. In non-equilibrium flows, a sudden temperature change shifts this equilibrium between reactants and products, but this shift is not instantaneous; there is a time-dependent change in the concentration of the species involved.

To predict this mechanism, it is necessary to evaluate the time evolution of species concentrations by modeling the *mass production terms*, $\omega_s = \frac{\partial \rho_i}{\partial t}|_{\text{ch}}$, for each species. An independent conservation equation (Equation 1.1) must be solved to calculate the amount of the s-th species at each step within the domain under consideration.

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \mathbf{v}) = \dot{\omega}_s \quad s \in [1, .., N_s] \tag{1.1}$$

Following the model shown in Refs. [15], a similar approach will be presented, although other models are also equivalent [3, 16]. To evaluate this term, each contribution must account for both the production and destruction of the s-th species in all r-th reactions, weighted by the stoichiometric coefficient of the species under consideration, as shown in Equations 1.2. The total reaction rates, $R_{f,r}$ and $R_{b,r}$, are calculated based on the species characteristics, concentrations, stoichiometric coefficients, and the reaction rate coefficients, $k_{f,r}$ or $k_{b,r}$. These coefficients are modeled using an Arrhenius-type law, which is driven by the so-called *rate-controlling temperature*. In the case of thermodynamic equilibrium, this temperature coincides with the unique temperature of the system, whereas in thermodynamic non-equilibrium, it becomes a function of the various temperatures considered in the model.

$$
\begin{aligned}
\dot{\omega}_s &= M_s \sum_{r=1}^{N_r} (\beta_{s,r} - \alpha_{s,r})(R_{f,r} - R_{b,r}) \\
R_{f,r} &= 1000 \left[k_{f,r} \prod_{s=1}^{N_s} \left(0.001 \frac{\rho_s}{M_s} \right)^{\alpha_{s,r}} \right] \\
R_{b,r} &= 1000 \left[k_{b,r} \prod_{s=1}^{N_s} \left(0.001 \frac{\rho_s}{M_s} \right)^{\beta_{s,r}} \right] \\
k_{f,r} &= C_{f,r} T_q^{n_{f,r}} \exp\left(\frac{-E_{f,r}}{k T_q} \right) \\
k_{b,r} &= C_{b,r} T_q^{n_{b,r}} \exp\left(\frac{-E_{b,r}}{k T_q} \right)
\end{aligned}
\tag{1.2}
$$

To complete the chemical model formulation, it is necessary to choose an appropriate kinetic model. This involves selecting the list of species and reactions that are most relevant for the application under consideration. It is important to note that in reality, a vast number of reactions occur between the selected species, but under the given conditions, the reaction rates of many of these are negligible. In this work, the considered species are [N_2, O_2, NO, N, O, He, Ar], as they are the most significant species observed and formed in hypersonic wind tunnel tests.

Regarding He and Ar, these noble gases do not chemically react with other gases; they can only collide with them. However, as explained later in this work, the inviscid hypothesis is considered. Therefore, if He and Ar are initially separated from the other gases, they will remain separated throughout the integration time, and thus they are not included in the kinetic model.

The kinetic model selected is partially based on the Dunn and Kang kinetic model, as reported in Table 1.1, where all the necessary constants for Equations 1.2 are listed.

r	Reaction	$C_{f,r}$	$n_{f,r}$	$\frac{E_{f,r}}{k}$	$C_{b,r}$	$n_{b,r}$	$\frac{E_{b,r}}{k}$
1	O_2 + M ↔ 2O + M (M = N, NO)	3.60E+18	-1.00	5.950E+04	3.00E+15	-0.50	0.000E+00
2	O_2 + O ↔ 2O + O	9.00E+19	-1.00	5.950E+04	7.50E+16	-0.50	0.000E+00
3	O_2 + O_2 ↔ 2O + O_2	3.24E+19	-1.00	5.950E+04	2.70E+16	-0.50	0.000E+00
4	O_2 + N_2 ↔ 2O + N_2	7.20E+18	-1.00	5.950E+04	6.00E+15	-0.50	0.000E+00
5	N_2 + M ↔ 2N + M (M = O, NO, O_2)	1.90E+17	-0.50	1.130E+05	1.10E+16	-0.50	0.000E+00
6	N_2 + N ↔ 2N + N	4.08E+22	-1.50	1.130E+05	2.27E+21	-1.50	0.000E+00
7	N_2 + N_2 ↔ 2N + N_2	4.70E+17	-0.50	1.130E+05	2.72E+16	-0.50	0.000E+00
8	NO + M ↔ N + O + M (M = O_2, N_2)	3.90E+20	-1.50	7.550E+04	1.00E+20	-1.50	0.000E+00
9	NO + M ↔ N + O + M (M = O, N, NO)	7.80E+21	-1.50	7.550E+04	2.00E+21	-1.50	0.000E+00
10	NO + O ↔ O_2 + N	3.20E+09	1.00	1.970E+04	1.30E+10	1.00	3.580E+03
11	N_2 + O ↔ NO + N	7.00E+13	0.00	3.800E+04	1.56E+13	0.00	0.000E+00

Table 1.1: Part of the Dunn and Kang kinetic model [15]

1.2 Thermodynamical properties of high enthalpy gas

As explained earlier, in highly energetic flows, some of the energy within a particle can be stored in various energetic degrees of freedom apart from the translational and rotational ones, which are typically activated at low temperatures. As the temperature and thus the internal energy of the particle increase, other degrees of freedom become excited. Specifically, vibrational and electronic degrees of freedom become energetically significant as the temperature rises [17].

Classical thermodynamics does not accurately predict the energy contribution of these degrees of freedom because it assumes continuous energy levels for each degree of freedom. However, quantum mechanics has demonstrated that energy levels are discrete, and transitioning between them requires a specific quantum of energy. By incorporating the probabilistic nature of these degrees of freedom, simple relationships can be derived to model these energy levels [18, 19].

With this in mind, we can observe in equation 1.3 that the internal energy of a species includes a term for each energetic degree of freedom we aim to model.

$$e_s = e_s^{tr} + e_s^{rot} + e_s^{vib} + e_s^{el} + e_s^0 \tag{1.3}$$

where e_s^0 is the internal energy at 0 K, representing the zero-point energy for each degree of freedom.

Evaluating this energy can be challenging; however, understanding that in thermofluid dynamic conservation laws we are generally concerned with the variation of energy, a common convention is to consider the enthalpy of formation of a species at zero-point instead of the internal energy [20]. This approach is valid because:

$$e_p^0 - e_r^0 = (\Delta h_f)_p^0 - (\Delta h_f)_r^0 \tag{1.4}$$

The variation in zero-point energy during a reaction is equal to the variation in the heat of formation at absolute zero. Unlike the internal energy at 0K, this value can be more easily calculated as it represents the amount of energy required to bring the particle to that state at 0 K, and it is usually tabulated.

As regards the other energy contributions, from quantum mechanics derivations, we have the following equilibrium relations [21, 3]:

- $e_s^{tr} = \frac{3}{2} R_s^* T_{tr} \left[\frac{J}{Kg}\right]$ energy associated with the translational degree of freedom of the particle
- $e_s^{rot} = \frac{L}{2} R_s^* T_{tr} \left[\frac{J}{Kg}\right]$ energy associated with the rotational degree of freedom of the particle.

- $e_s^v = \frac{R_s^* \Theta_{vs}}{e^{\frac{\Theta_{vs}}{T_{vs}}} - 1} \left[\frac{J}{Kg}\right]$ energy associated with the vibrational degree of freedom of the particle, modeled using the harmonic oscillator assumption to describe the vibrational degree of freedom [18, 22]
- e_s^{el} this is not considered in this work, but formulations are available from quantum mechanics considerations [18]

Where L is the rotational number of degrees of freedom of the s-th species.

1.2.1 Thermodynamical non-equilibrium flow

A gas can be defined as being in thermal equilibrium if its energetic degrees of freedom can be approximated as being in equilibrium, meaning $T_{tr} = T_{rot} = T_v = T_{el}$ [15]. In our case, we aim to establish a physical model capable of describing hypersonic flows, where rapid temperature changes with very steep gradients are present, and the maximum values are sufficient to certainly excite the vibrational degrees of freedom. Given these characteristics of the flow under consideration, a more general non-equilibrium approach was adopted. However, two assumptions are made:

- The translational and rotational energy degrees of freedom were considered fully excited, consistent with the chosen application. This allows us to use the previously mentioned relations and assume $T_{tr} = T_{rot}$ (hereafter referred to as T_{tr})
- As mentioned earlier, electronic energy is not considered. The electronic degree of freedom, which is significantly excited at much higher temperatures than the other energy degrees of freedom, could play an important role in some extreme applications. In such cases, careful attention must be given to the reliability of the model described here.

In non-equilibrium flows, an additional conservation equation must be included in the model for each energy mode assumed to be out of equilibrium. In this case, one equation was considered for each species with vibrational degrees of freedom.

In these equations the left hand side account for the convection of the vibrational energy of the s-th species and the source term arising from the so-called vibrational relaxation.

When a sudden change in internal energy occurs, the rotational-translational and vibrational degrees of freedom are excited differently, resulting in non-equilibrium conditions. The particles in the system gradually exchange energy between the degrees of freedom through collisions, driving the system toward equilibrium.

This process is modeled by the previously mentioned relaxation term on the right-hand side of Equation 1.5.

$$\frac{\partial(\rho e_s^v)}{\partial t} + \nabla \cdot (\rho e_s^v \mathbf{v}) = \rho \frac{e_s^{v,eq} - e_s^v}{\tau_s^{V-T}}, \quad s = 1, N_v \tag{1.5}$$

Analyzing the relaxation term, we observe that the driving force is precisely the difference between the vibrational energy and its value under equilibrium conditions (calculated using the previously shown relation with $T = T_{tr}$), scaled by a relaxation time that indicates how quickly the vibrational mode of the s-th species relaxes.

Regarding the vibrational relaxation time, it is just one of the possible relaxation processes between different energy degrees of freedom, but it is the only one considered in this work. It models the energy exchange through particle collisions within the vibrational degree of freedom [23, 24].

Equation 1.6 shows how the relaxation time for a single species is derived from the relaxation times corresponding to the interactions of that particular species.

$$\tau_s^{\text{V-T}} = \frac{1}{\mathcal{M} \sum_{j=1}^{N_s} \frac{q_j}{\tau_{s,j}^{\text{V-T}}}}. \tag{1.6}$$

where the relaxation time $\tau_{i,j}$ is modeled according to [25, 26], where it is divided into two contributions. The first models the relaxation time, and the second is a correction for high-temperature conditions, which extend the time for elastic collisions to occur, avoiding underestimation of $\tau_{i,j}$.

Finally, in our model, regarding energy conservation, we have one equation that conserves the total energy of the system (internal + kinetic energy) and N_v equations for vibrational energy conservation, shown in Equation 1.5. The total energy is represented as:

$$\begin{aligned} E = \rho e &= \rho \left\{ \sum_{s=1}^{N_s} Y_s \left(e_s^{tr} + e_s^{rot} + e_s^{vib} + e_s^{el} + (\Delta h_f)_s^0 \right) \right\} + \rho \frac{1}{2}(u^2 + v^2 + w^2) \\ &= \rho \left\{ \sum_{s=1}^{N_s} Y_s \left(\frac{3}{2} R_s T + \frac{L_s^{rot}}{2} R_s T + e_s^{vib} + e_s^{el} + (\Delta h_f)_s^0 \right) \right\} + \frac{1}{2} \left[\frac{(\rho u)^2 + (\rho v)^2 + (\rho w)^2}{\rho} \right] \end{aligned} \tag{1.7}$$

Chapter 2

Hypersonic wind tunnels

Hypersonic and supersonic aerodynamic wind tunnels are composed of several components, each designed to accelerate the flow from an initial quiescent state to the desired conditions in the test section. Different hypersonic wind tunnels exist, each conceived to study specific physical phenomena or to offer particular operational and experimental advantages. Here, an overview of some of the principal hypersonic wind tunnel architectures is provided [27].

- **Free-Piston Driven Wind Tunnels**: A high-pressure reservoir acting on a free-moving piston compresses the driver gas, creating a very high-enthalpy condition that feeds the shock tube and, subsequently, the nozzle [28].
- **Arc-Heated Wind Tunnels**: The gas is heated and dissociated by an electric arc, and the resulting gas is then expanded in a nozzle [29].
- **Detonation-Driven Wind Tunnels**: The process starts with a detonation that creates the high-enthalpy condition, which feeds a shock tube and then a nozzle in succession [30].
- **Blowdown Wind Tunnels**: Driven by the pressure differential created between a high-pressure condition upstream and a very low-pressure condition downstream [31].

In general, a hypersonic tunnel consists of several preliminary steps, listed above, to reach very high enthalpy conditions in the gas, which is then expanded in a nozzle.

Regarding Free-Piston Wind Tunnels, on which this work focuses, they are divided into three main components. The first is the compression tube, which serves to pressurize the flow. The second part is the shock tube, where the high-pressure flow from the compression stage interacts with a low-pressure region,

generating a shock wave that travels along the tube. This shock wave compresses and accelerates the gas toward the nozzle.

The nozzle is the third fundamental part. The gas, arriving with very high energy in terms of temperature and pressure, is expanded in the nozzle to reach the desired Mach number in the test section, which is located at the nozzle's exit.

To further investigate and catalog this type of facility, HYQ1D, a quasi-one-dimensional finite volume code, was developed, with a strong emphasis on the modeling of free-piston driven wind tunnels. The code was tested with several test cases from some of the main Free-Piston Wind Tunnels, as detailed in sections 9.3 and 10, using QU-T4 and DLR-HEG test cases.

Chapter 3

HYQ1D: Hypersonic Quasi-1-Dimensional

3.1 Introduction

The HYQ1D (Hypersonic Quasi-1-Dimensional) is a research code designed to simulate internal hypersonic aerodynamics. One of its closest applications is simulating the internal flow in hypersonic wind tunnels, characterizing all components of the facility. To better accommodate this purpose, the code was designed to be modular. In the discretization of the computational domain, different sections can be defined, and for each section, the user can choose different types of spatial or temporal discretization or flux evaluation methods. Each section communicates with others through boundary conditions, which the user can decide to either isolate or permit communication between sections. Additionally, there is the option to integrate a free piston via a *free piston model* developed in this work (Chapter 5) or to select a diaphragm connection between sections, which bursts at a specific pressure.

3.2 Physical Model and Discretization

3.2.1 Equation Derivation

In this section, the physical model and its discretization in the software are presented. As mentioned in Chapter 1, viscosity is generally neglected in this work, but certain effects are considered through the source term in the governing equations as showed in Section 3.2.2.

The model development begins with the Euler equations for 3D inviscid fluids in their differential form, as shown in Equation 3.1 [32].

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \\
&\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + p\mathbf{I}) = 0 \\
&\frac{\partial E}{\partial t} + \nabla \cdot [\mathbf{v}(E + p)] = 0
\end{aligned} \tag{3.1}$$

With these equations in mind, we now consider that we are dealing with a multicomponent flow. Therefore, there is a need for a mass conservation equation for each gas present in the flow. Additionally, since the flow is chemically reactive, a chemical source term must be included. This leads us to use Equation 1.1, presented in Section 1.1.2.

Another important aspect to consider is the thermodynamic framework outlined in Section 1.2.1, where Equation 1.5 introduces an additional set of equations to account for the conservation of vibrational energy for each species with vibrational degrees of freedom.

$$\begin{aligned}
&\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \mathbf{v}) = \dot{\omega}_s \quad s \in [1, .., N_s] \\
&\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + p\mathbf{I}) = 0 \\
&\frac{\partial E}{\partial t} + \nabla \cdot [\mathbf{v}(E + p)] = 0 \\
&\frac{\partial (\rho e_s^v)}{\partial t} + \nabla \cdot (\rho e_s^v \mathbf{v}) = \rho \frac{e_s^{v,eq} - e_s^v}{\tau_s^{V-T}}, \quad s \in [1, .., N_v]
\end{aligned} \tag{3.2}$$

The following equation of state and definitions are included to close the system:

- Partial density definition: $\rho_s = Y_s \rho$; $\rho = \sum_s \rho_s$.
- Ideal gas law: $p = \rho R T$; $p_s = \frac{\rho_s R T}{M M_s}$.
- Internal energy definition in Equation 1.4 and total energy definition in Equation 1.7.

At this stage, it is important to understand which type of discretization will be used. When dealing with finite differences, the starting point for discretization is the differential form of the equations, whereas for finite volume discretization, the integral form of the equations is required. Considering that the code developed in this work implements a finite volume discretization, we integrate the equations 3.2 in the volume and, by applying Gauss' theorem, we obtain the integral form as shown in Equations 3.3.

$$
\begin{aligned}
&\frac{\partial}{\partial t}\int_\Omega \rho_s \, dV + \int_{\partial\Omega} (\rho_s \mathbf{v}) \cdot \mathbf{n} \, dS = \int_\Omega \dot{\omega}_s \, dV \quad s \in [1, .., N_s] \\
&\frac{\partial}{\partial t}\int_\Omega \rho \mathbf{v} \, dV + \int_{\partial\Omega} (\rho \mathbf{v}\mathbf{v} + p\mathbf{I}) \cdot \mathbf{n} \, dS = 0 \\
&\frac{\partial}{\partial t}\int_\Omega E \, dV + \int_{\partial\Omega} [\mathbf{v}(\rho E + p)] \cdot \mathbf{n} \, dS = 0 \\
&\frac{\partial}{\partial t}\int_\Omega \rho e_s^v \, dV + \int_{\partial\Omega} (\rho e_s^v \mathbf{v}) \cdot \mathbf{n} \, dS = \int_\Omega \rho \frac{e_s^{v,eq} - e_s^v}{\tau_s^{V-T}} \, dV \quad s \in [1, .., N_v]
\end{aligned}
\tag{3.3}
$$

As the aim of this work is to simulate internal flow in elongated geometries, typically with circular cross-sectional areas, the equations are simplified into a quasi one-dimensional form. This means a one-dimensional form that allows for variations in the cross-sectional area along the x-coordinate.

By considering the control volume shown in Figure 3.1 and applying the no-slip condition on the lateral surface, the Equations 3.4 are derived.

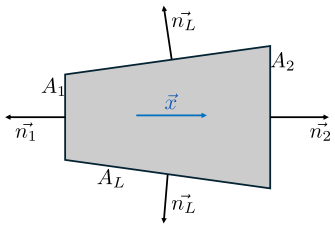


Figure 3.1: Control volume in quasi 1D approximation.

$$
\begin{aligned}
&\frac{\partial}{\partial t}\int_x \rho_s A \, dx + |\rho_s u A|_{x_1}^{x_2} = \int_x \dot{\omega}_s A \, dx \quad s \in [1, .., N_s] \\
&\frac{\partial}{\partial t}\int_x \rho u A \, dx + |(\rho u u + p)A|_{x_1}^{x_2} = \int_{S_L} p \, n_L \, dS_L \\
&\frac{\partial}{\partial t}\int_x E A \, dx + |u(E + p)A|_{x_1}^{x_2} = 0 \\
&\frac{\partial}{\partial t}\int_x \rho e_s^v A \, dx + |\rho e_s^v u A|_{x_1}^{x_2} = \int_x \rho \frac{e_s^{v,eq} - e_s^v}{\tau_s^{V-T}} A \, dx \quad s \in [1, .., N_v]
\end{aligned}
\tag{3.4}
$$

At this stage, we introduce a discretization of the x-coordinate into N cells with width Δx and define the following:

- The vector U, in Equation 3.5, as the integral average of the conservative variables over Δx, referred to as the conservative variable.

- The vector F, as shown in Equation 3.6 , referred to as the fluxes.
- The vector S, in Equation 3.7, as the integral average of the right-hand side term, multiplied by Δx, referred to as the source term.

$$U = \begin{pmatrix} \frac{1}{\Delta x}\int_{x_1(t)}^{x_2(t)} \rho_s A\, dx \quad s \in [1,..,N_s] \\ \frac{1}{\Delta x}\int_{x_1}^{x_2} \rho u A\, dx \\ \frac{1}{\Delta x}\int_{x_1}^{x_2} E A\, dx \\ \frac{1}{\Delta x}\int_{x_1}^{x_2} \rho e_s^v A\, dx \quad s \in [1,..,N_v] \end{pmatrix} \tag{3.5}$$

$$F = \begin{pmatrix} \rho_s u A \quad s \in [1,..,N_s] \\ (\rho u u + p)A \\ (E+p)\, uA \\ \rho e_s^v u A \quad s \in [1,..,N_v] \end{pmatrix} \tag{3.6}$$

$$S = \begin{pmatrix} \int_x \dot{\omega}_s A\, dx \quad s \in [1,..,N_s] \\ \int_{S_L} p\, n_L\, dS_L \\ 0 \\ \int_x \rho \frac{e_s^{v,eq} - e_s^v}{\tau_s^{V-T}} A\, dx \quad s \in [1,..,N_v] \end{pmatrix} \tag{3.7}$$

Now, introducing the convention where j represents the cell center and $j \pm \frac{1}{2}$ denotes the cell faces, and assuming Δx remains constant over time, we can express the conservative discretized equations in space in a more compact form, as shown in Equation 3.8 [33].

$$\frac{dU_j}{dt} = -\frac{1}{\Delta x}\left(F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}}\right) - \frac{1}{\Delta x} S_j \tag{3.8}$$

The final step is to discretize the equations in time. The type of discretization will determine the accuracy of the time integration. In this work, only explicit time integration is considered, and the method used for the time discretization will be discussed in Section 4.1.

3.2.2 Viscous losses treatment

In this work, viscous effects are not directly considered in the conservation equations; however, some of them are modeled by source terms added to the equations, following the approach explained in [4, 34].

A force was added to the left-hand side of the momentum equation to account for the friction force exerted by the wall on the fluid (F_{wall}), and a second force was included to account for losses due to rapid area changes (F_{loss}). This leads to the final form of the equation, as shown in Equation 3.9.

$$\begin{aligned}
&\frac{\partial}{\partial t}\int_x \rho_s A\,dx + |\rho_s u A|_{x_1}^{x_2} = \int_x \dot{\omega}_s A\,dx \quad s \in [1,..,N_s] \\
&\frac{\partial}{\partial t}\int_x \rho u A\,dx + |(\rho u u + p)A|_{x_1}^{x_2} = \int_{S_L} p\,n_L\,dS_L + F_{wall} + F_{loss} \\
&\frac{\partial}{\partial t}\int_x E A\,dx + |u(E+p)A|_{x_1}^{x_2} = 0 \\
&\frac{\partial}{\partial t}\int_x \rho e_s^v A\,dx + |\rho e_s^v u A|_{x_1}^{x_2} = \int_x \rho \frac{e_s^{v,eq} - e_s^v}{\tau_s^{V-T}} A\,dx \quad s \in [1,..,N_v]
\end{aligned} \tag{3.9}$$

with the redefinition of the source term S that becomes 3.10:

$$S = \begin{pmatrix} \int_x \dot{\omega}_s A\,dx \quad s \in [1,..,N_s] \\ \int_{S_L} p\,n_L\,dS_L + F_{wall} + F_{loss} \\ 0 \\ \int_x \rho \frac{e_s^{v,eq} - e_s^v}{\tau_s^{V-T}} A\,dx \quad s \in [1,..,N_v] \end{pmatrix} \tag{3.10}$$

F_{wall} treatment

Here a complete model is provided to calculate the contribution of F_{wall}:

$$F_{wall} = \tau\,\pi\,\overline{D}\,\left(x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}\right)$$

where τ is modeled as stated in the Darcy formula, and D is the diameter of the circular section of the geometry.

$$D = 2\left(\frac{A}{\pi}\right)^{1/2}$$

$$\tau_0 = \frac{-\rho f u|u|}{8}$$

The friction factor f is the Darcy-Weisbach factor and is calculated according to the following correlations:

$$f = \frac{64}{\Lambda\,Re}, \quad Re < 2000$$

$$f = \frac{0.032}{\Lambda}\left(\frac{Re}{2000}\right)^{0.3187}, \quad 2000 \le Re \le 4000$$

$$f = \frac{1}{\Lambda}\left[1.8\log_{10}(Re) - 1.5147\right]^{-2}, \quad Re > 4000$$

$$Re = \frac{\rho^* D|u|}{\mu^*}$$

where the Reynolds number is evaluated at the condition that corresponds to the conditions at the Eckert reference temperature.(T^*).

$$T^* = T + 0.5(T_w - T) + 0.22(T_{aw} - T)$$

$$\rho^* = \frac{\rho T}{T^*}$$

$$T_{aw} = \Lambda T$$

$$\Lambda = 1 + \frac{(\gamma - 1)}{2} r\, M^2$$

Where r is the recovery factor which is taken:

$$r = \sqrt{Pr}, \quad Re < 2000$$

$$r = Pr^{\frac{1}{3}} \quad Re \geq 2000$$

F_{loss} treatment

Here a complete model is provided to calculate the contribution of F_{loss}:

$$F_{loss} = \frac{\Delta P_{\text{loss}}}{L_{\text{loss}}} \frac{1}{A} \left(x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}} \right)$$

$$\Delta P_{\text{loss}} = -K_L \frac{1}{2} \rho u|u|$$

where K_L is the loss coefficient, and for an abrupt area change, it is taken to be equal to 0.25, according to [4, 34].

3.3 Fluxes determination

As shown in Equation 3.8, by integrating the conservative equation, we can obtain the value of the conservative variable at the cell center, making the approximation that the integral average coincides with the cell center value.

It is simple to understand that in a finite volume discretization, all the emphasis is on how to calculate the fluxes. Because, as mentioned, after the integration, we

only have values at the cell center; however, to calculate the fluxes, we need the fluid dynamic variables at the cell interfaces.

In order to determine the fluid dynamic variables at the cell interfaces, we must reconstruct the solution that we have at the cell center to recreate the function's trend up to the cell edge.

The type of reconstruction used determines the spatial order of the discretization. In Section 4.2, constant and a particular case of fifth-order reconstruction will be discussed.

After the reconstruction, we encounter a key moment in the flux determination process. At this stage, we will have, for example, a reconstructed function in cell j and another in cell $j+1$. Each of these functions will yield a different value at the cell interface$j+\frac{1}{2}$. We can denote these two values respectively as $U_{j+\frac{1}{2}^-}$ and $U_{j+\frac{1}{2}^+}$, as explained is showed in figure 3.2.

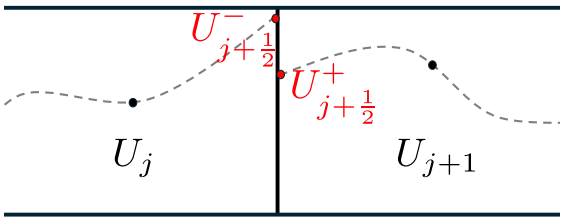


Figure 3.2: Interface reconstruction discontinuity

To chose the right flux on the interface, in applications where the convective term is dominant in the equations, such as hypersonic flow, it is important to respect the physics of the problem to ensure stability. Specifically, it is crucial to understand the source of the information that arrives at that interface in the real physical problem. This approach is known as the upwind approach [35, 36, 37, 38].

3.3.1 Riemann problem

To calculate the correct flux value at the cell interface, one method is to solve a Riemann problem [39], which involves understanding how the discontinuities created by the reconstruction will evolve while respecting the proper propagation of information along the system.

This is the approach used in the family of methods known as Godunov methods [40], which this work will follow.

When a discontinuity in the solution is present at the cell interface, a Riemann problem arises. One possible configuration is shown in Figure 3.3.

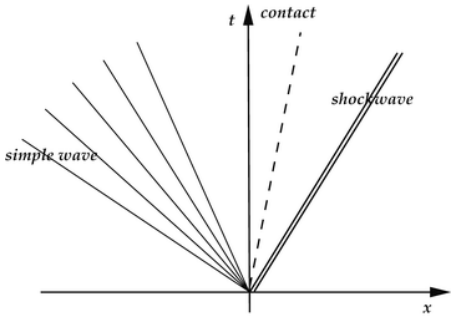


Figure 3.3: Possible wave configuration in a Riemann problem.

Resolving a Riemann problem means calculating the value of the fluid dynamic variable on the left and right sides of the contact surface generated by the problem and understanding in which zone the interface is situated . This is done by knowing the condition on the left of the first wave (in our case, $F(U_{j+\frac{1}{2}^-})$) and on the right of the third wave (in our case, $F(U_{j+\frac{1}{2}^+})$). Here, $F(U)$ represents the flux vector calculated at the condition U. The methods for calculating this will be detailed in Section 3.3.2.

In hyperbolic equations, such as those we are dealing with, information travels in the form of waves. In fluid dynamic problems, these waves can be expansions (first or third family waves), shocks (first or third family waves), or contact surface waves (second family waves).

It is possible to determine mathematically the exact solution of a Riemann problem (which will not be detailed here); however, this approach is usually computationally expensive and more unstable than solving the problem approximately, often leading to errors that are generally negligible.

Various approximate Riemann solvers have been developed, and in Section 4.3.2, one of them will be detailed. The basic principle relies on making assumptions about the wave configuration. For example, a very common Riemann solver is the Osher method [41], which treats the waves of the first and third families as expansion waves.

3.3.2 Flux Variables from the Fluid Dynamic Variables

When solving the simple Euler equation in Equation 3.1, an algebraic relation (shown in 3.11) can be derived to calculate the flux function $F(U)$ in terms of the conservative variables. Unfortunately, this relation can only be derived in the case of thermally and calorically perfect gases.

$$
\begin{aligned}
F_1 &= U_2 \\
F_2 &= \frac{U_2^2}{U_1}\left(\frac{3-\gamma}{2}\right) + U_3(\gamma - 1) \\
F_3 &= \frac{U_2}{U_1}\left[U_3\gamma - \frac{1}{2}\frac{U_2^2}{U_1^2}(\gamma - 1)\right]
\end{aligned}
\tag{3.11}
$$

In our case, the gas specific heat, and consequently the specific heat ratio, varies with temperature. Therefore, in order to calculate the flux function, it is necessary to determine the temperature separately. This is the most critical step in Section 3.4, where the derivation of fluid dynamic variables from conservative variables is detailed in the developed model.

Once the fluid dynamic variables are calculated, it is straightforward to compute the flux function using the definition in Equation 3.6.

3.4 Fluid dynamic variable calculation

In this section, the calculation of all fluid dynamic variables is presented, starting from the primitive variables and progressing to the derived ones.

3.4.1 Calculation of Primitive Variables

The calculation of the considered primitive variables is presented below:

$$
U' = \begin{pmatrix} \rho_s & s \in [1, .., N_s] \\ & u \\ & e \\ e_s^v & s \in [1, .., N_v] \end{pmatrix}
\tag{3.12}
$$

$$
\rho_s = \frac{U[s]}{A} \quad s \in [1, .., N_s]
$$

Where U[s] represents the s-th variable of the vector U introduced in Equation 3.5; For convenience, the total density is also computed.

$$
\rho = \sum_{s=1}^{N_s} \rho_s
$$

Now, the velocity can be easily calculated as:

$$
u = \frac{U[QDM]}{\rho A}
$$

where U[QDM] corresponds to the momentum component as defined in Equation 3.5. The internal energy e is computed as:

$$e = \frac{U[E]}{\rho A} - \frac{1}{2}\left(\frac{U[QDM]}{\rho A}\right)^2$$

where U[E] corresponds to the total energy component as defined in Equation 3.5.

$$e_s^v = \frac{U[E+s]}{\rho A} \quad s \in [1, .., N_v]$$

3.4.2 Derived variables calculation

$$Y_s = \frac{\rho_s}{\rho} \quad s \in [1, .., N_s]$$

$$X_s = \frac{\frac{Y_s}{MM_s}}{\sum_s \frac{Y_s}{MM_s}} \quad s \in [1, .., N_s]$$

$$MM = \sum_{s=1}^{N_s} X_s MM_s$$

$$R^* = \frac{R}{MM}$$

We now arrive at a crucial step in the implementation of the fluid dynamic variables calculation in this model: the temperature is calculated differently compared to the predefined model. In Sections 3.4.2, 3.4.2, and 3.4.2, the calculation method for all possible settings used in the current discretization will be detailed.

To calculate the specific heat at constant volume for individual species, NASA polynomials are used. In cases where the system is in thermodynamic non-equilibrium, these values are evaluated at T_{v_s} [15, 42]; otherwise, they are calculated at the standard system temperature.

The specific heat for the mixture is given by:

$$c_v = \sum_{i=1}^{N_s} c_{v_s} X_s$$

When chemical reactions are activated, the species production or depletion rate $\dot{\omega}_s$ is calculated using the model presented in Section 1.1.2.

For the determination of T_q the Park model [15, 25, 26], as shown in Equation 3.13, is used if thermodynamic non-equilibrium is considered. Otherwise, $T_q = T_{tr}$:

$$Tq = \sqrt{T_v T_{tr}} \tag{3.13}$$

Finally, if thermodynamic non-equilibrium is considered, it is necessary to calculate the relaxation time for each vibrational energy component. The detailed method for this calculation is provided in Section 1.2.1.

Calorically and Thermally Perfect Gas

In the approximation of a calorically perfect gas, the specific heat is constant, and the relationship between internal energy and temperature becomes algebraic: $e = c_v T$

In this case, since the specific heat is constant, it is an input for the calculation. From the definition above, the temperature can be calculated.

(In this simplified approximation, the vibrational conservation equations are not solved, and thermodynamic non-equilibrium is not accounted for.)

Calorically Non-Perfect Gas

When dealing with a calorically non-perfect gas, the definition of specific heat to be used is the more general one: $c_v = \frac{de}{dT}$

In this approximation, the specific heat is a function of temperature, so at this stage we do not have both c_v and T; the former is a function of the latter.

To calculate the temperature, starting from this definition, a Newton-Raphson-like method [43, 44] is used. For each iteration, the specific heat at constant volume and the internal energy are calculated using the NASA 9-coefficient polynomials [20], based on the temperature of that iteration.

(In this simplified approximation, the vibrational conservation equations are not solved, and thermodynamic non-equilibrium is not considered.)

Gas in Thermodynamic Non-Equilibrium

The case of a gas in thermodynamic non-equilibrium is the most general one, as it accounts for different temperatures in the system, whose calculation will be detailed here.

First, we calculate the translational-rotational temperature of the system, T_tr. To do this, we need to invert the internal energy definition, which is part of Equation 1.7, arriving at the explicit relation in Equation 3.14.

$$T_{tr} = \frac{e - \sum_{s=1}^{N_s} Y_s \left(e_s^{vib} + (\Delta h_f)_s^0 \right)}{\sum_{s=1}^{N_s} Y_s \left(\frac{3}{2} R_s^* + \frac{L_s}{2} R_s^* \right)} \tag{3.14}$$

Now, we can proceed to calculate the vibrational temperature for each vibrational energy considered in the system. This can be done easily using the model presented

in Section 1.2.1. A simple relationship exists at each instant between e_s^v and T_{v_s}, as shown in Equation 3.15

$$T_{v_s} = \frac{\Theta_{vs}}{\ln \frac{R_s^* \Theta_{vs}}{e_s^v} + 1} \tag{3.15}$$

If only one vibrational temperature is considered for the trigger temperature T_q defined in Equation1.2 temperature definition, this specific temperature must be computed. To calculate this temperature, we begin with the definition of the total vibrational energy in Equation 3.16 and invert this more complicated equation [3].

$$E_v = \sum_{s=1}^{N_v} \rho_s \frac{R}{M_s} \frac{\Theta_{vs}}{e^{\Theta_{vs}/T_v} - 1} \tag{3.16}$$

In this work, to calculate the "total" T_v and invert the equation 3.16 , an iterative Newton-Raphson-like method [43, 44] was used. The algorithm is based on iterating Equation 3.18 until convergence on T_v is reached. This equation is derived from Equation 3.17 , where the total vibrational energy is known, isolating one of the s-th vibrational energies on the right-hand side and the others on the left side.

$$\rho_1 \frac{R}{M_1} \frac{\Theta_{v1}}{e^{\Theta_{v1}/T_v^{k+1}} - 1} = E_v - \sum_{s=2}^{N_v} \rho_s \frac{R}{M_s} \frac{\Theta_{vs}}{e^{\Theta_{vs}/T_v^k} - 1} \tag{3.17}$$

$$T_v^{k+1} = \frac{\Theta_{v1}}{\ln \left[\left(\frac{E_v - \sum_{s=2}^{N_v} \rho_s R_s^* \frac{\Theta_{vs}}{e^{\Theta_{vs}/T_v^k} - 1}}{\rho_1 R_1^* \Theta_{v1}} \right)^{-1} + 1 \right]} \tag{3.18}$$

Where T_{k+1} and T_k are successive steps in the iterative method.

3.5 Δt calculation

A critical step in the explicit CFD code to ensure stability is the appropriate choice of Δt for integrating the conservative equations. The general idea for explicit integration is to resolve all physical phenomena over time.

In the model presented in Section 3.2, different physical mechanisms are present, each with a specific characteristic time to resolve the phenomena. In this model, the physical phenomena include convection, viscous losses, chemical reactions, and vibrational relaxation.

For the first mechanism, to respect the independence domain in the integration, we must satisfy the Courant–Friedrichs–Lewy (CFL) condition, which in this case is imposed as:

$$\Delta t = CFL \frac{\Delta x}{S_{max}}$$

where S_{max} is the maximum wave propagation speed, $S_{max} = max(a + |u|)$ [45].

As regards the other three expressed phenomena, they are accounted for in the equation through the source term S defined in 3.10. These phenomena make the equation become stiff for time integration because the source term often becomes larger than the conservative variables and the fluxes. This leads to rapid instability in the integration if Δt is not tuned properly.In this work, this tuning is based on the principle of keeping the factor $\frac{\Delta t S}{\Delta c}$ smaller than the conservative variable of a quantity K; this way, we prevent the previously explained situation and ensure stability. The Δt will result in:

$$\Delta t = \frac{U \Delta x}{SK}$$

where S and U refer to the components linked to the considered phenomenon.

By properly tuning the factor K based on the speed of the physical phenomena and experience in the simulation, we obtain an estimation of Δt for each phenomenon. Finally, we calculate Δt as:

$$\Delta t = min\left(CFL\frac{\Delta x}{S_{max}}, \frac{U\Delta x}{SK_{viscosity}}, \frac{U\Delta x}{SK_{chemistry}}, \frac{U\Delta x}{SK_{vibrational}}\right)$$

A more precise model for the estimation can be obtained by calculating the Jacobian matrix for the system and defining the time step, Δt, accordingly. A more detailed procedure for chemical phenomena is provided in [46].

3.6 Boundary Condition Management

Here, a brief overview will be provided on how boundary conditions are imposed in this code and on some types of boundary conditions that are implemented.

In particular, as explained in Section 3.1, the software is designed to be modular, so each *module* has its own boundary conditions, which can generally be Neumann or Dirichlet [35] type boundary conditions, or wall boundary conditions that simulate the presence of a boundary at the last or first cell interface in the module where it is set. This is implemented by solving a half Riemann problem, where the wall coincides with the contact surface and, obviously, has zero velocity.

For example, considering a wall condition to be imposed on the right side of the domain, knowing the condition at the left of the first family wave from the reconstruction in the left-hand cell, we can calculate the condition between this wave and the contact surface, which corresponds to the condition on the wall surface. Once this condition is calculated, it is straightforward to apply the flux functions to the fluid dynamic variables and impose the correct fluxes to simulate a wall interface.

Furthermore, two particular conditions can be imposed in this code: an *open* condition and a diaphragm burst.

The *open* condition is a unique feature of this code, as it connects the cells of different modules in the code's spatial discretization at each time step, treating them as linked. This approach allows them to be integrated together.

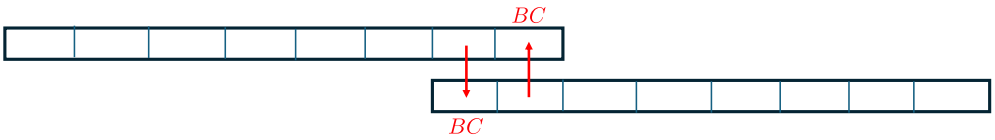


Figure 3.4: Control volume in quasi 1D approximation.

The condition is imposed by appropriately copying the conservative variables at the *modules* interfaces. Figure 3.4 graphically illustrates how this condition is imposed.

Regarding the diaphragm modeling, they are modeled through a wall boundary condition until the pressure differential exerted on them is less than a specified burst pressure. After this threshold is reached, they suddenly transition to an *open* boundary condition.

Chapter 4

HYQ1D: Numerical Methods

In this chapter, several numerical methods and techniques implemented in the HYQ1D code will be illustrated, focusing first on time integration and then on spatial reconstruction and flux evaluation.

4.1 Time integration

In Chapter 3.2.1, we explained how to discretize the spatial integrals. However, with equation 3.8, the process was halted before performing time integration of the complete conservation laws. In this section, we will continue from that point, illustrating how those integrals are discretized using the chosen time integration scheme.

4.1.1 First order approximation

Integrating equation 3.8 in time and discretizing it with a time step Δt, we obtain:

$$\int_t^{t+\Delta t} \frac{\partial U_j}{\partial t}\, dt = -\int_t^{t+\Delta t} \frac{1}{\Delta x}\left(F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}}\right) dt - \int_t^{t+\Delta t} \frac{1}{\Delta x} S_j\, dt \qquad (4.1)$$

Now, by applying the Leibniz approach, we can eliminate the time differentials on the left-hand side. Using the *fundamental theorem of calculus*, this term can be easily integrated, resulting in the following form:

$$U_j^{N+1} = U_j^N - \int_t^{t+\Delta t} \frac{1}{\Delta x}\left(F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}}\right) dt - \int_t^{t+\Delta t} \frac{1}{\Delta x} S_j\, dt \qquad (4.2)$$

The superscripts $N+1$ and N refer to the evaluation of the respective quantity at times $t+\Delta t$ and t, representing two consecutive steps in the integration process.

Up to this point, no approximations have been made in the time integration. However, at this stage, we must decide how to discretize the time integral of the flux and source term functions.

The simplest approach was chosen here: the functions under the integral sign are considered constant within the time step and equal to their value at time t. This allows us to factor them out of the integral, leading to the following discretized form [35]:

$$U_j^{N+1} = U_j^N - \frac{\Delta t}{\Delta x}\left(F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}}\right) - \frac{\Delta t}{\Delta x} S_j \tag{4.3}$$

Since the integrands in the time integrals were assumed to be constant in the discretization, this results in a first-order accuracy in the integral approximation and, consequently, a first-order accuracy in the time discretization.

4.1.2 RK3-SSP (Runge-Kutta 3 - Strong Stability Preserving)

Here, a completely different and more sophisticated approach will be introduced compared to the previous section.

Runge-Kutta (RK) methods are a generalized family of methods for integrating differential equations in time.

In this context, a three-step Runge-Kutta method is employed, which essentially divides the time integration into three sub-steps. This increases the order of accuracy of the time discretization to third order.

Specifically, the TVD-RK3 method is used, as presented by Chi-Wang Shu and Stanley Osher in [47]. As mentioned, this scheme is total variation diminishing (TVD), making it particularly effective for simulating fluxes prone to instability, such as those considered in this work.

The method is formally summarized by the integration steps shown in Equation4.4 [47, 48].

$$\begin{aligned} u^{(1)} &= u^n + \Delta t L(u^n), \\ u^{(2)} &= \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L(u^{(1)}), \\ u^{n+1} &= \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L(u^{(2)}). \end{aligned} \tag{4.4}$$

In our case, u represents the conservative variable U, and the operator $L(u)$

corresponds to the right-hand side of equation 3.8, specifically:

$$L(u) = -\frac{1}{\Delta x}\left(F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}}\right) - \frac{1}{\Delta x}S_j$$

This method, due to its advantageous characteristics, is widely used, and further optimizations and variations have been proposed, such as an implicit version of the current scheme [49].

4.1.3 Strang Splitting

Since the HYQ1D code is generally designed to handle chemical and thermodynamic non-equilibrium, as previously mentioned, the source term S makes the integration stiff, particularly due to the presence of source terms in the species conservation equations and the vibrational energy conservation equations. One technique to manage stiff equations while reducing computational cost is to apply a time splitting method [50].

Let $f(U)$ represent the stiff part of the right-hand side of the compact equation 3.8, and $g(U)$ the non-stiff part. This type of method essentially handles the time integration in different steps, first integrating equation 3.8 with only the stiff part and then with only the non-stiff part of the right-hand side.

In this work, specifically, the *Strang Splitting* method [51] was chosen for the implementation in the code, with its steps summarized in equation 4.5.

$$\begin{aligned}
\frac{dU^*}{dt} &= f(U^*), \quad U^*(0) = U(t), \quad \text{on } \left[0, \frac{\Delta t}{2}\right], \\
\frac{dU^{**}}{dt} &= g(U^{**}), \quad U^{**}(0) = U^*\left(\frac{\Delta t}{2}\right), \quad \text{on } [0, \Delta t], \\
\frac{dU^{***}}{dt} &= f(U^{***}), \quad U^{***}\left(\frac{\Delta t}{2}\right) = U^{**}(\Delta t), \quad \text{on } \left[\frac{\Delta t}{2}, \Delta t\right], \\
U(t + \Delta t) &= U^{***}(\Delta t).
\end{aligned} \tag{4.5}$$

As shown, the method consists of first integrating the equations considering the stiff part for a time step of $\frac{\Delta t}{2}$, then integrating only the non-stiff part for a full Δt, and finally performing another integration of the stiff part for $\frac{\Delta t}{2}$.

This approach results in a reduced error in the integration of the stiff part compared to other methods organized in only two steps.

4.2 Spatial Reconstruction

In the finite volume method, the integration of the conservation laws only allows for the calculation of the cell-averaged value of the conservative variables, typically

associated with the cell center. However, physically, the conservative variable exhibits a specific variation within the cell. This chapter focuses on how this variation is reconstructed, proposing two approaches: constant reconstruction and WENO (Weighted Essentially Non-Oscillatory) reconstruction.

4.2.1 Constant Reconstruction

Constant reconstruction is the simplest method; it assumes that the conservative variables remain constant within the cell, with a value equal to the cell-averaged value. This approach results in a first-order accuracy in the spatial discretization.

4.2.2 WENO (Weighted Essentially Non Oscillatory)

The second scheme implemented in the HYQ1D code is a Weighted Essentially Non-Oscillatory (WENO) scheme, specifically the version illustrated in [52], but really useful reference to implement the scheme are also [53, 54]. The WENO scheme is based on Essentially Non-Oscillatory (ENO) schemes and employs a weighting mechanism to reconstruct a polynomial of fifth-order accuracy (in the case considered) from three ENO reconstructions.

Using u as the representative value for reconstruction, equation 4.6 presents the three ENO reconstructions along with the corresponding stencils used.

$$\begin{aligned} u^{(1)}_{j+1/2} &= \frac{1}{3}\bar{u}_{j-2} - \frac{7}{6}\bar{u}_{j-1} + \frac{11}{6}\bar{u}_j \quad on \quad \{I_{j-2};\, I_{j-1};\, I_j\} \\ u^{(2)}_{j+1/2} &= -\frac{1}{6}\bar{u}_{j-1} + \frac{5}{6}\bar{u}_j + \frac{1}{3}\bar{u}_{j+1} \quad on \quad \{I_{j-1};\, I_j;\, I_{j+1}\} \\ u^{(3)}_{j+1/2} &= \frac{1}{3}\bar{u}_j + \frac{5}{6}\bar{u}_{j+1} - \frac{1}{6}\bar{u}_{j+2} \quad on \quad \{I_j;\, I_{j+1};\, I_{j+2}\} \end{aligned} \tag{4.6}$$

The calculation of the weights to combine these three polynomials begins with determining the so-called *smoothness indicator*, which measures the relative smoothness of the variable being reconstructed on the selected stencil. The expression for the *smoothness indicator* is presented in Equation 4.7.

$$\begin{aligned} \beta_1 &= \frac{13}{12}\left(\bar{u}_{j-2} - 2\bar{u}_{j-1} + \bar{u}_j\right)^2 + \frac{1}{4}\left(\bar{u}_{j-2} - 4\bar{u}_{j-1} + 3\bar{u}_j\right)^2 \\ \beta_2 &= \frac{13}{12}\left(\bar{u}_{j-1} - 2\bar{u}_j + \bar{u}_{j+1}\right)^2 + \frac{1}{4}\left(\bar{u}_{j-1} - \bar{u}_{j+1}\right)^2 \\ \beta_3 &= \frac{13}{12}\left(\bar{u}_j - 2\bar{u}_{j+1} + \bar{u}_{j+2}\right)^2 + \frac{1}{4}\left(3\bar{u}_j - 4\bar{u}_{j+1} + \bar{u}_{j+2}\right)^2 \end{aligned} \tag{4.7}$$

Now we can define the nonlinear weights as follows:

$$
\begin{aligned}
&w_i = \frac{\alpha_i}{\alpha_1 + \alpha_2 + \alpha_3} \qquad where: \\
&\alpha_i = \frac{\gamma_i}{(\epsilon + \beta_i)^2}, \quad i = 1,2,3 \quad and \quad \gamma_1 = \frac{1}{10}, \quad \gamma_2 = \frac{3}{5}, \quad \gamma_3 = \frac{3}{10}
\end{aligned} \tag{4.8}
$$

With all the elements established, the WENO reconstructed polynomial can be defined in equation 4.9, resulting in a fifth-order approximation defined over the five-cell stencil $\{I_{j-2};\ I_{j-1};\ I_j;\ I_{j+1};\ I_{j+2}\}$.

$$u_{j+1/2} = w_1 u^{(1)}_{j+1/2} + w_2 u^{(2)}_{j+1/2} + w_3 u^{(3)}_{j+1/2} \tag{4.9}$$

It is important to emphasize that the nonlinear weights are defined to adaptively avoid including oscillations; therefore, in the polynomial definition presented in equation 4.9, nonsmooth ENO stencils will be excluded. Due to this mechanism, when the reconstruction is applied in the presence of discontinuities, the order of accuracy decreases, at most to third order.

Further improvements and considerations can be made to the WENO scheme to enhance its performance, as demonstrated in [55]. However, these aspects are not addressed in this work.

4.3 Flux evaluation

As illustrated in section 3.3 , a crucial aspect of a finite volume algorithm is flux determination. In the HYQ1D code, the choice between the left and right reconstructed variables or fluxes (as illustrated in Figure 3.2) can be made using various methods. In the following, L and R in the subscript will indicate the left and right positions with respect to the cell interface.

4.3.1 Lax-Wendroff

The first method presented is the Lax-Wendroff method (Equation 4.10), which is discussed separately from the others because it is a non-upwind method. It is linearly stable up to a CFL number of unity for the model hyperbolic equation, and it is second-order accurate in both space and time; however, it is non-monotone and therefore prone to oscillations in the solution [56]. This method is typically used in the code without previously reconstructing the variables in the cells.

$$F_{j+\frac{1}{2}} = F(U_{j+\frac{1}{2}}), \quad U_{j+\frac{1}{2}} = \frac{1}{2}\left(U^n_j + U^n_{j+1}\right) - \frac{1}{2}\frac{\Delta t}{\Delta x}\left[F(U^n_{j+1}) - F(U^n_j)\right] \tag{4.10}$$

4.3.2 Approximate Riemann Solver

This section presents the methods that incorporate flux choices based on the estimation of characteristic wave velocities. All of these methods are monotonic schemes.

For all the proposed approximate Riemann solvers, no extensions are made to the simple case of the Euler equations. The homogeneous part of the additional equations introduced by chemistry and vibrational relaxation adds more waves to the system however, all these waves share the velocity of the contact surface wave (the second family of waves). While extensions of the solver could be made to better account for multicomponent gases, such improvements are not considered in this work.

Rusanov

The Rusanov method, presented in Equation 4.11 [57], is one of the simplest approximate Riemann solvers formulated. It is based on the two-wave formulation of the HLL method, where the velocity of the left wave ($-S^+$) and the velocity of the right wave (S^+) are taken equal to the following defined velocity S^+.

$$S^+ = \max\left\{|u_L| + a_L, |u_R| + a_R|\right\}$$

$$F_{i+\frac{1}{2}} = \frac{1}{2}\left(F_L + F_R\right) - \frac{1}{2}S^+\left(U_R - U_L\right) \tag{4.11}$$

Lax-Friedrichs

Starting from the approach of the Rusanov method [57] and defining S^+ as the maximum wave speed present at a given time according to the CFL condition, $S^+ = S^n_{\max} = \frac{C_{fl}\Delta x}{\Delta t}$, we arrive at the Lax-Friedrichs method, as reported in equation 4.12 [58].

$$F_{j+\frac{1}{2}} = \frac{1}{2}\left[F(U^n_j) + F(U^n_{j+1})\right] - \frac{1}{2}\frac{\Delta x}{\Delta t}\left(U^n_{j+1} - U^n_j\right) \tag{4.12}$$

HLLC

A more sophisticated, yet still approximate, Riemann solver is the HLLC solver. This method is more comprehensive because it considers a three-wave approximation that travels with velocities S_L, S_*, and S_R, making it suitable for the Euler equations. The solver applies Rankine–Hugoniot conditions across each of the waves and, in the implemented variant, enforces the exact condition of equal pressure across the contact surface.

The main steps of the HLLC method are outlined below, referring to the *Variant 1* HLLC scheme presented in [59]:

- Pressure estimation

$$\mathbf{p}_* = \max(0, p_{\text{pvrs}}), \quad p_{\text{pvrs}} = \frac{1}{2}(p_L + p_R) - \frac{1}{2}(u_R - u_L)\bar{\rho}\bar{a} \tag{4.13}$$

$$\textit{where} \quad \bar{\rho} = \frac{1}{2}(\rho_L + \rho_R), \quad \bar{a} = \frac{1}{2}(a_L + a_R)$$

- Wave speed estimation

$$\mathbf{S_L} = u_L - a_L q_L, \quad \mathbf{S_R} = u_R + a_R q_R \tag{4.14}$$

$$\textit{where} \quad q_K = \begin{cases} 1 & \text{se } p_* \leq p_K \\ \left[1 + \frac{\gamma+1}{2\gamma}\left(\frac{p_*}{p_K} - 1\right)\right]^{1/2} & \text{se } p_* > p_K \end{cases}$$

$$\mathbf{S}_* = \frac{p_R - p_L + \rho_L u_L(S_L - u_L) - \rho_R u_R(S_R - u_R)}{\rho_L(S_L - u_L) - \rho_R(S_R - u_R)} \tag{4.15}$$

- Flux evaluation

$$\mathbf{F}^{\text{hllc}}_{\mathbf{i+\frac{1}{2}}} = \begin{cases} F_L & \text{se } 0 \leq S_L \\ F_{*L} & \text{se } S_L \leq 0 \leq S_* \\ F_{*R} & \text{se } S_* \leq 0 \leq S_R \\ F_R & \text{se } 0 \geq S_R \end{cases} \tag{4.16}$$

$$\textit{where} \quad F_{*K} = \frac{S_*(S_K U_K - F_K) + S_K\left(p_K + \rho_L(S_K - u_K)(S_* - u_K)\right) D_*}{S_K - S_*}$$

$$\textit{and} \quad D_* = [0, A, 0, 0, AS_*]^T$$

The vector D was multiplied by the area A of the cell because the area is included in the definitions of U and F, as the formulation is quasi one-dimensional rather than strictly one-dimensional.

Chapter 5

Free Piston model

5.1 Introduction

Various models have been developed to simulate a free piston-driven shock tunnel, often employing a Lagrangian approach to track piston movement along the compression tube [4, 5, 6]. In this study, we develop a free piston model that integrates with the Eulerian formulation, eliminating the need for a Lagrangian approach, allowing all facilities to be simulated with the same code, as demonstrated in section 9.3.

A free-driven piston is a key component for compressing and generating the high-energy/enthalpy conditions required in hypersonic facilities.

In the context of a finite volume method, the presence of an external object moving through the domain presents challenges, especially when aiming for a globally conservative formulation and a global control domain that does not change in size over time.

To overcome these difficulties, the piston was modeled as an interface with no spatial extension. This interface moves through the computational finite volume cells, splitting any cell it passes through into two sub-cells. A sub-grid model was developed to represent the piston's behavior within a specific cell, with two main objectives: to simulate the piston's dynamics and to capture the fluid behavior within the standard finite volume cells.

5.2 Sub-grid piston model

To achieve the previously stated goals, a sub-grid model was developed. This model is based on the splitting of a cell into two sub-cells at the piston interface (shown in Fig. 5.1).

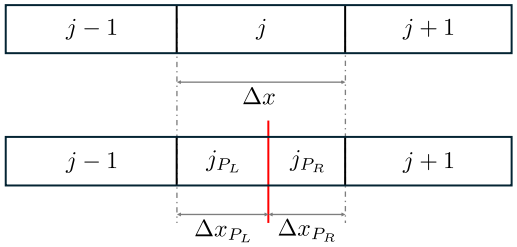


Figure 5.1: Cell splitting originating from the piston interface.

The two sub-cells originating from this cell splitting result have one interface moving at the piston velocity (one of them interfaces being the piston interface).

To describe the fluid dynamics of the two sub-cells using a conservative formulation, a new set of Euler equations extended to the case of control volumes in motion with a generic velocity were derived in Section 5.2.1.

The other goal, reproducing the piston dynamics, is achieved based on mechanical considerations. We are describing a freely driven piston, so its motion is determined by the forces acting on it. Refer to the free body diagram in Fig. 5.2, which is presented in the context of d'Alembert's principle.

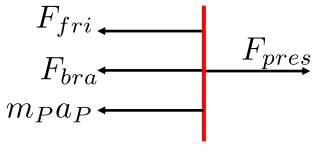


Figure 5.2: Free body diagram.

$$F_{pres} - F_{fri} - F_{bra} = m_p a_p \tag{5.1}$$

$$\text{where} \quad F_{fri} = \mu F_{\perp} \quad ; \quad F_{pres} = (p_{P_L} - p_{P_R}) A_P \quad .$$

Considering that the piston area (A_P), friction coefficient (μ), and brake force (F_{bra}) are characteristic parameters of the specific facilities, in order to evaluate the piston acceleration, we need to know the pressure at the piston faces. In this context, they represent a cell interface for the sub-grid cell. By integrating the conservation law derived in Section 5.2.1, we obtain the conservative variables at the cell centers and from them, the primitive variables at the cell centers. However, we need the pressure on the cell interface, which was achieved by solving a half-Riemann

problem, on the reconstructed variable, on the piston faces (set-up and resolution process explained in Section 5.2.2). Once the pressure differential on the piston is obtained, we have all the necessary information to calculate the acceleration of this component at the current time step (t^N).

To close the equation in Section 5.2.2 and to evaluate the piston's kinematics and its position at time $t^{N+1} = t^N + \Delta t$, we need to calculate the new position (${x_P}^{N+1}$).

Starting from the definition of acceleration, $a = \frac{dV}{dt}$, and using a first-order forward finite difference approximation, we can express:

$$a^N = \frac{V^{N+1} - V^N}{\Delta t_{NEW}} \tag{5.2}$$

Following the same approach, we can express V^{N+1} and V^N using the definition of velocity, $V = \frac{dx}{dt}$. By applying a backward discretization, we arrive at Equation 5.3:

$$a^N = \frac{\frac{x^{N+1} - x^N}{\Delta t_{NEW}} - \frac{x^N - x^{N-1}}{\Delta t_{OLD}}}{\Delta t_{NEW}} \tag{5.3}$$

where $\Delta t_{\mathrm{OLD}} = t^N - t^{N-1}$ and $\Delta t_{\mathrm{NEW}} = t^{N+1} - t^N$.

The equation now derived, along with Equation 5.2, are expressed in therm of x^{N+1} and V^{N+1} in Equations 5.4-5.5 , thus closing the model and fully capturing the piston's kinematic behavior.

$$V^{N+1} = a^N \Delta t_{NEW} + V^N \tag{5.4}$$

$$x^{N+1} = a^N \Delta t_{NEW}^2 + x^N (1 + \frac{\Delta t_{NEW}}{\Delta t_{OLD}}) - x^{N-1} \frac{\Delta t_{NEW}}{\Delta t_{OLD}} \tag{5.5}$$

Up to this point, the model allows us to track both the piston and the fluid behavior within a specific cell. However, when the piston moves too close to a cell boundary, certain issues arise. If this model would be used with an explicit time integration approach, we must satisfy the Courant–Friedrichs–Lewy (CFL) condition, which in this case is imposed as:

$$\Delta t = CFL \frac{\Delta x}{S_{max}}$$

where S_{max} is the maximum wave propagation speed, $S_{max} = max(a + |u|)$.

It becomes evident that as the piston moves, it reduces the size of the sub-cell in the direction of motion while expanding the size of the other sub-cell. The sub-cell that decreases in size requires, according to the CFL condition, a progressively

smaller time step. As $\Delta x_{sub-cell} \to 0$, this leads to $\Delta t \to 0$, eventually causing the code to stall.

To overcome these difficulties, a local remeshing mechanism (explained in Section 5.2.3) was developed to retain $\Delta x_{P_{L,R}}$ above a certain minimum trigger value, Δx_{trig}, with the aim of making this model applicable in a CFD code using explicit time integration.

5.2.1 Quasi 1D Euler equation for control volume deforming with a generic velocity

Here, the derivation is presented only for the Euler equations because they affect the time derivative of the conservative laws. Therefore, the derivation remains unchanged even if source terms are added to the set of equations under consideration. Similarly, no difference will be found between the derivation of the continuity equation and the conservation balance for the i-th species in a multicomponent fluid model, so only the first one was showed.

Starting from the differential conservative form of the Euler equations presented and derived in [32]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \qquad \text{(Balance of mass)} \tag{5.6}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + p\mathbf{I}) = 0 \qquad \text{(Balance of momentum)} \tag{5.7}$$

$$\frac{\partial (E)}{\partial t} + \nabla \cdot ((E + p)\mathbf{v}) = 0 \qquad \text{(Balance of energy)} \tag{5.8}$$

where E is the total energy per unit volume $\left[\frac{J}{m^3}\right]$: $E = \rho\left(\frac{1}{2}|\mathbf{v}|^2 + e\right)$, with e being the specific internal energy $\left[\frac{J}{Kg}\right]$.

We can integrate the presented equations on a generic control volume Ω moving with a generic velocity (eventually different from the fluid velocity).

$$\int_{\Omega(t)} \frac{\partial \rho}{\partial t}\, dV + \oint_{\partial\Omega(t)} \rho \mathbf{v} \cdot \mathbf{n}\, dS = 0 \qquad \text{(Balance of mass)} \tag{5.9}$$

$$\int_{\Omega(t)} \frac{\partial (\rho \mathbf{v})}{\partial t}\, dV + \oint_{\partial\Omega(t)} (\rho \mathbf{v}\mathbf{v} + p\mathbf{I}) \cdot \mathbf{n}\, dS = 0 \qquad \text{(Balance of momentum)} \tag{5.10}$$

$$\int_{\Omega(t)} \frac{\partial\,(E)}{\partial t}\, dV + \oint_{\partial\Omega(t)} ((E + p)\mathbf{v}) \cdot \mathbf{n}\, dS = 0 \qquad \text{(Balance of energy)} \tag{5.11}$$

From this formulation, it is clear that the motion and shape changes of the control volume do not allow the integration operation to be interchanged with the

temporal derivative. To address this, the Reynolds transport theorem ([60]), as reported in Eq. 5.12, was used.

$$\frac{d}{dt}\int_{\Omega(t)} f(\mathbf{x},t)\,dV = \int_{\Omega(t)} \frac{\partial f(\mathbf{x},t)}{\partial t}\,dV + \oint_{\partial\Omega(t)} f(\mathbf{x},t)\frac{d\mathbf{x}}{dt}\cdot\mathbf{n}\,dS = 0 \tag{5.12}$$

where $f(\mathbf{x},t)$ is the vector or scalar function to be integrated, which in our case will be ρ, $\rho\mathbf{v}$, or E. Here, $\mathbf{x}$ is the coordinate of a point within the control volume, and the temporal derivative of $\mathbf{x}$ will be denoted as $\mathbf{V_V}$.

The derivation procedure is the same for each equation; thus, only the derivation of the energy equation is detailed.

Starting from Equation 5.11, we can add and subtract at left hand side the following term: $\oint_{\partial\Omega(t)} E\mathbf{V_V}\cdot\mathbf{n}\,dS$. Now apply the Reynolds transport theorem we arrive at the form:

$$\frac{d}{dt}\int_{\Omega(t)} E\,dV + \oint_{\partial\Omega(t)} (E(\mathbf{v}-\mathbf{V_V})+p\mathbf{v})\cdot\mathbf{n}\,dS = 0$$

With the complete integral 3D set of equations which, in accordance with [61]-[62],became:

$$\frac{d}{dt}\int_{\Omega(t)} \rho\,dV + \oint_{\partial\Omega(t)} \rho(\mathbf{v}-\mathbf{V_V})\cdot\mathbf{n}\,dS = 0 \quad \text{(Balance of mass)} \tag{5.13}$$

$$\frac{d}{dt}\int_{\Omega(t)} \rho\mathbf{v}\,dV + \oint_{\partial\Omega(t)} (\rho\mathbf{v}(\mathbf{v}-\mathbf{V_V})+p\mathbf{I})\cdot\mathbf{n}\,dS = 0 \quad \text{(Balance of momentum)} \tag{5.14}$$

$$\frac{d}{dt}\int_{\Omega(t)} E\,dV + \oint_{\partial\Omega(t)} (E(\mathbf{v}-\mathbf{V_V})+p\mathbf{v})\cdot\mathbf{n}\,dS = 0 \quad \text{(Balance of energy)} \tag{5.15}$$

$$\frac{d}{dt}\int_{\Omega(t)} \rho\,dV + \oint_{\partial\Omega(t)} \rho(\mathbf{v}-\mathbf{V_V})\cdot\mathbf{n}\,dS = 0 \quad \text{(Balance of mass)} \tag{5.16}$$

$$\frac{d}{dt}\int_{\Omega(t)} \rho\mathbf{v}\,dV + \oint_{\partial\Omega(t)} (\rho\mathbf{v}(\mathbf{v}-\mathbf{V_V})+p\mathbf{I})\cdot\mathbf{n}\,dS = 0 \quad \text{(Balance of momentum)} \tag{5.17}$$

$$\frac{d}{dt}\int_{\Omega(t)} E\,dV + \oint_{\partial\Omega(t)} (E(\mathbf{v}-\mathbf{V_V})+p\mathbf{v})\cdot\mathbf{n}\,dS = 0 \quad \text{(Balance of energy)} \tag{5.18}$$

Now, the resulting equations are simplified in the case of a quasi-1D approximation, which means the control volume can change its area, and the fluid properties vary only in the x-direction.

In the following consideration to pass from 3D equation to quasi 1D ones we will use the general control volume showed in Figure 3.1.

We can rewrite the fluxes in each equation expliciting the surface integrals on each surface of the considered control volume and consider the volume differential as: $dV = dAdx$. In compact form we have:

$$\frac{d}{dt}\int_{\Omega(t)} \rho \, dAdx + \int_{S_{1,2,L}(t)} \rho(\mathbf{v} - \mathbf{V_V}) \cdot \mathbf{n_{1,2,L}} \, dS = 0 \qquad \text{(Balance of mass)} \tag{5.19}$$

$$\frac{d}{dt}\int_{\Omega(t)} \rho\mathbf{v} \, dAdx + \int_{S_{1,2,L}(t)} (\rho\mathbf{v}(\mathbf{v} - \mathbf{V_V}) + p\mathbf{I}) \cdot \mathbf{n_{1,2,L}} \, dS = 0 \qquad \text{(Balance of momentum)} \tag{5.20}$$

$$\frac{d}{dt}\int_{\Omega(t)} E \, dAdx + \int_{S_{1,2,L}(t)} (E(\mathbf{v} - \mathbf{V_V}) + p\mathbf{v}) \cdot \mathbf{n_{1,2,L}} \, dS = 0 \qquad \text{(Balance of energy)} \tag{5.21}$$

Now we can integrate on the area the first integral in the equations and for the second integral we can applying the scalar product and do the surface integrals arriving at the quasi 1D form of the starting equations.

$$\frac{d}{dt}\int_{x(t)} \rho A \, dx = -|\rho(v - V_V)A|_{x_1(t)}^{x_2(t)} \qquad \text{(Balance of mass)} \tag{5.22}$$

$$\frac{d}{dt}\int_{x(t)} \rho v A \, dx = -|(\rho v(v - V_V) + p)A|_{x_1(t)}^{x_2(t)} - p|A|_{x_1(t)}^{x_2(t)} \qquad \text{(Balance of momentum)} \tag{5.23}$$

$$\frac{d}{dt}\int_{x(t)} EA \, dx = -|\left(E(v - V_V) + pv\right)A|_{x_1(t)}^{x_2(t)} \qquad \text{(Balance of energy)} \tag{5.24}$$

where v and V_v represent the x-component of the fluid velocity and the control volume velocity, respectively, and a linear area variation is considered within the cell.

To integrate the obtained equation in time, it is necessary to determine how to evaluate the time integrals on the right-hand side. This choice will define the time scheme of the employed method. The simplest approach is to assume that the fluxes remain constant during the integration, which corresponds to using an explicit first-order time discretization.

In the context of the control volume method, where j denotes the cell center and $j \pm \frac{1}{2}$ represents the cell interfaces, we now present Equation 5.25, which we aimed

to derive and is used in the model described in this manuscript. This equation differs slightly from the one presented in [61], as no approximation is made to avoid having the control volume dimension at t^{N+1} on the right-hand side. This is because, by expressing the piston coordinate law as shown in Equation 5.5, the resulting calculation remains explicit.

$$U_j^{N+1} = U_j^N \frac{\Delta x^N}{\Delta x^{N+1}} - \frac{\Delta t}{\Delta x^{N+1}} \left(F^N_{j+\frac{1}{2}} - F^N_{j-\frac{1}{2}} \right) - \frac{\Delta t}{\Delta x^{N+1}} S_j^N \tag{5.25}$$

$$\text{where } U_j^N = \begin{pmatrix} \frac{1}{\Delta x^N} \int_{x_1(t)}^{x_2(t)} \rho A \, dx \\ \frac{1}{\Delta x^N} \int_{x_1(t)}^{x_2(t)} \rho v A \, dx \\ \frac{1}{\Delta x^N} \int_{x_1(t)}^{x_2(t)} E A \, dx \end{pmatrix}, \; F^N_{j\pm\frac{1}{2}} = \begin{pmatrix} |\rho(v - V_V)A|^N_{j\pm\frac{1}{2}} \\ |(\rho v(v - V_V) + p)A|^N_{j\pm\frac{1}{2}} \\ |\left(E(v - V_V) + pv\right) A|^N_{j\pm\frac{1}{2}} \end{pmatrix}$$

$$\text{and } S_j^N = \begin{pmatrix} 0 \\ p_j(A_{j+\frac{1}{2}} - A_{j-\frac{1}{2}}) \\ 0 \end{pmatrix}.$$

5.2.2 Riemann problem on the piston faces

To compute the fluid dynamic variables at the piston interface, a half Riemann problem was solved. The piston interface imposes its velocity on the adjacent fluid, and this boundary condition was enforced by aligning the piston interface with the second family wave, which travels with the fluid's Eulerian velocity and arises from the flow discontinuity.

In more detail, we consider the case of the piston moving to the right, focusing on the fluid on its left side (the other cases are treated symmetrically).

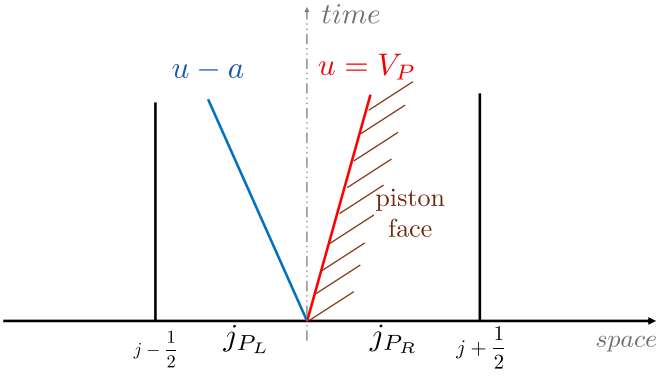


Figure 5.3: Half Riemann problem representation.

Figure 5.3 illustrates the problem setup, where the second family wave coincides with the piston face, dividing the phase plane into two regions. In this case, the

left reconstruction from the cell center of the piston semi-cell to the wall represents the state on the left of the first family wave. With this known state and the piston interface velocity, calculated using Eq. 5.2, an appropriate Riemann solver can determine the pressure and other fluid dynamic variables in the zone between the first and second family waves, which coincide with the fluid dynamic variables at the piston face.

5.2.3 Local remeshing mechanism

As previously mentioned, in the context of explicit time discretization, a remeshing mechanism is necessary to overcome the issue of decreasing size in one of the two sub-cells. The purpose of this section is to present a remeshing mechanism capable of retaining $\Delta x_{P_{L,R}}$ above a certain minimum value, Δx_{trig}.

The routine consists of two primary remeshing phases. Initially, it locally adjusts the mesh where the small-width problem occurs. The second phase, after a certain spatial margin from the critical condition, restores the original mesh.

The two steps of the routine are discussed with the representation of the first step shown in 5.5 and the second in 5.4. In the illustrations, the piston interface is displayed in red, and the mechanism is demonstrated with the piston moving from left to right. The case of opposite velocity exhibits a completely mirrored mode of operation.

1. First remeshing phase The routine begins when the calculated piston position from equation 5.3 does not guarantee the minimal value of $\Delta x_{P_{L,R}}$. Mathematically, this condition is expressed as $\Delta x_{P_L} \leq \Delta x_{trig}$ (as shown in the upper part of figure 5.4). At this point, the routine resizes the right cell to incorporate the piston interface, ensuring that the distance between the piston interface and the neighboring cell interfaces is greater than Δx_{trig} (condition illustrated in the lower part of figure 5.4). The remeshed cell dimension became: $\Delta x_{R_{L,R}} = \Delta x \mp 2\Delta x_{trig}$.

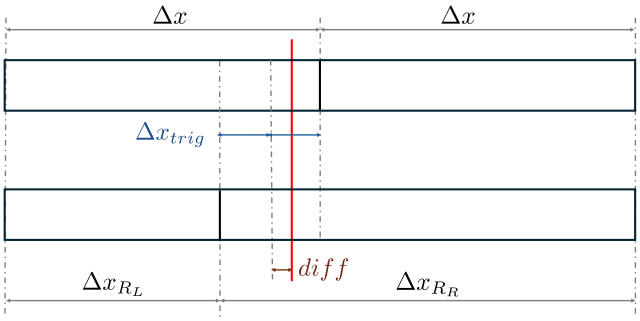


Figure 5.4: First remeshing phase.

2. Second remeshing phase Once the first phase is completed, the routine activates the possibility to trigger the second remeshing phase. As the piston continues its motion, it moves further away from the left interface of the cell in which it is located, with Δx_{P_L} increasing, as shown in the upper part of figure 5.5. Eventually, the piston reaches a position that would have been acceptable for the previous mesh configuration, a condition mathematically expressed as $\Delta x_{P_L} > 3\Delta x_{trig}$. When this condition is met, the trigger for the second remeshing phase is activated, and the mesh is reshaped to its original dimensions, as illustrated in the lower part of figure 5.5.

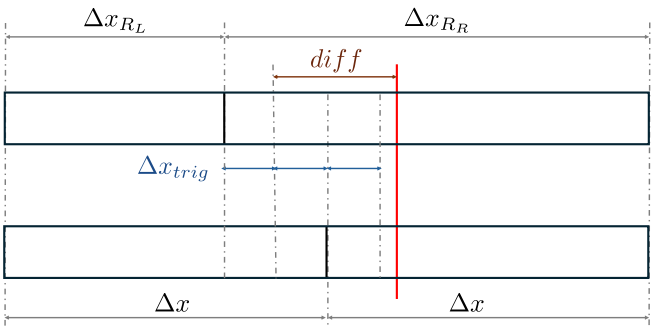


Figure 5.5: Second remeshing phase.

5.3 Standalone Piston simulation

Here, a stand-alone simulation isolating the behavior of the free piston model is presented. The test setup consists of a tube with the free piston inside, with a high-pressure condition on the left side of the piston and a low-pressure condition

on the right side.

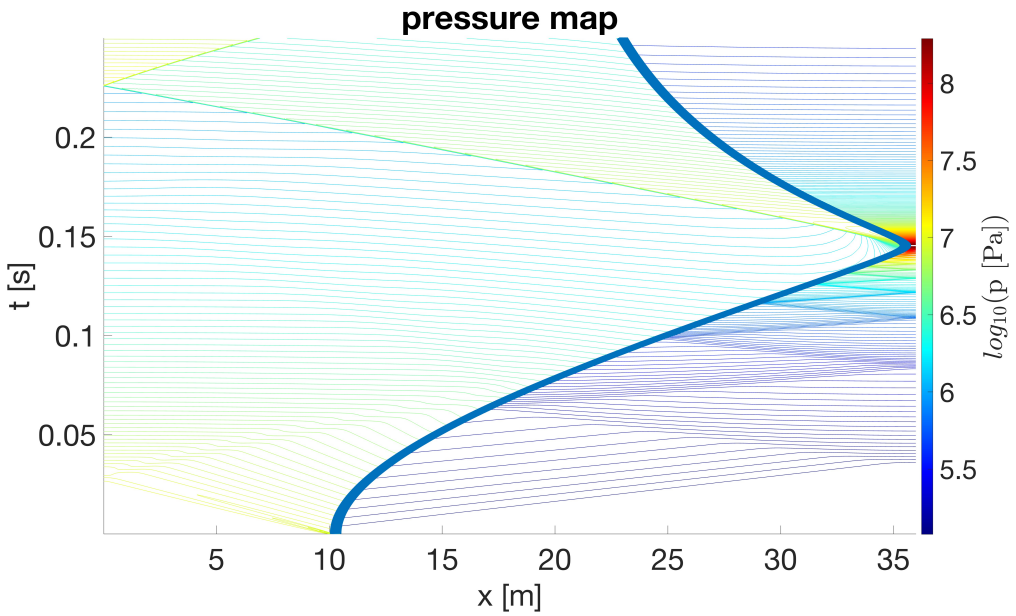


Figure 5.6: Contour map of the base-10 logarithm of pressure

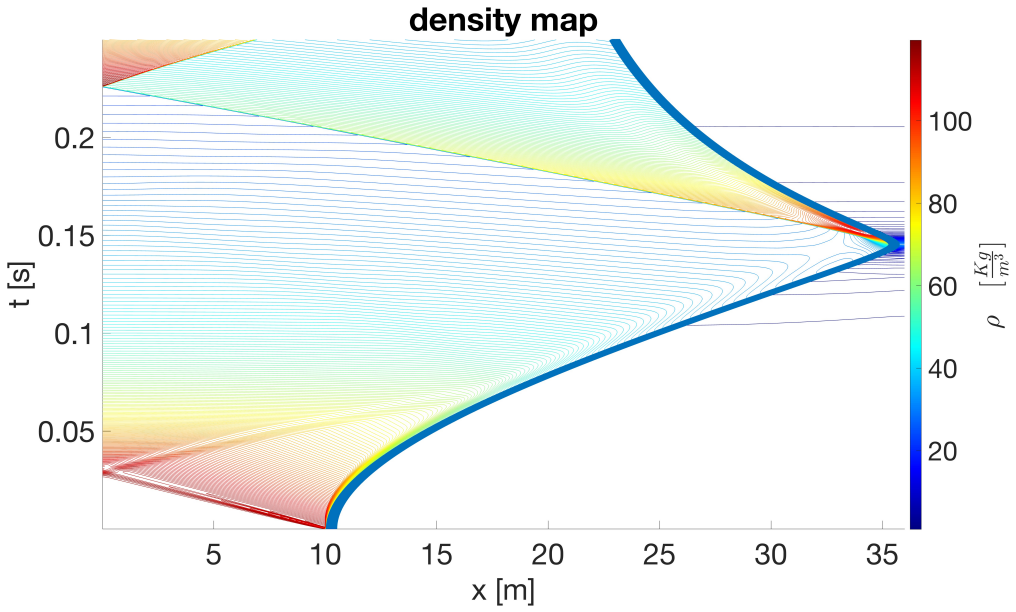


Figure 5.7: Contour map of the density

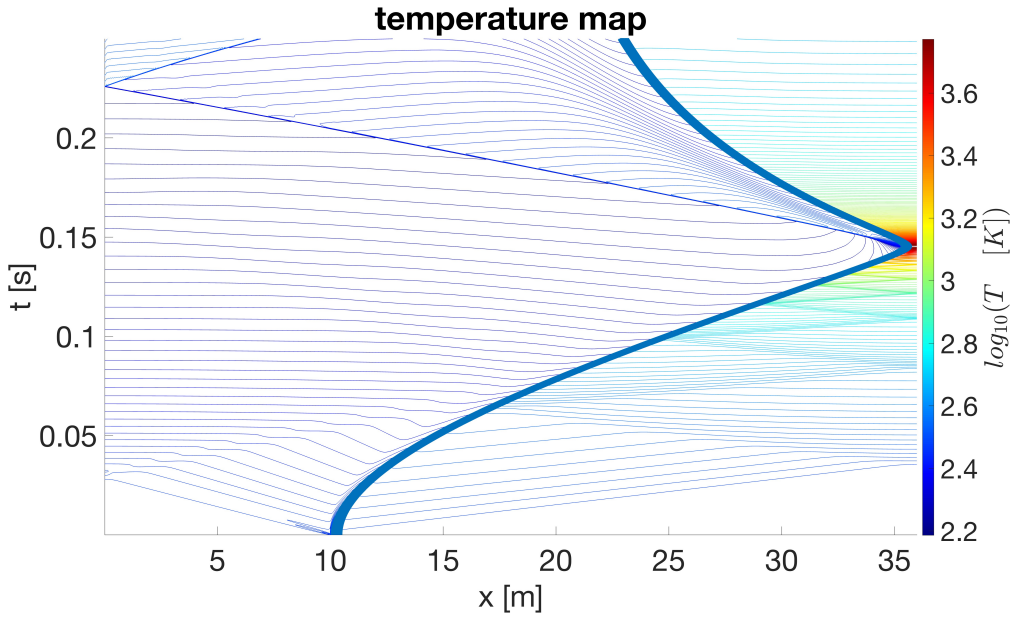


Figure 5.8: Contour map of the base-10 logarithm of temperature

In Figures 5.6,5.7 and 5.8, pressure, density, and temperature isolevel maps of the time history solution are presented, with the piston track highlighted by a solid blue line. It is visible how the pressure differential across the piston, shown in Figure 5.9, causes the piston to move and accelerate. As the piston moves forward, the gas behind it expands, while the gas in front of it compresses. Since the piston is moving in subsonic conditions, compression waves are generated, which reach the left wall and reflect back as expansion waves, and vice versa on the right side. These waves are visible in the contour maps, as well as in Figure 5.9 and 5.11, which shows the pressure on the left and right walls of the tunnel. At a certain point, the gas on the right side reaches a pressure condition that causes the piston to decelerate until the pressure differential becomes zero, after which the piston begins to accelerate in the opposite direction.

To have a complete overview on the piston cinematic the pisto track and the velocity are plotted in figrure 5.10.

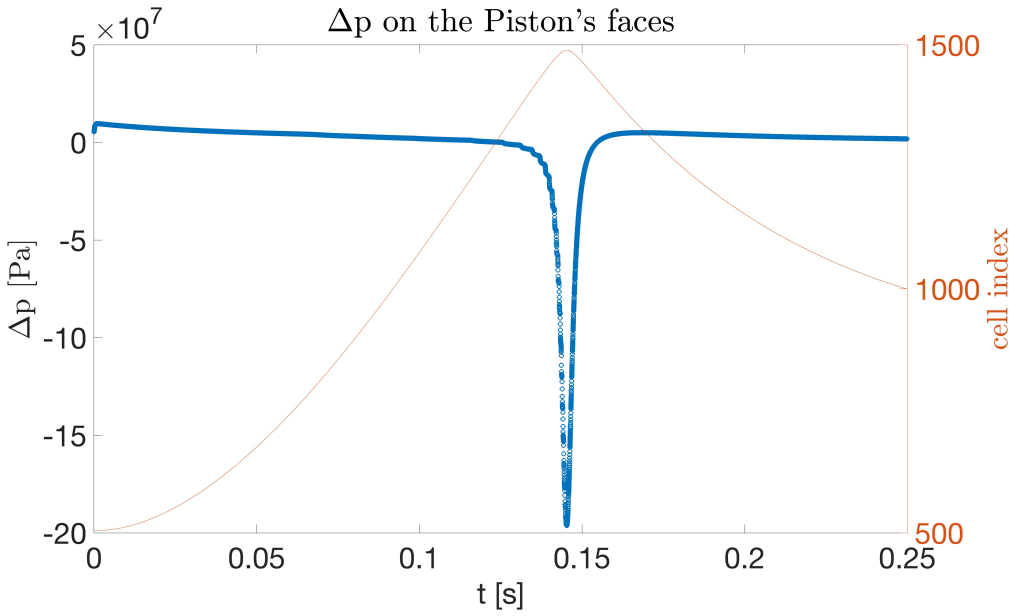


Figure 5.9: Pressure differential on the piston faces

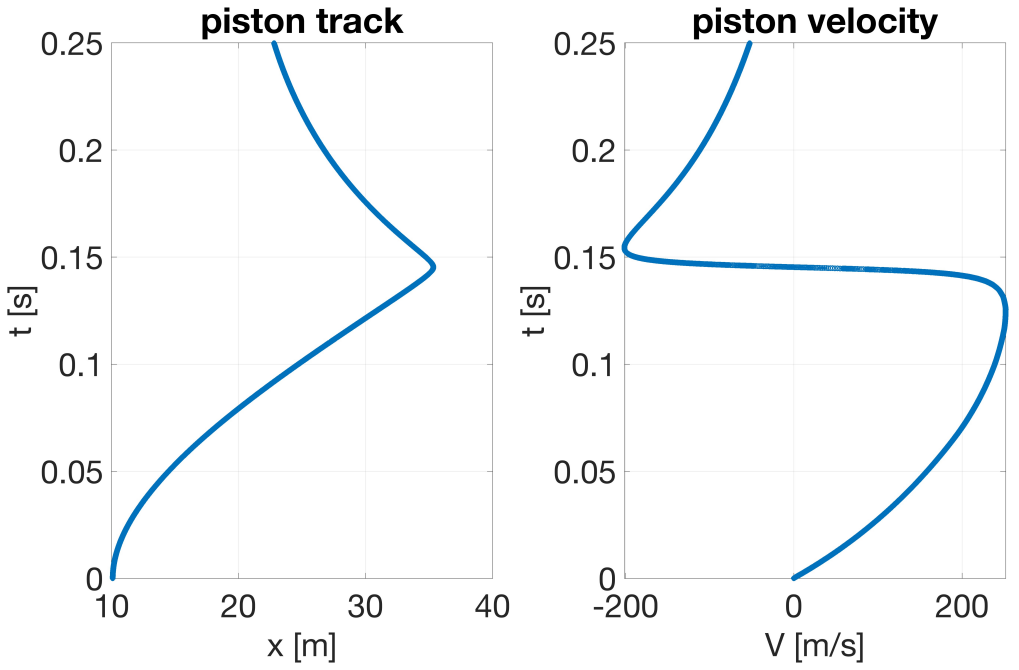


Figure 5.10: Piston cinematic

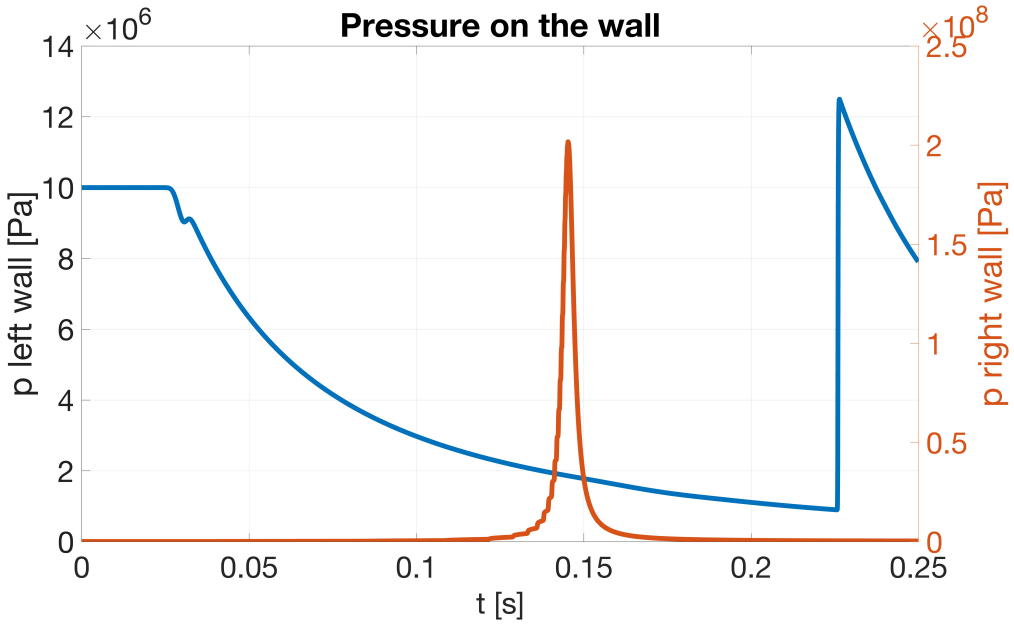


Figure 5.11: Pressure on the walls

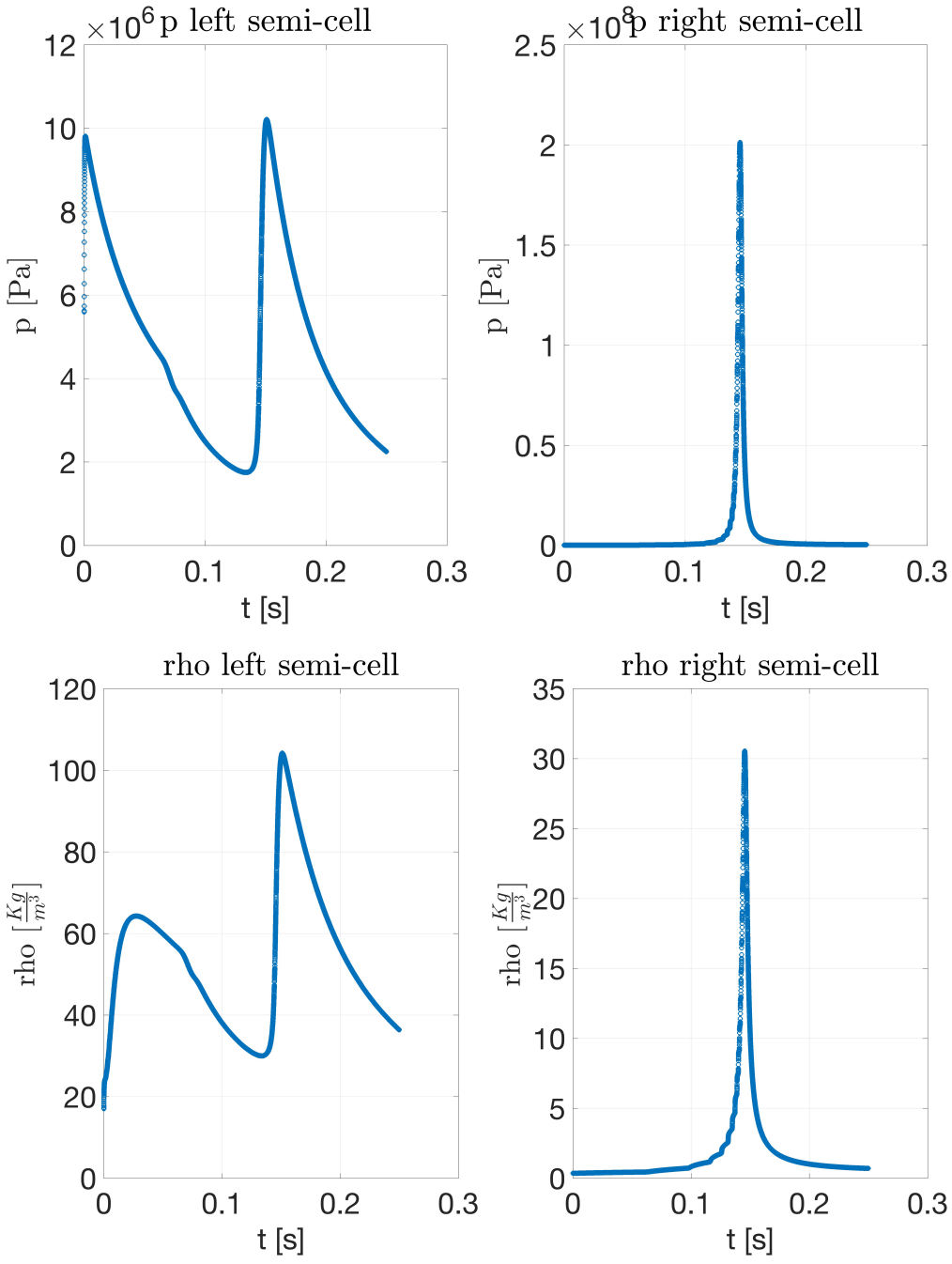


Figure 5.12: Pressure and density on the piston faces

Chapter 6

PLATO: PLAsmas in Thermodynamic nOn-equilibrium [1]

PLATO (PLAsmas in Thermodynamic nOn-equilibrium) is a high-fidelity library for the thermodynamics, transport, kinetics, and radiation of equilibrium and non-equilibrium multi-component plasmas. Its source files are implemented in Fortran 2003/2008 but can also be used in C and C++ codes, as is the case in this work.

6.1 Library Description

The PLATO library is designed for local thermodynamic equilibrium/non-equilibrium multi-component plasmas. It is intended to account for these local phenomena in more complex codes, eliminating the need to implement the physics related to non-equilibrium kinetic theory for dilute gases. This library can be used to handle the source terms in the transport equations.

PLATO is highly versatile and offers a wide range of physical complexities, from legacy multi-temperature models to grouping and/or collisional-radiative models. It was developed and is widely used at the Center for Hypersonics and Entry Systems Studies (CHESS) at the University of Illinois [63, 64].

Some of the built-in applications in the PLATO library include:

- box: This program simulates an isochoric, isothermal, or adiabatic chemical reactor.
- shock: This program computes the inviscid non-equilibrium flow behind a

normal shock wave.

- lagr: This program solves the governing equations in Lagrangian form.

Standalone examples are provided in Section 6.2.

In the HYQ1D code, the library can be used to calculate the chemical source terms in the species conservation equations 1.1.

6.2 PLATO Stand-Alone Test

In this section, several stand-alone simulations are presented to demonstrate some of the possible uses of PLATO.

6.2.1 Shock Test

Here, the PLATO shock application is tested. Two specific cases are shown, referred to as the MAC 5.55 and MAC 10.15 conditions. These are candidate conditions for wind tunnel comparison at the University of Queensland for the T4 reflected shock tunnel and the X3 expansion tunnel [65]. Figure 6.1 displays the test case setup for a typical PLATO input file.

```
PATH     =/Users/luigi//Desktop/MSC_Purdue/PLATO/database
MIXTURE  = air5
REACTION = air5
TRANSFER = TTv

#--------------------------
#Free-strem conditions (pressure[Pa], temperature[K], velocity[m/s] and mole fractions)
P_OO  = 13400.0d0 #2238.3d0
T_OO  = 1360.d0 #492.d0
U_OO  = 4000.d0 #4501.2d0
COMP_OO = O2 N2
XC_OO   = 0.21d0 0.79d0

#--------------------------
#Space-grid parameters
XS = 0.d0
XE = 0.1d0
DX_MAX = 5.d-3
DX_MIN = 1.d-14
SF = 1.01d0

#--------------------------
#ODE solver settings
ATOL = 1.d-12
RTOL = 1.d-9
```

- INPUT 1 (M=5.55)
- INPUT 2 (M=10.15)

Figure 6.1: Shock case simulation setup in a typical PLATO input file

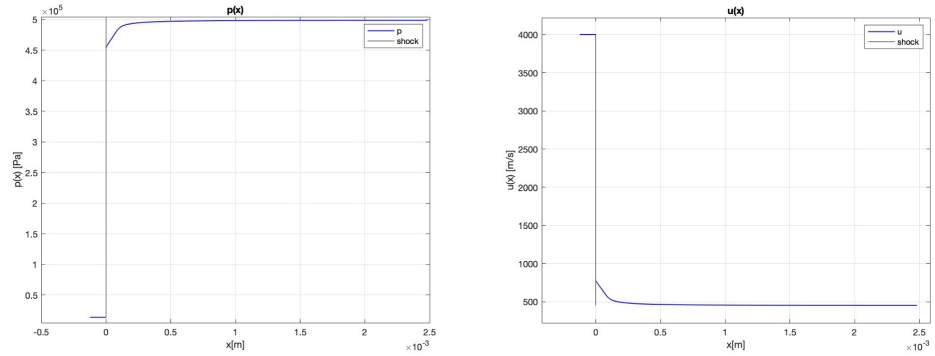


Figure 6.2: Pressure and density plots for the M5.55 condition

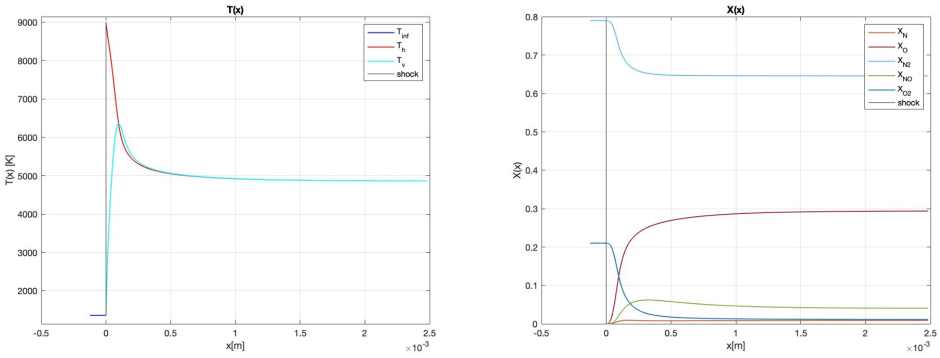


Figure 6.3: Temperature and molar fraction plots for the M5.55 condition

In Figure 6.2, the pressure and density solutions for the shock case are shown. The classical shock jump is visible, and following this jump, further changes in the fluid dynamic variables occur due to the advancement of chemical reactions triggered by the high temperature and non-equilibrium phenomena caused by the instantaneous translational-rotational temperature change. In Figure 6.3, the temperatures and molar fractions are displayed. For the temperature, the classical post-shock relaxation behavior is observed: the translational-rotational temperature changes instantaneously and then relaxes towards the vibrational temperature, transferring energy to this degree of freedom until they converge, leading to a new thermodynamic equilibrium. For the molar fractions, the typical non-equilibrium chemical phenomena are captured, with significant dissociation occurring, which reaches equilibrium not instantaneously but after some distance.

For the MAC 10.15 case, the qualitative trends shown in Figures 6.4 and 6.5 are the same as those for the MAC 5.55 case, but they are amplified due to the

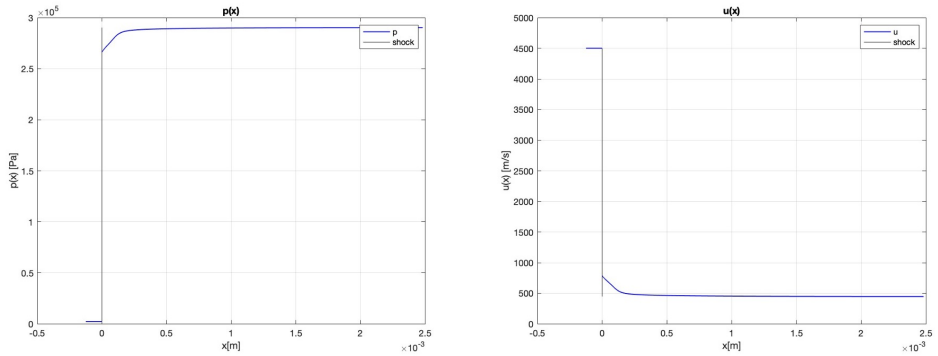


Figure 6.4: Pressure and density plots for the M10.15 condition

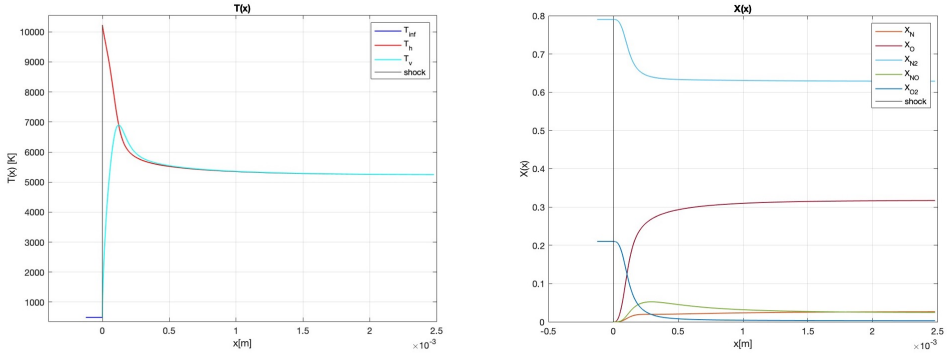


Figure 6.5: Temperature and molar fraction plots for the M10.15 condition

higher Mach number. This leads to a stronger shock and more severe post-shock conditions and phenomena.

6.2.2 Box test

Here, several tests using PLATO's box application are reproduced. Specifically, two different tests are performed: the first with the gas temperature equal to the chemical bath temperature, and the second with the two temperatures initialized differently, with the latter being higher than the former. By comparing these two cases, the influence of thermally non-equilibrium phenomena, present only in the second case, is highlighted. The results for the first case are shown on the left side of the following figures, and the results for the second case are shown on the right. Figure 6.6 provides the simulation settings for the box test, illustrating a typical PLATO input file.

```
#----------------------------
# Path to data and mixture, reaction, transfer files
PATH = /Users/luigi/Desktop/MSC_Purdue/PLATO/database
MIXTURE = air5
REACTION = air5
TRANSFER = TTv

#----------------------------
#Initial conditions (pressure temperature and mole fractions)
P0 = 100000
T0 = 10000 = T_t0 = T_v0
COMP0 = N  O  N2  NO  O2
XC0 = 0 0 0.79 0 0.21

#Bath temperature
T_BATH = 10000 = T_0
#----------------------------
#Time-grid parameters
TS = 0
TE = 0.001
DT_MAX = 1e-06
DT_MIN = 1e-14
SF = 1.01

#----------------------------
#ODE solver settings
ATOL = 1e-12
RTOL = 1e-09
```

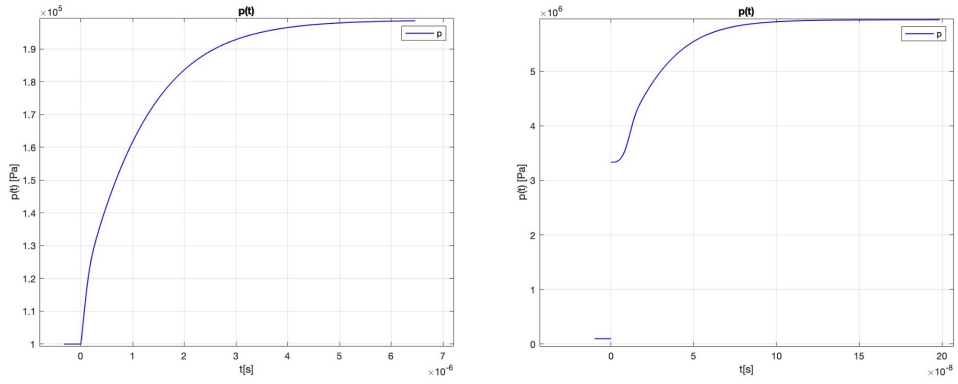
Box app (T_bath = T_0)

```
#----------------------------
# Path to data and mixture, reaction, transfer files
PATH = /Users/luigi/Desktop/MSC_Purdue/PLATO/database
MIXTURE = air5
REACTION = air5
TRANSFER = TTv

#----------------------------
#Initial conditions (pressure temperature and mole fractions)
P0 = 100000
T0 = 300 = T_t0 = T_v0
COMP0 = N  O  N2  NO  O2
XC0 = 0 0 0.79 0 0.21

#Bath temperature
T_BATH = 10000
#----------------------------
#Time-grid parameters
TS = 0
TE = 0.001
DT_MAX = 1e-06
DT_MIN = 1e-14
SF = 1.01

#----------------------------
#ODE solver settings
ATOL = 1e-12
RTOL = 1e-09
```

Box app (T_bath ≠ T_0)

(a) $T_{bath} = T_0$ **(b)** $T_{bath} \neq T_0$

Figure 6.6: Isocoric case simulation setup in a typical PLATO input file

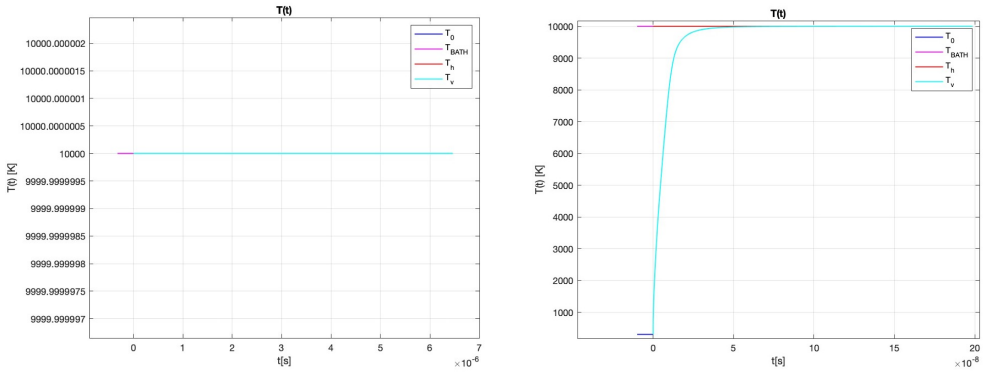


(a) $T_{bath} = T_0$ **(b)** $T_{bath} \neq T_0$

Figure 6.7: Pressure plot comparison

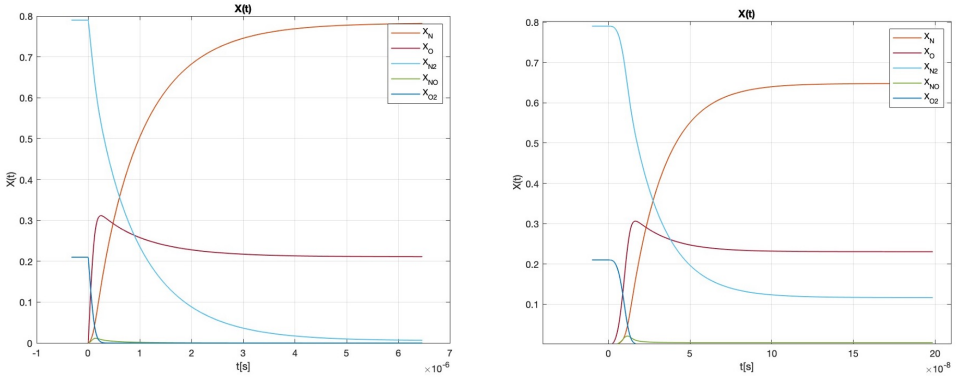
In Figure 6.7 and 6.8, the pressure and temperature comparisons for the two cases are provided. In the second case, where the bath's temperature differs from the initial system temperature, there is a noticeable jump in pressure and translational-rotational temperature, with the vibrational temperature relaxing after a few fractions of a second.

In Figure 6.9, the molar fraction comparisons are presented. It is evident from the time axis that chemical reactions are significantly enhanced in the second case, as the bath temperature transfers energy to the system.



(a) $T_{bath} = T_0$ **(b)** $T_{bath} \neq T_0$

Figure 6.8: Temperature plot comparison



(a) $T_{bath} = T_0$ **(b)** $T_{bath} \neq T_0$

Figure 6.9: Molar fraction plot comparison

Chapter 7

HYQ1D: Software architecture

This section provides a detailed overview of the HYQ1D architecture and code organization. HYQ1D was written in C++ and extensively tested using GCC[66] compilers on macOS[67] and Linux[68] operating systems.

The source code is organized into different routines, which are properly combined based on user settings. The code's outermost routines consist of seven main components that execute the fundamental steps of a CFD code.

- time_integration_managing
- inspections
- fluid_dynamic_variable_calculation
- Δt_calculation
- source_term_calculation
- spatial_reconstruction
- flux_evaluation
- time_history_storage

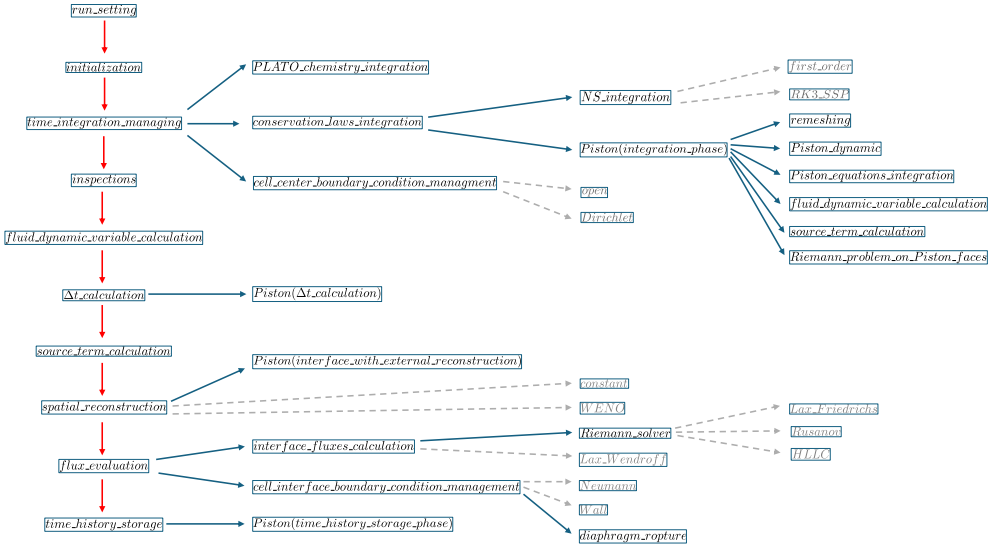


Figure 7.1: Code organization

In Figure 7.1, the code organization is shown. The red arrows represent the integration procedure, the blue arrows indicate the links between subroutines, and the gray dashed arrows represent some notable algorithms within the routines.

7.1 time_integration_managing

This routine manages the time integration throughout the code and is based on a Strang-Splitting approach, analyzed in Section 4.1.3, which can be extremely useful when dealing with PDEs containing stiff source terms, such as in chemically reacting fluids. This method allows for the separate integration of the non-stiff and stiff parts of the PDE, providing significant computational cost relief in explicit codes.

A flag, configurable in the settings file, can activate or deactivate the Strang-Splitting logic. In the latter case, the PDE integration logic is reduced to the (non-stiff) routine integration one. The `time_integration_managing` routine arranges the integration using the following routines:

- PLATO_chemistry_integration
- conservation_laws_integration
- cell_center_boundary_condition_managment

7.1.1 PLATO_chemistry_integration

This routine links HYQ1D with the applications and libraries of the PLATO software, which are extensively described in Chapter 6, using the Strang-Splitting logic. The PLATO software routines handle the integration of chemical source terms.

After executing the PLATO source code, re-initialization of the fluid-dynamic and integration variables is necessary to ensure compatibility with the HYQ1D code.

7.1.2 conservation_laws_integration

In this routine, the integration of the *non-stiff* part of the conservation equations (in the case of Strang-Splitting) or the entire conservation law (in other cases) is carried out.

The routine is subdivided into two subroutines: the first manages the integration of the conservation laws on the regular mesh, while the second, activated when the *free piston model* is enabled (Chapter 5), calls the piston routine, which performs the integration on the sub-cell mesh using the modified conservation laws presented in Section 5.2.1.

The user can set the time integration discretization to either first-order, using a constant time integration (Section 4.1.1), or third-order, employing a Runge-Kutta 3 Strong Stability Preserving algorithm (Section 4.1.2). In both cases, the time integration is explicit.

7.1.3 cell_center_boundary_condition_managment

In this routine, all boundary conditions imposed through cell center values are managed. Some examples include:

- `open`: Links two adjacent sections.
- `Dirichlet`: Specific conditions are imposed at the cell center.

7.2 inspections

This routine has two main objectives:

- Inspect the integration variables vector for potential code blow-up. If detected, an error flag is activated to abort the code execution.

- Calculate the difference between the current time step and the previous one using the L^2 norm of the difference between actual and precedent value of the integration variables vector, checking for solution convergence (useful when seeking a steady-state solution).

7.3 fluid_dynamic_variable_calculation

In this routine, all calculations of the relevant fluid-dynamic variables are performed as showed in Section 3.4.

7.4 Δt_calculation

In this routine, the estimation of Δt is performed. In an explicit time integration scheme, this is a fundamental step that must be handled carefully to ensure numerical stability during time advancement (Section 3.5).

The Δt management routine also calls the piston routine to account for theΔt limitations imposed by the integration specified in the *free piston model*.

7.5 source_term_calculation

In this routine, all source terms, depending on the specific physical model selected for the given section, are calculated. These include chemical, viscous loss, and relaxation source terms (Section 3.10).

7.6 spatial_reconstruction

In this routine, spatial reconstruction is performed by calling the selected scheme based on the user's settings. More details on the implemented reconstruction schemes can be found in Section 4.2.

7.7 flux_evaluation

In a finite volume discretization, the focus is on how to calculate the fluxes at the cell interfaces, and this routine addresses that requirement.

This routine consists of two parts: `interface_fluxes_calculation` and `cell_interface_boundary_condition_management` .

7.7.1 interface_fluxes_calculation

Based on the settings used and the different numerical methods implemented, the routine can utilize the Lax-Wendroff scheme (Section 4.3.1) or resolve the Riemann problem at the interface using one of the implemented Riemann solvers: HLLC (Section 4.3.2), Rusanov scheme (Section 4.3.2), or Lax-Friedrichs scheme (Section 4.3.2).

7.7.2 cell_interface_boundary_condition_management

In this section, all boundary conditions that involve fluxes or the imposition of interface variables are established. Some of these boundary conditions may include:

- `Wall`: Simulates the presence of a wall at the domain's boundary.
- `Neumann`: Imposes a specific flux value.

The routine also manages diaphragm rupture based on the wall value calculated for the wall boundary condition.

7.8 time_history_storage

To optimize memory management, the code stores only the current time step variables and a few variables from the previous step. This part of the code is designed to store the variables at a specified frequency during time integration in a file, providing a time history of the stored data.

Chapter 8

HYQ1D: Numerical results

In this section, several tests and numerical results obtained with HYQ1D will be presented. These tests will assess the accuracy of the CFD code in comparison to the exact solution for an ideal model case. Additionally, some test cases will be modified to reflect hypersonic conditions, allowing for the evaluation of the complete non-equilibrium model.

8.1 Shock Tube Tests

The accuracy and performance of the various numerical models implemented in the code, as described in Section 4, will be tested and compared with exact solutions. Five standard shock tube problems, taken from [69], are considered. Table 8.1 lists the fluid dynamic conditions on the left and right sides of the initial discontinuity, the position of the discontinuity, and the time of observation and plotting.

For each test, the left-hand plot will present a first-order approximation in space and time, while the right-hand plot will display results using RK3-SSP for time integration and WENO for spatial reconstruction. When convergent, the Lax-Wendroff method, being a second-order approximation in both space and time, will be included in both figures.

The spatial discretization consists of 100 cells, and the CFL number is taken as large as possible for each case.

8.1.1 TEST 1

This test is a modified version of the Sod shock tube problem. Its solution consists of a right-moving shock wave, a right-moving contact discontinuity, and a left-moving sonic rarefaction wave.

Test	ρ_L	u_L	p_L	ρ_R	u_R	p_R	x_0	t^*
1	1.0	0.75	1.0	0.125	0.0	0.1	0.3	0.2
2	1.0	-2.0	0.4	1.0	2.0	0.4	0.5	0.15
3	1.0	0.0	1000.0	1.0	0.0	0.01	0.5	0.012
4	5.99924	19.5975	460.894	5.999242	-6.19633	46.0950	0.4	0.035
5	1.0	-19.59745	1000.0	1.0	-19.59745	0.01	0.8	0.012

Table 8.1: Test cases with initial conditions for ρ, u, and p on the left and right states.

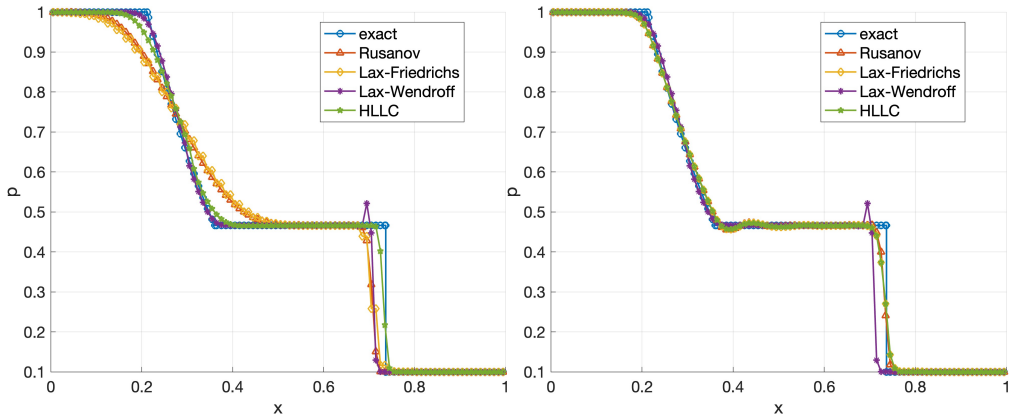


Figure 8.1: Pressure plots, 1° order vs 5° order reconstruction

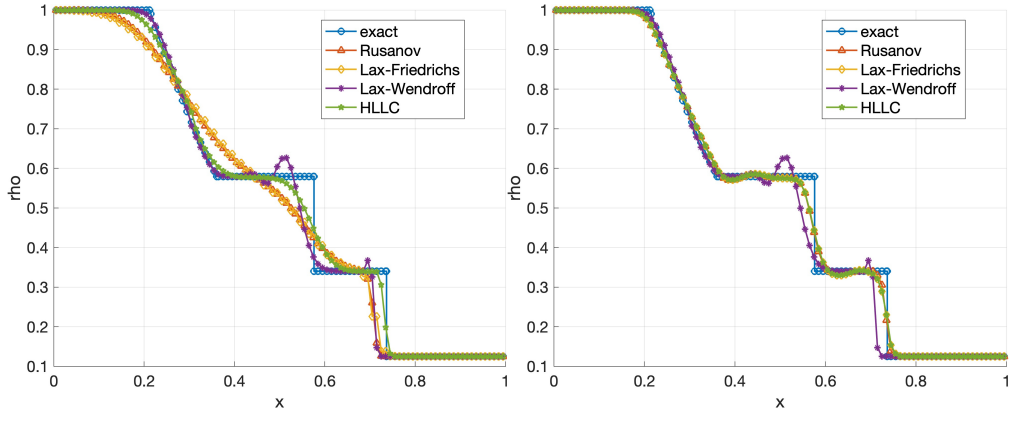


Figure 8.2: Density plots, 1° order vs 5° order reconstruction

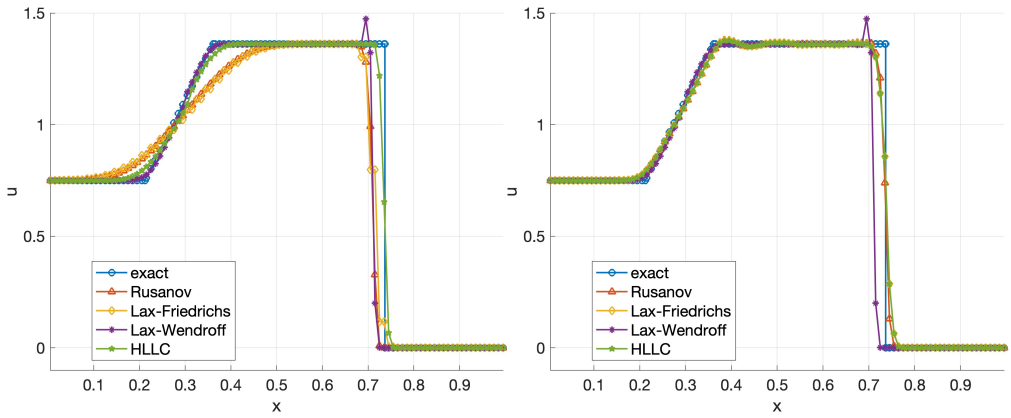


Figure 8.3: Velocity plots, 1° order vs 5° order reconstruction

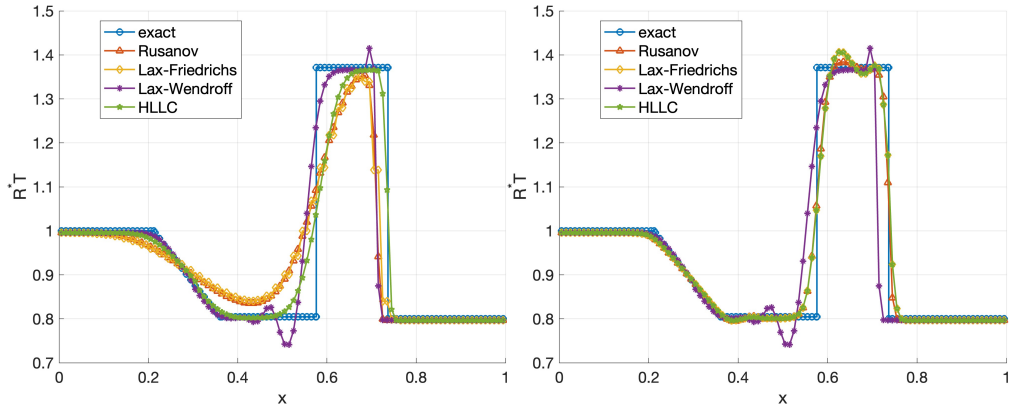


Figure 8.4: R^*T plots, 1° order vs 5° order reconstruction

From the plots presented in this section, it is evident that the accuracy of the solution improves with the accuracy of the Riemann solver for first order solution. This is particularly noticeable when the HLLC solver is used, as it significantly enhances the solution compared to other cases. When high order resolution is used (RK3-SSP + WENO) all the Riemann problem tend to converge on the exact solution, we can say that this high order reconstruction is the sensibility to the used Rieamnn solver decrease. While the Lax-Wendroff method is more accurate than all first-order approximations, it still exhibits some oscillatory behavior typical of these schemes, as well as of high-order non monotonic approximations.

In general, it is visible how high-order schemes, such as RK3-SSP combined with WENO, and the Lax-Wendroff method (when convergent) have a solution closer to the exact one, with the first better than the latter. The shock wave is consistently

well-captured, though slightly anticipated in some first-order approximations. It should be noted that the accuracy of WENO, which is fifth order accurate on smooth solutions can degrade in the presence of discontinuities to third order.

These observations apply to all five tests, though more detailed analysis is provided for this particular case. Notably, the differences between high-order and first-order schemes are most visible on expansion and contact surface, with the latter which is typically the most challenging discontinuity to capture, as shown in Figure 8.3.

8.1.2 TEST 2

Test 2 consists of two symmetric rarefaction waves and a trivial contact discontinuity. The region between the left and right waves approaches vacuum, making this problem an effective test for evaluating the performance of numerical methods in low-density flows.

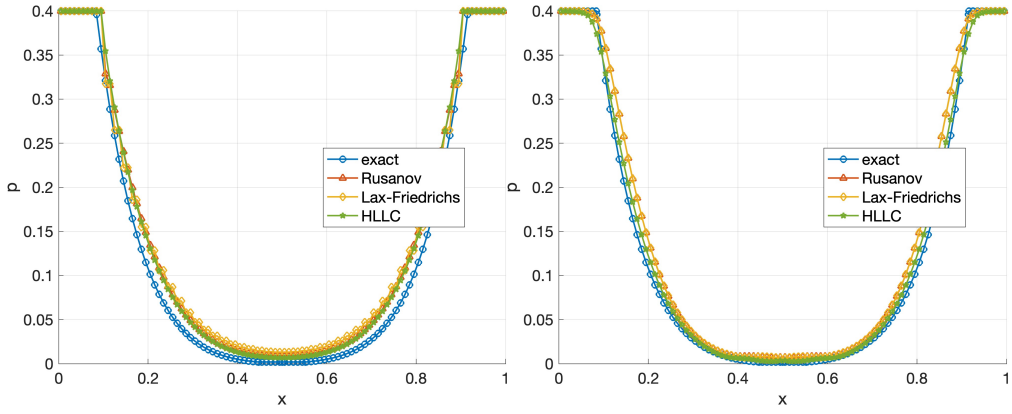


Figure 8.5: Pressure plots, 1° order vs 5° order reconstruction

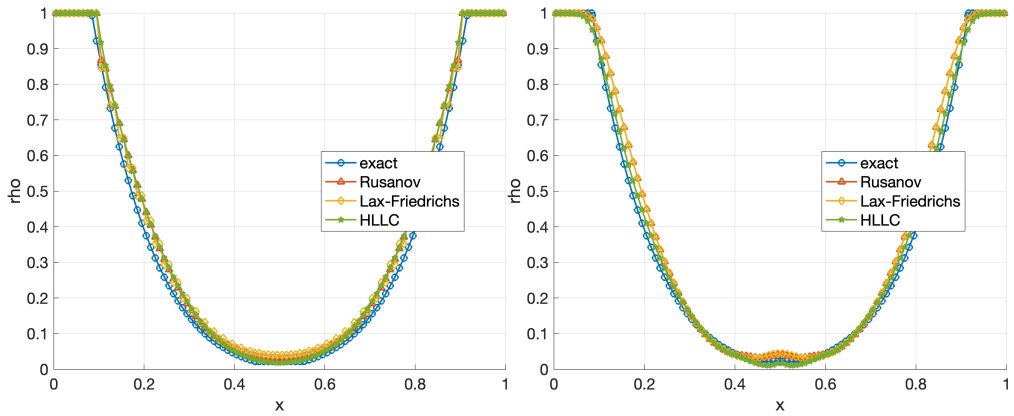


Figure 8.6: Density plots, 1° order vs 5° order reconstruction

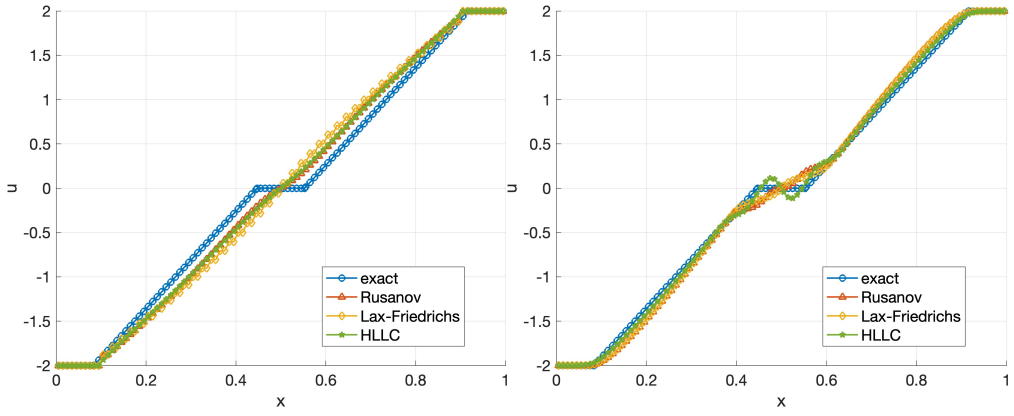


Figure 8.7: Velocity plots, 1° order vs 5° order reconstruction

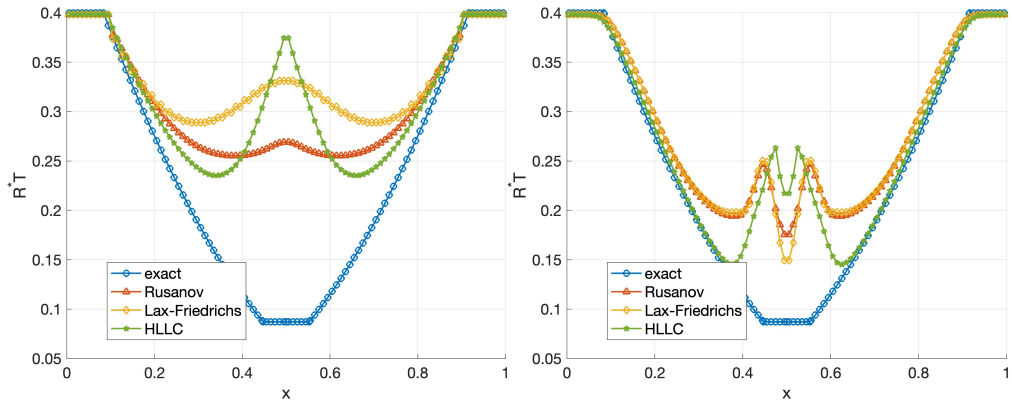


Figure 8.8: R^*T plots, 1° order vs 5° order reconstruction

This test is the most challenging for all the numerical schemes, as seen in the temperature plot, which highlights their limitations in this case. Neither the higher order solution, laso if it iss better, reach a good approximation of the exact solution. The Lax-Wendroff scheme fails to connverge, even with a low CFL number, likely due to its lack of monotonicity. In contrast to the other tests, the Rusanov Riemann solver provides the most accurate solution in this case for first order solution.

8.1.3 TEST 3

This test is designed to assess the robustness and accuracy of numerical methods. Its solution consists of a strong shock wave with a Mach number of 198, a contact surface, and a left-moving rarefaction wave.

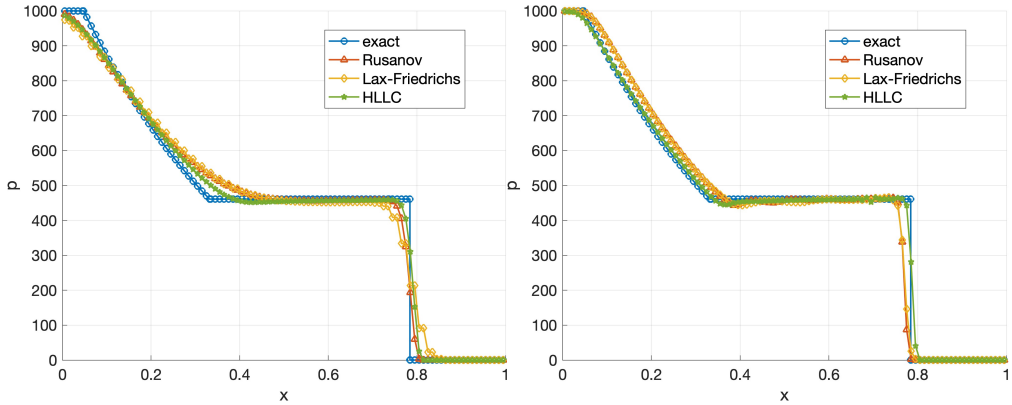


Figure 8.9: Pressure plots, 1° order vs 5° order reconstruction

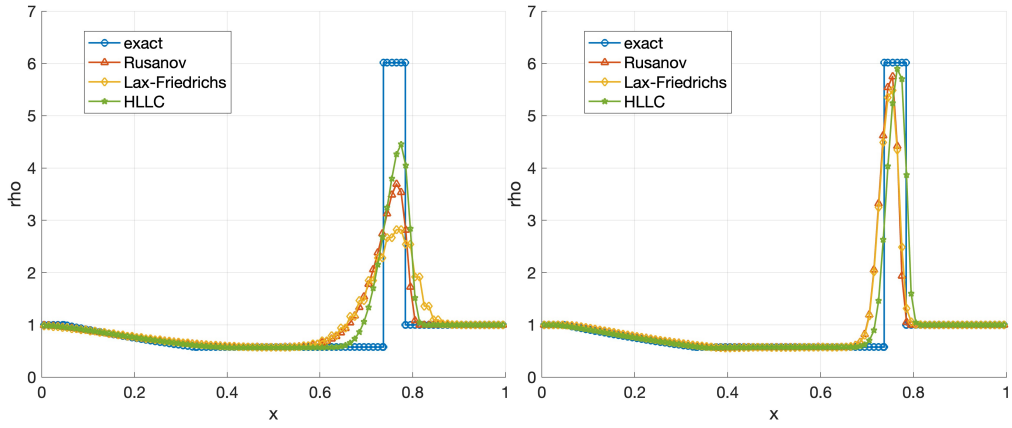


Figure 8.10: Density plots, 1° order vs 5° order reconstruction

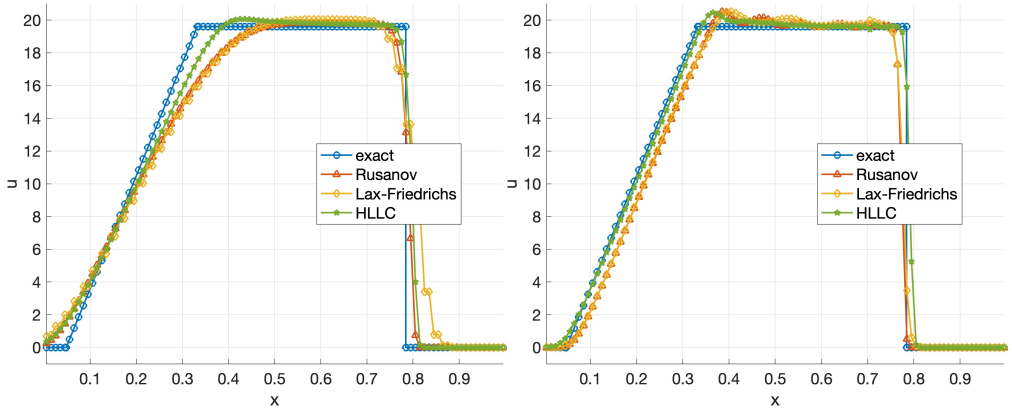


Figure 8.11: Velocity plots, 1° order vs 5° order reconstruction

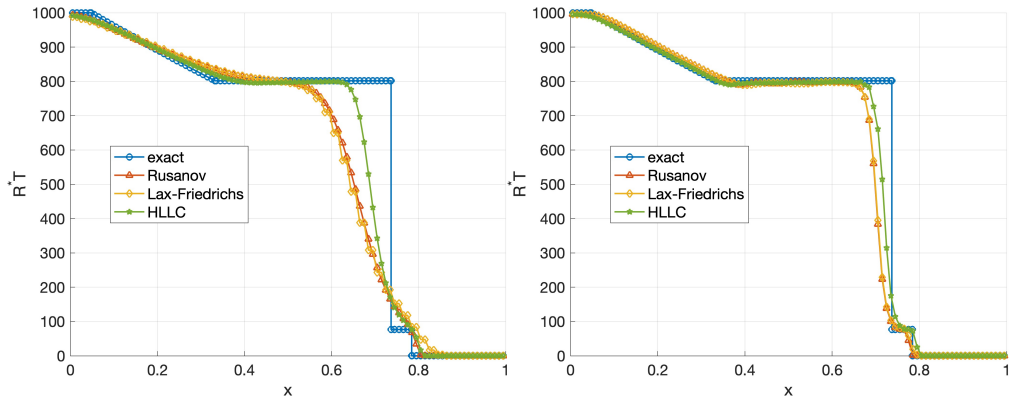


Figure 8.12: R^*T plots, 1° order vs 5° order reconstruction

The plots show that, in general, higher-order solutions capture the contact surface and rarefaction wave more accurately, as seen in Figure 8.9. The Lax-Wendroff scheme was unstable in this test case.

8.1.4 TEST 4

This test is also highly challenging, with a solution that consists of three strong discontinuities moving to the right. From left to right, these are: a left-facing shock, a contact discontinuity, and a shock wave.

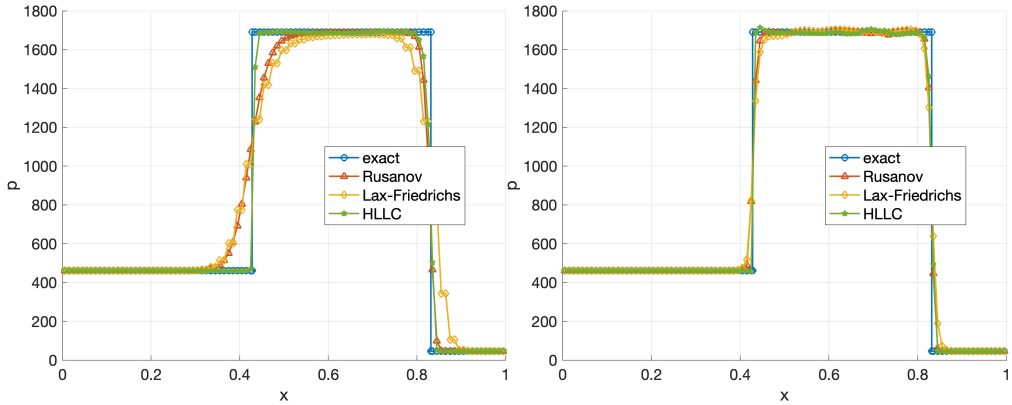


Figure 8.13: Pressure plots, 1° order vs 5° order reconstruction

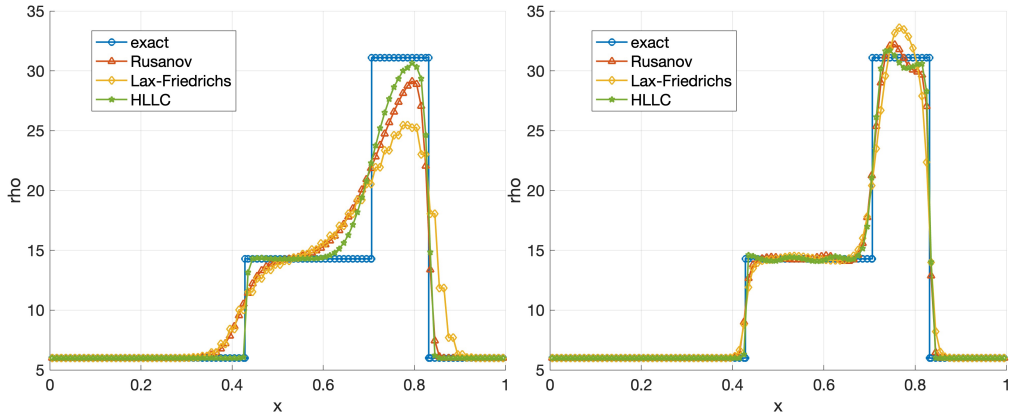


Figure 8.14: Density plots, 1° order vs 5° order reconstruction

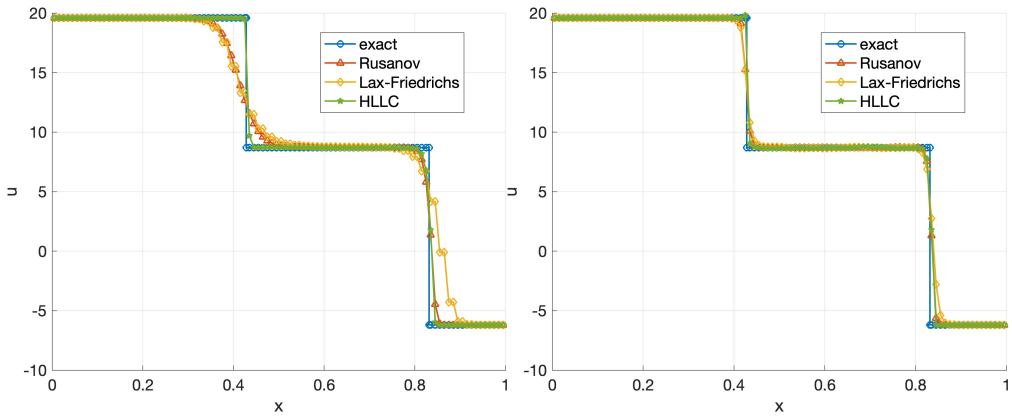


Figure 8.15: Velocity plots, 1° order vs 5° order reconstruction

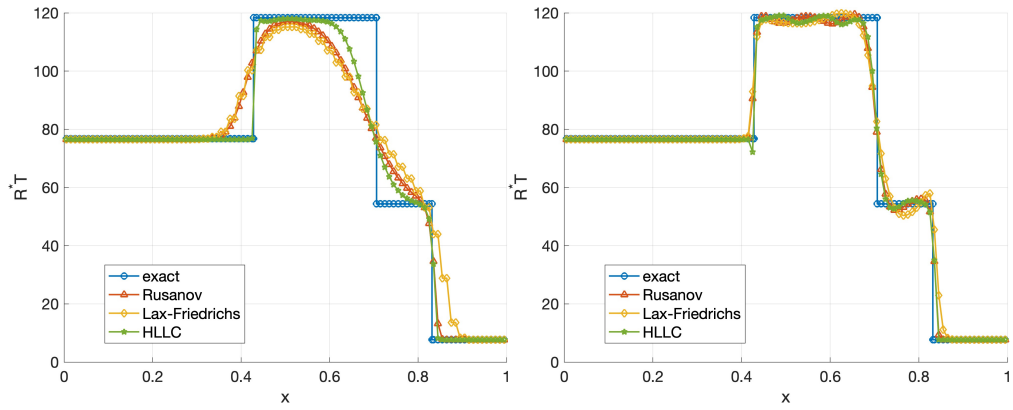


Figure 8.16: R^*T plots, 1° order vs 5° order reconstruction

Both shock waves are sharply captured by the HLLC Riemann solver, though the contact discontinuity is less well-resolved. However, higher-order solutions improve this aspect. The other solvers, in order of accuracy compared to the exact solution, are Rusanov and Lax-Friedrichs, both of which show improvement with higher-order reconstruction. The Lax-Wendroff scheme was unstable in this test case.

8.1.5 TEST 5

Test 5 is specifically designed to evaluate the robustness of numerical methods, with a primary focus on how well these methods resolve slowly moving contact discontinuities. The exact solution for this test consists of a left-moving rarefaction wave, a right-moving shock wave, and a stationary contact discontinuity.

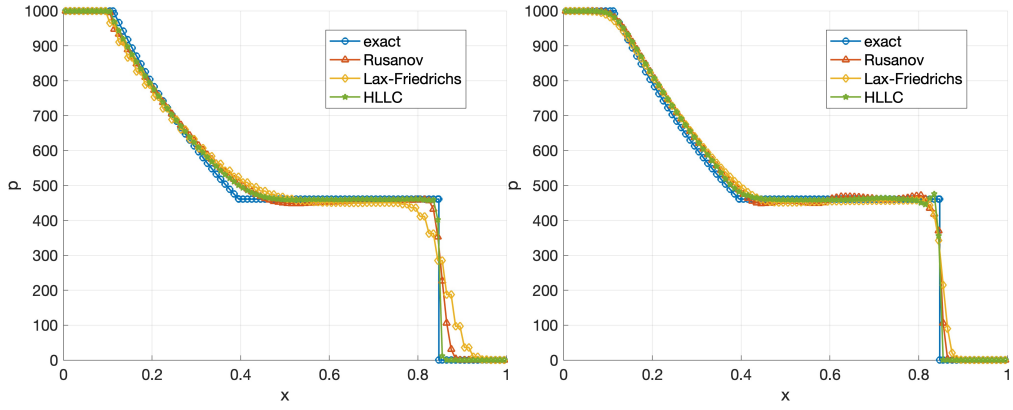


Figure 8.17: Pressure plots, 1° order vs 5° order reconstruction

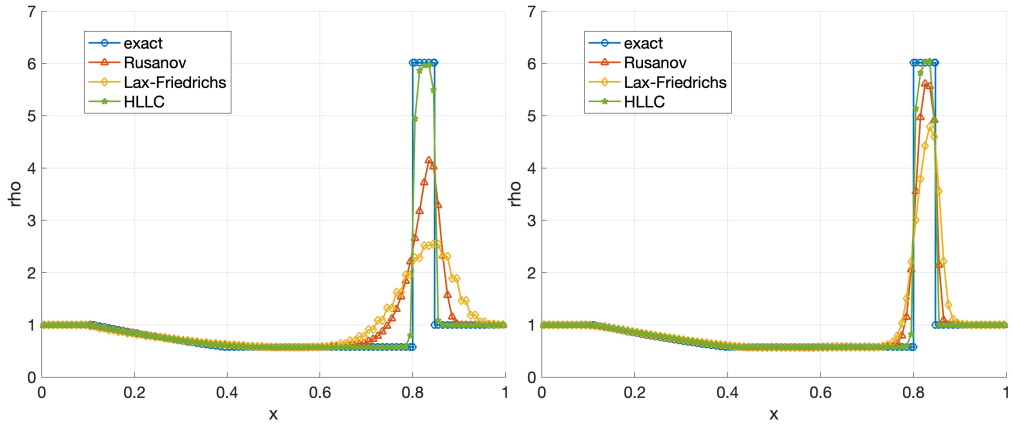


Figure 8.18: Density plots, 1° order vs 5° order reconstruction

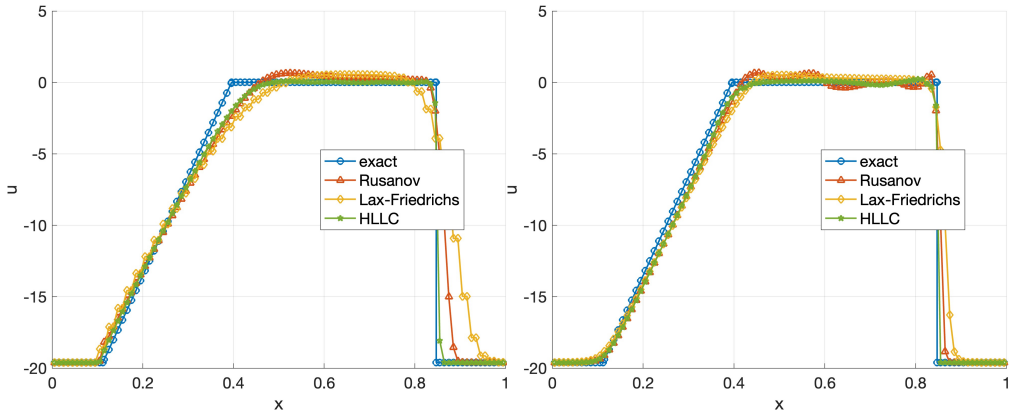


Figure 8.19: Velocity plots, 1° order vs 5° order reconstruction

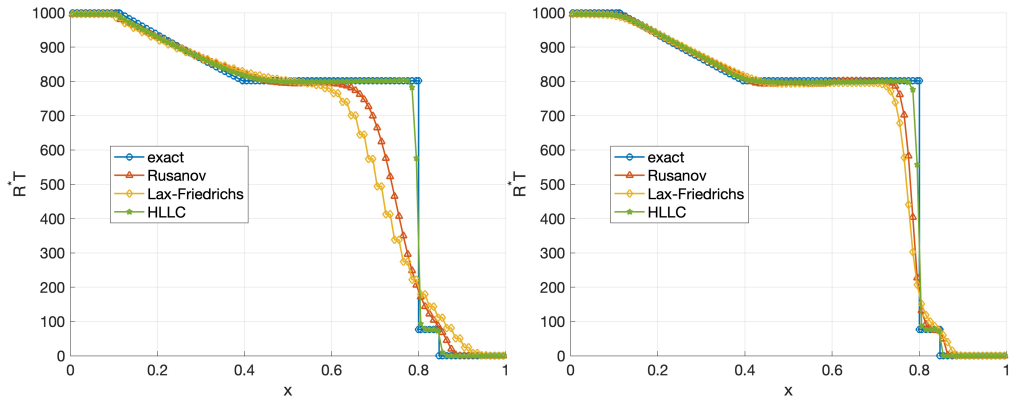


Figure 8.20: R^*T plots, 1° order vs 5° order reconstruction

From this test, we can see that the HLLC solver resolves the slow contact discontinuity quite accurately, with no major issues. However, it does slightly diffuse the starting and ending points for both orders of accuracy considered. The most challenging discontinuity to capture in this test appears to be the contact surface, for which the higher-order reconstruction exhibits great improvment respect to first order.

8.1.6 Modified TEST 1 for a standalone vibrational relaxation test

Beginning with TEST 1 in Table 8.1, the pressure was increased to achieve a sufficiently high temperature that highlights the key aspects of the thermodynamic non-equilibrium model. A modified test case was conducted with this objective.

Test	ρ_L	u_L	p_L	ρ_R	u_R	p_R	x_0	t^*
1 mod.	1.0	0.75	1000000	0.125	0.0	100000	0.4	0.00023

Table 8.2: Test case with initial conditions for ρ, u, and p on the left and right states.

The gas used in the test is air, specifically composed of 21%O_2 and 79%N_2. The chemistry is considered frozen to isolate the behavior of thermodynamic relaxation phenomena. The simulation was performed using RK3-SSP and WENO for time integration and spatial reconstruction, with HLLC as the Riemann solver. To discretize the domain, 2000 cells were employed to accurately capture the characteristic shape of the temperatures during relaxation after a shock wave. The time step Δt was estimated as described in Section 3.5.

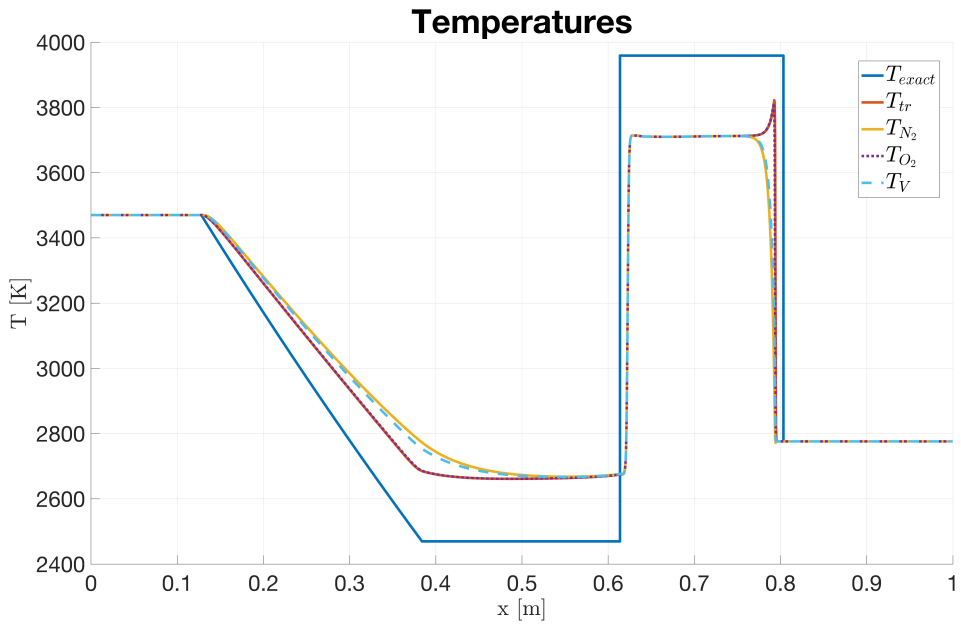


Figure 8.21: Vibrational relaxation standalone test case

In figure 8.21the exact solution for the ideal case (ideal and non-reactive gas in thermodynamic equilibrium) is plotted first, followed by the temperatures calculated in the test case. The order of the plotted temperatures is as follows: translational-rotational temperature, vibrational temperature for each diatomic species (O_2 and N_2), and finally the vibrational temperature defined from the total vibrational energy.

It is evident that the plot of the numerical solution appears rescaled compared to the ideal solution. This occurs because energy is redistributed among the different degrees of freedom and is not solely stored in the translational-rotational energy, which initially holds nearly all the energy after the shock. As the system "relaxes," it transfers a portion of the energy to the vibrational degrees of freedom, causing the related temperature to increase as equilibrium is approached.

Furthermore, it is interesting to note, beyond the post-shock phenomenology, that disequilibrium also begins during an expansion. When an expansion occurs, energy is promptly converted from internal energy to kinetic energy. In this case, the internal energy available for conversion consists of two parts: the translational-rotational part, which responds instantaneously in this model, and the vibrational part, which requires a specific "relaxation time."

8.2 Nozzle test

In this section, the area variation capability of HYQ1D is tested by comparing its results to the exact solutions for an ideal nozzle [70]. Further investigation into real gas effects is conducted on one of the subsequent cases considered. The nozzle shape analyzed is shown in Figure 8.22; it is a convergent-divergent nozzle.

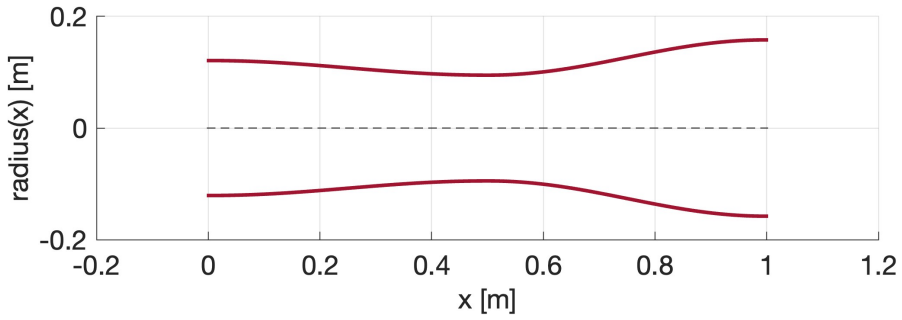


Figure 8.22: Nozzle geometry

All simulations are performed using a computational model composed of RK3-SSP for time integration, WENO for spatial reconstruction, and HLLC as the Riemann solver. The spatial domain is discretized with 200 cells, and the CFL number is set at 0.9. The gas used to initialize the solution is always air ($21\%O_2$ and $79\%N_2$).

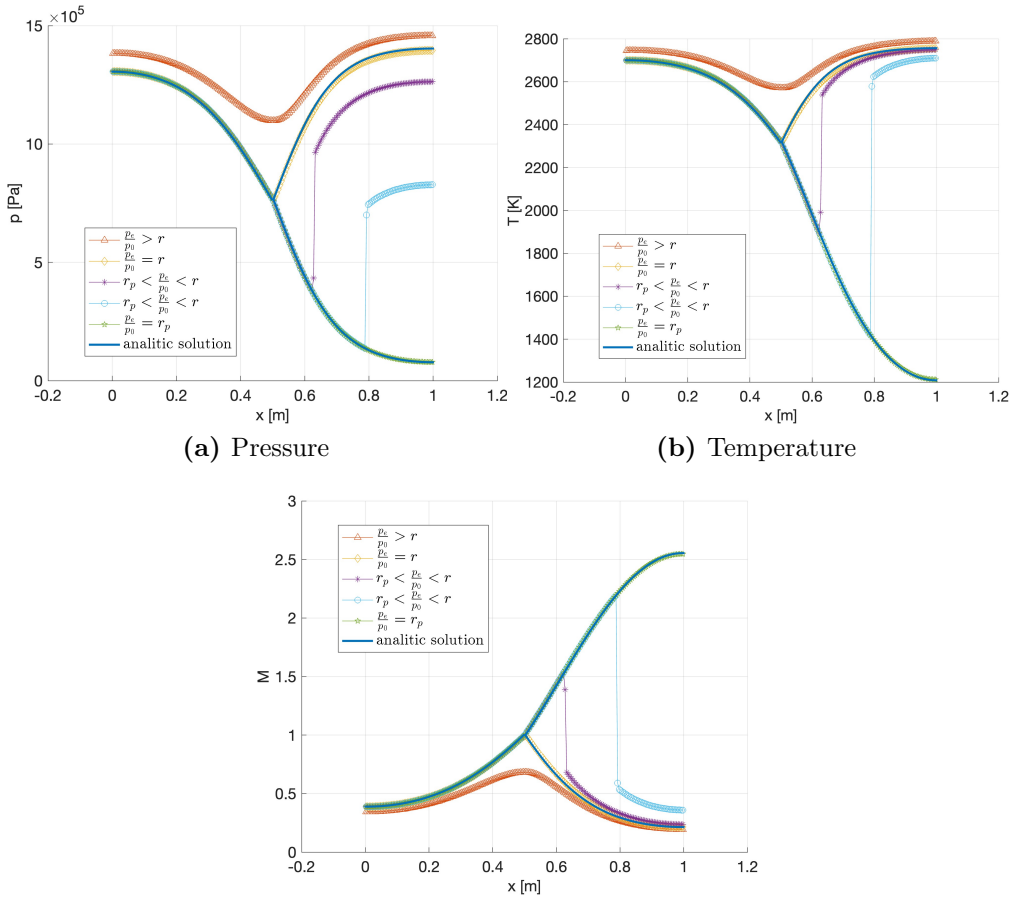
Regarding the boundary conditions, we consider two cases: subsonic inlet with supersonic outlet, or subsonic inlet with subsonic outlet. According to the characteristic theory [35], for a subsonic inlet, two conditions must be imposed between T, ρ and p; for a subsonic outlet, only one condition is necessary; and for a supersonic outlet, no conditions are needed.

8.2.1 Calorically and thermally perfect gas

Regarding the ideal gas model calculations, since a maximum of two conditions are necessary between T, ρ and p, Table 8.3 presents the case conditions for pressure and temperature imposed in the calculations as Dirichlet boundary conditions, following the same order as the legend in Figure 8.23.

Case	$p_L \quad [MPa]$	$T_L \quad [K]$	$p_R \quad [Mpa]$	$T_R \quad [K]$
1	1.300	2500	1.1032	2500
2	1.300	2500	1.0254	2500
3	1.300	2500	0.9000	2500
4	1.300	2500	0.5000	2500
5	1.300	2500	0.1000	2500

Table 8.3: Test case with conditions for p and T on the left and right boundary.



(a) Pressure **(b)** Temperature

(c) Mac number

Figure 8.23: Ideal nozzle flow solutions

In Figure 8.23, the ideal solution is plotted as a blue solid line, along with the five different test cases listed in Table 8.3, which were obtained by decreasing the ratio $\frac{p_e}{p_0}$, where p_e represents the exit pressure and p_0 represents the inlet pressure.

It is evident that cases 2 and 5 match the analytical solution perfectly; however,

the other cases also follow the theoretical trend, exhibiting a shock wave that originates in the divergent section to adapt the exit pressure to the imposed value. In the plot legend, r represents the pressure ratio that results in a perfectly adapted subsonic outlet, while r_p denotes the ratio that achieves a perfectly adapted supersonic outlet.

8.2.2 Non-ideal gas physical models

In this section, comparisons between the solutions of progressively more accurate physical models are presented. The case used is case 4 in Table 8.3, which corresponds to the case with the strongest shock in Figure 8.23.

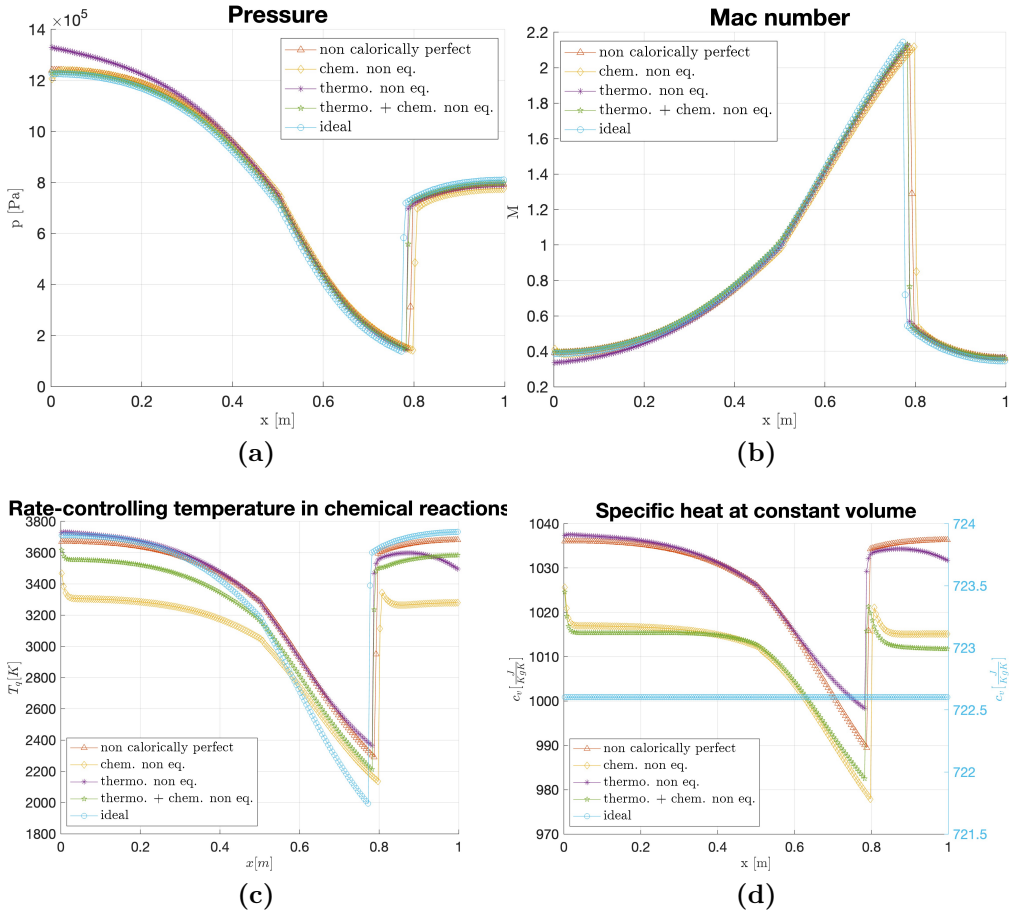


Figure 8.24: Real nozzle flow solutions

The first comparison is made in Figure 8.24 between the solutions obtained for an ideal gas, a non-calorically perfect gas ($cp(T)$), a chemically non-equilibrium

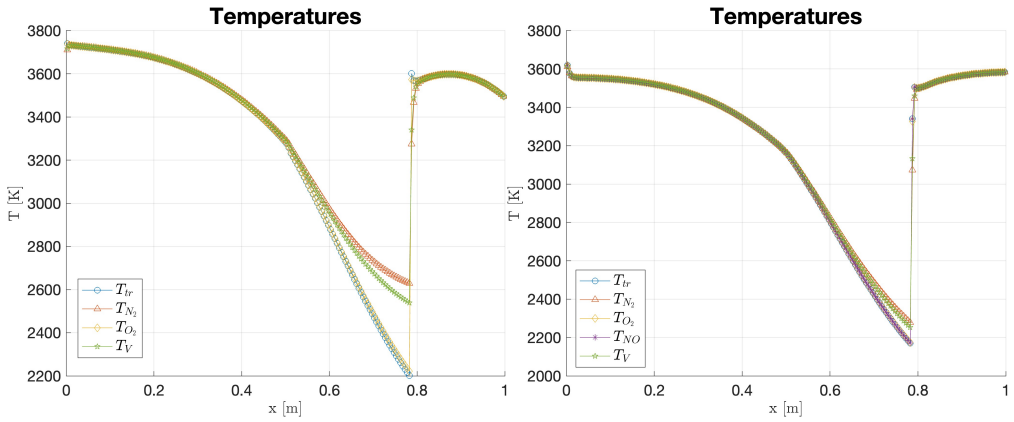
gas, a thermodynamically non-equilibrium gas, and a model incorporating both thermodynamic and chemical non-equilibrium.

Since in the thermodynamic non-equilibrium case the system is not characterized by a single temperature, the chemical rate-controlling temperature (which corresponds to the unique temperature in equilibrium and is presented in Equation 3.13) is showed here. A detailed overview of the different vibrational temperatures is provided in Figure 8.25.

It is evident that the position of the shock changes slightly due to the different gas conditions resulting from the various models used. A significant difference is shown in subfigure "d", where the specific heat at constant volume (c_v) is displayed. Given the temperature, all the models present a c_v that is completely different from that of the ideal case.

Two major distinctions can be made between the cases with chemical reactions and those with thermodynamically non-equilibrium. The presence of chemical reactions drastically decreases c_v due to the formation of O, N and NO , which are lighter and have different c_v values. This behavior is more pronounced in the case of chemical and thermodynamic non-equilibrium than in the case with only chemical reactions, as shown in Figure 8.26 thermal non equilibrium strongly enhance chemical reaction. This effect is due, in part, to a higher chemical-rate controlling temperature.

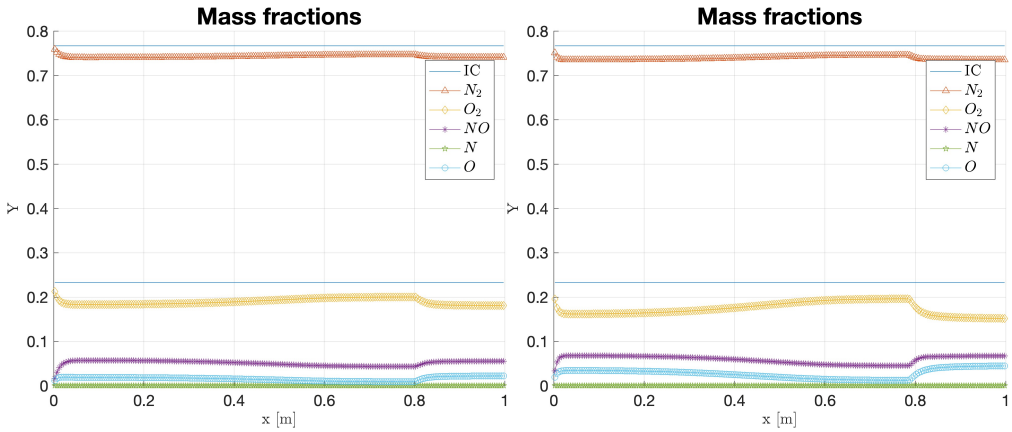
In chemically reacting gases, the temperature is lower than in the non-calorically perfect case because the chemical reactions considered are endothermic. When thermal non-equilibrium is activated, the previous mechanism is mitigated by the processes described in Section 8.1.6.



(a) thermo. non eq. only **(b)** chem. and therm. non eq.

Figure 8.25: Comparison of vibrational relaxation between frozen and non-equilibrium chemistry cases

Another important consideration can be highlighted with the comparison in Figure 8.25, which shows how, in the case of chemical non-equilibrium, the "relaxation process is mitigated" compared to thermodynamic non-equilibrium. This mitigation may be attributed to the chemical reactions, which absorb energy and result in a less pronounced change in the translational-rotational temperature. Consequently, the vibrational energies take less time to "relax".



(a) chem. non eq. only **(b)** chem. and therm. non eq.

Figure 8.26: Comparison of mass fractions between equilibrium and non-equilibrium thermodynamics cases

Chapter 9

UQ-T4 wind tunnel simulation results

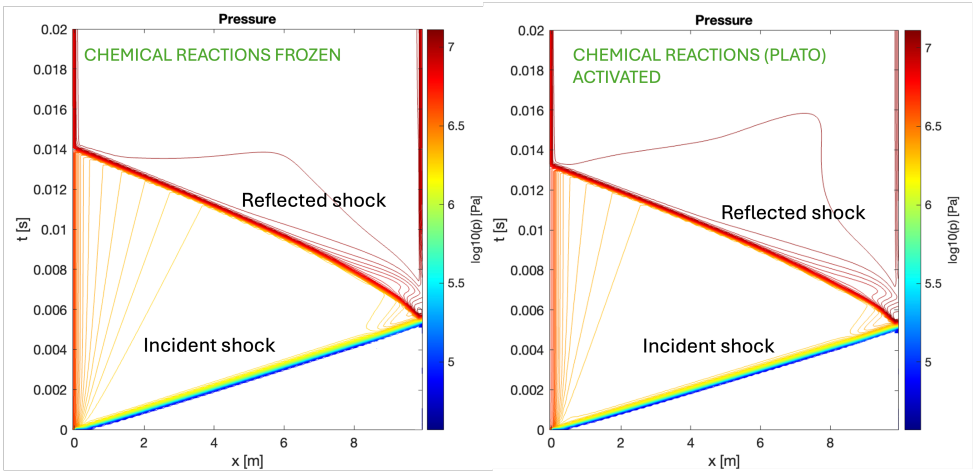
In this section, the HYQ1D code will be used to simulate various sections of the UQ-T4 tunnel as well as the entire facility, comparing the results with the *L1d* code results [34] and experimental measurements. The facility can be schematized as shown in figure 9.10.

9.1 Shock Tube Only Test

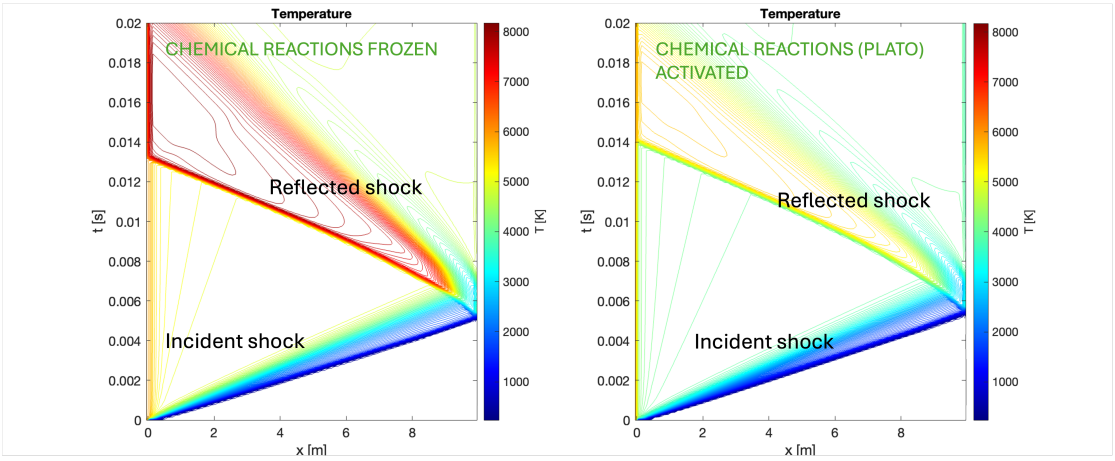
In this section, we begin the analysis of the facility with a stand-alone shock tube simulation, using the UQ-provided data as the initial condition, a Dirichlet boundary condition on the left side, and a Wall boundary condition on the right, which switches to a Neumann condition when the pressure exceeds the diaphragm burst pressure. Various comparisons are presented using different models implemented in the HYQ1D code.

9.1.1 Non-reactive Gas vs. Chemical Non-equilibrium

For the first comparison, shown in figure 9.1, we can see how the chemical model drastically changes the results in terms of fluid dynamic variables for the temperatures reached in the tube. Specifically, the temperature is much lower, in chemical reactive case, because the reactions are endothermic. Therefore, it is crucial to account for chemical reactions in the model.



(a)

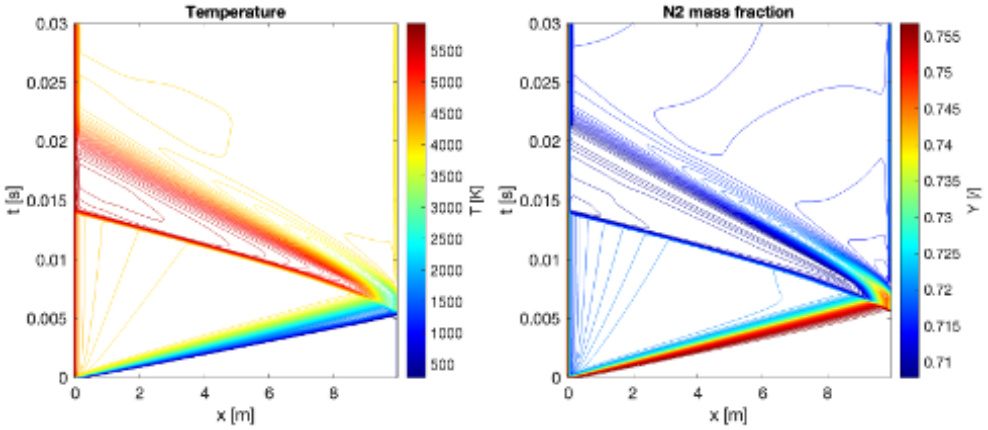


(b)

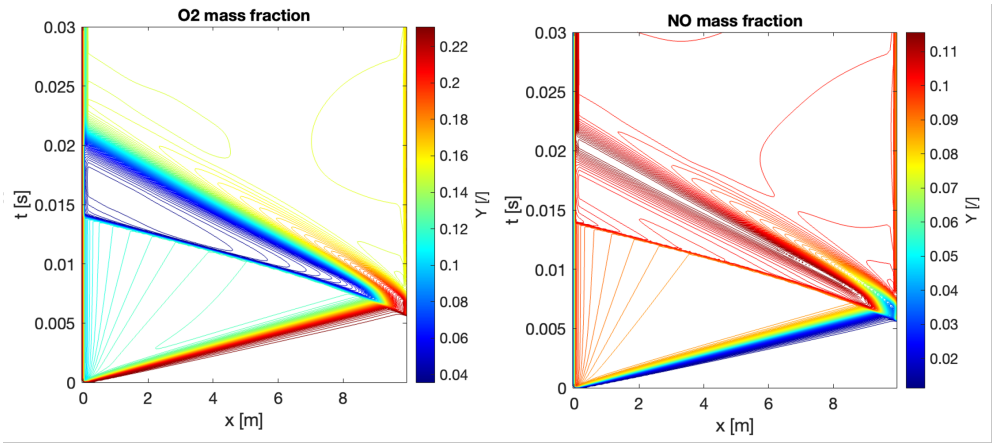
Figure 9.1: Non reactive gas vs chemical non equilibrium pressure and temperature comparison

9.1.2 HYQ1D + PLATO Simulation: 5-Species Model vs. 11-Species Model

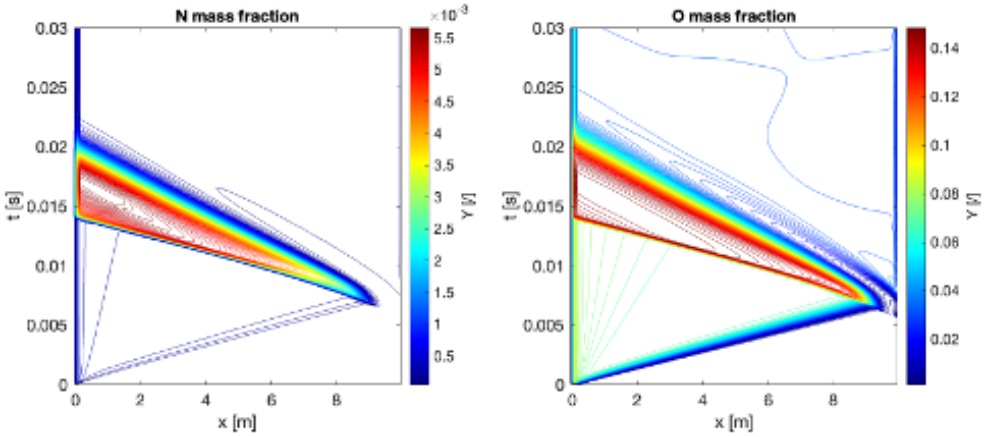
In this section, we present a comparison between the results obtained using a chemical reaction model where the source terms are calculated with the PLATO library. The comparison is made between a 5-species model (with no ionization considered) and an 11-species model, which includes ions. This analysis aims to determine whether the 5-species model is insufficiently accurate and if ionization must be accounted for.



(a)

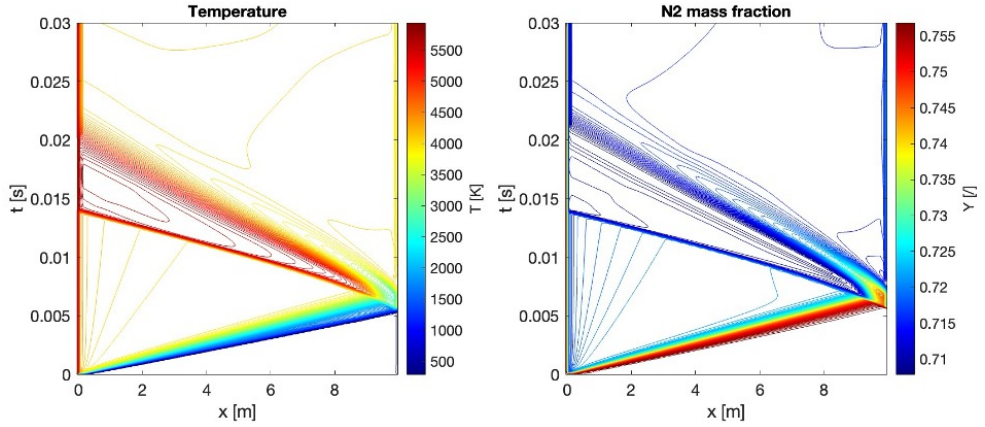


(b)

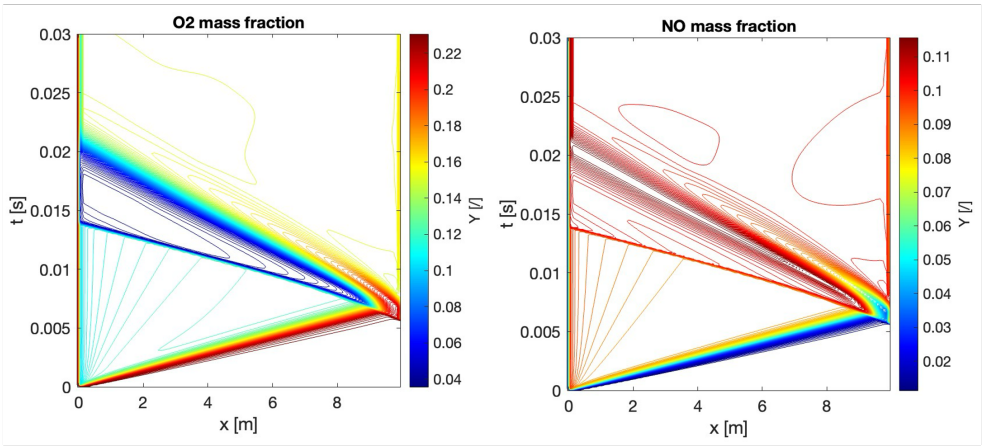


(c)

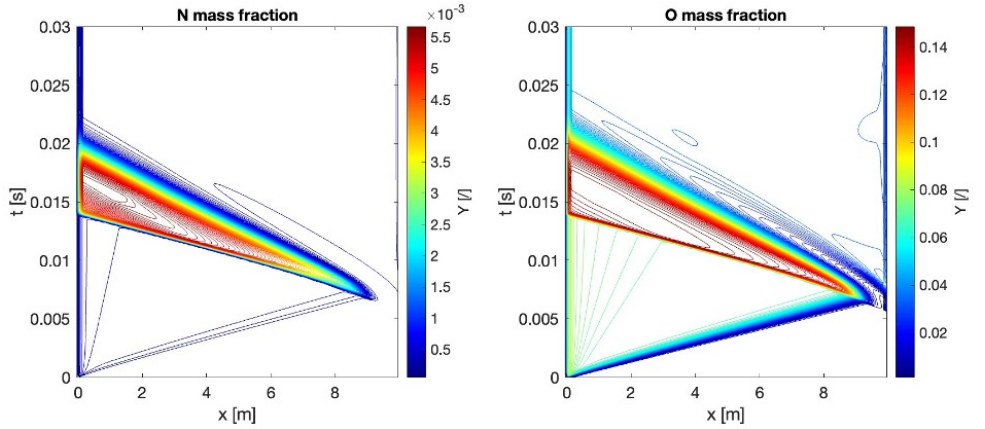
Figure 9.2: Mass fractions, 5 species model



(a)

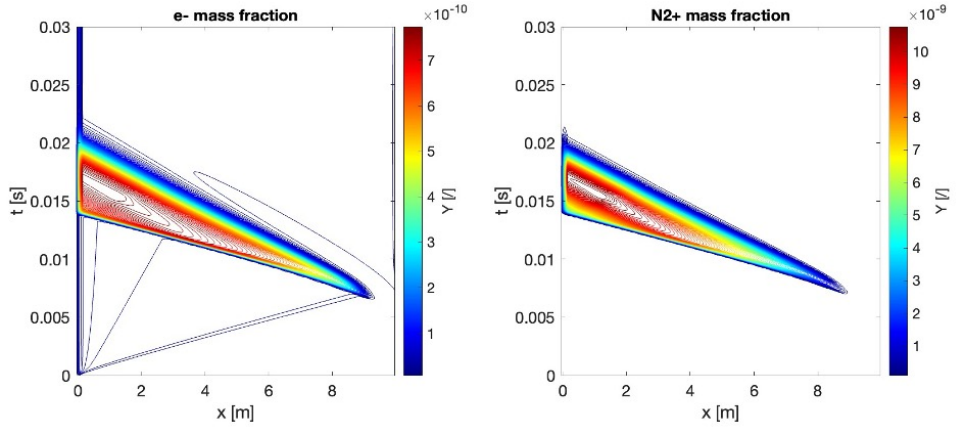


(b)

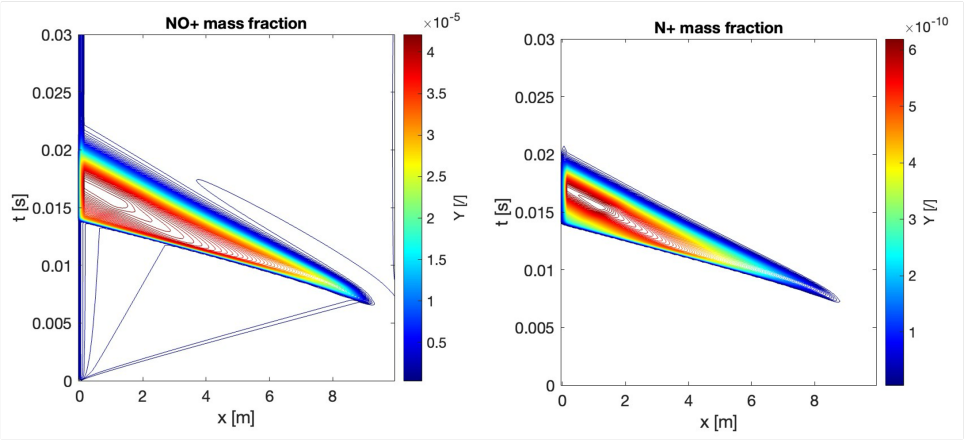


(c)

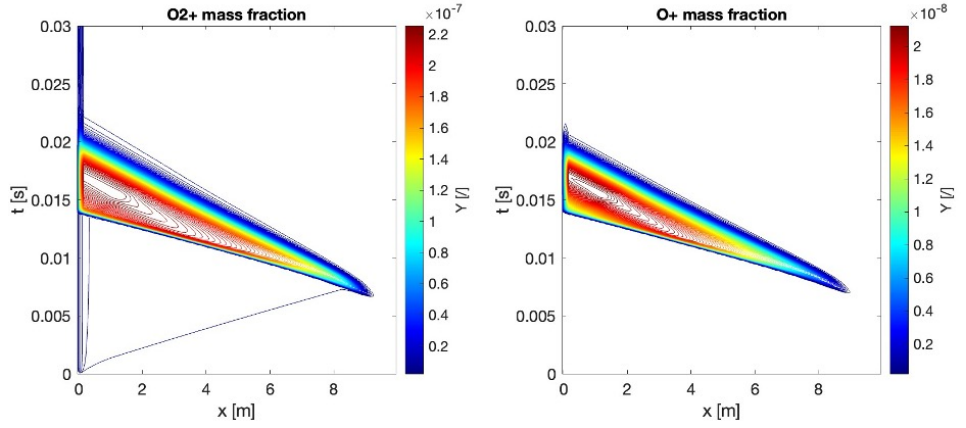
Figure 9.3: Mass fractions, 11 species model, part 1



(a)



(b)



(c)

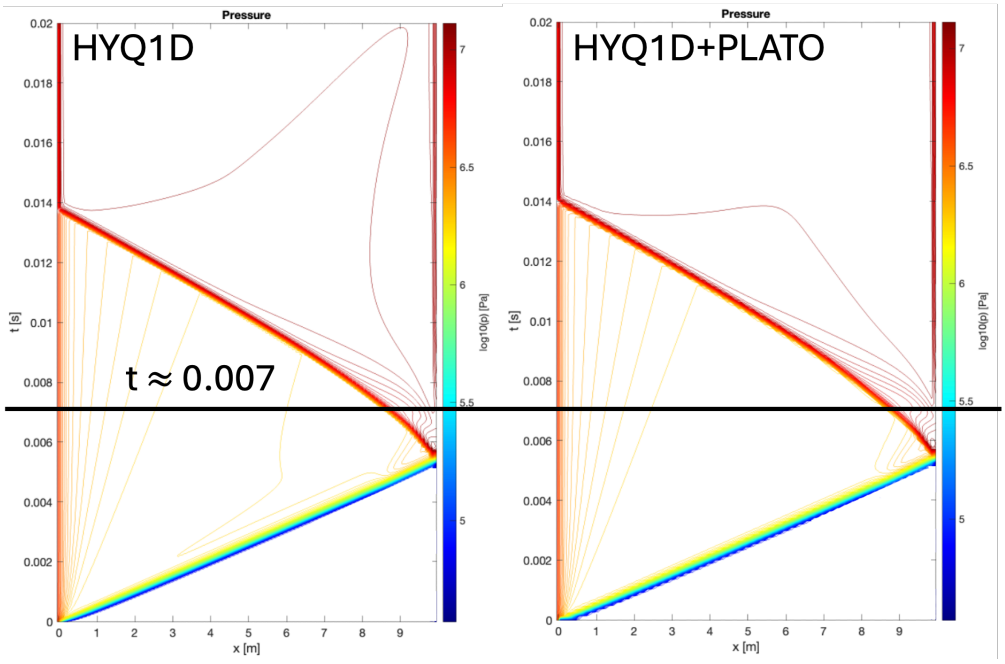
Figure 9.4: Mass fractions, 11 species model, part 2

In figure 9.2, the results for mass fractions in the 5-species model are presented, and in figures 9.3 and 9.6, the results for the 11-species model are shown. It is clear that the results are substantially the same, with the ions reaching the highest mass fraction being NO^+, which has a maximum mass fraction of 10^{-5}—still very low.

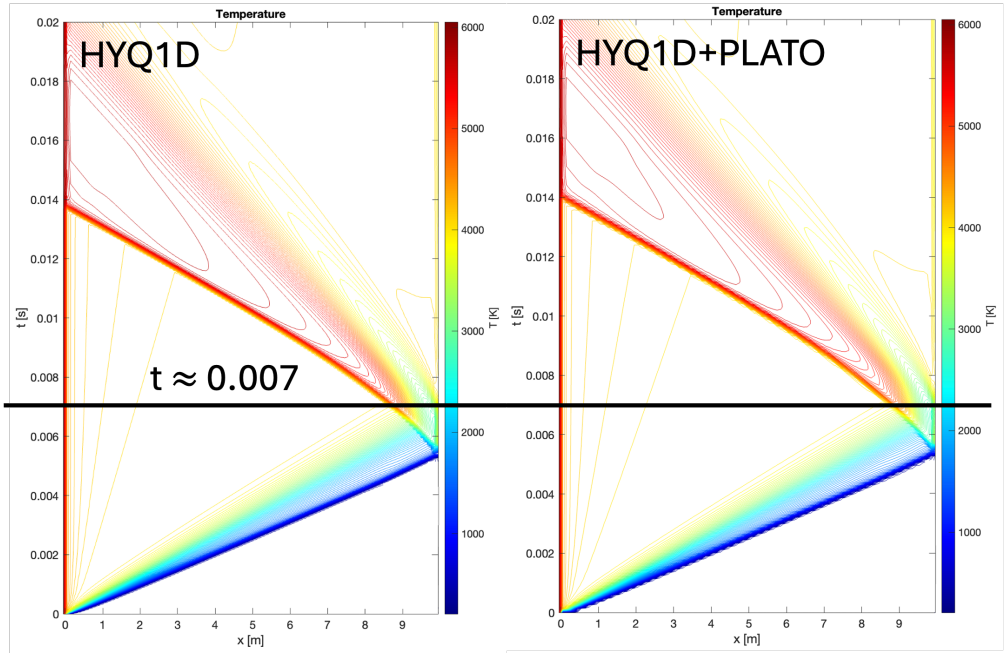
9.1.3 HYQ1D vs. HYQ1D + PLATO

In this section, we compare the chemical non-equilibrium results obtained from the 5-species model in the self-implemented HYQ1D chemical model with the results obtained by calculating the chemical source term using the PLATO library.

From figure 9.5, we can conclude that no substantial differences are noticeable in the temperature and pressure contour plots. To better compare the two calculations, the x-t plot was cut at $t \simeq 0.007s$, which is a time after the diaphragm burst when the temperature is high enough to highlight any potential differences present in the molar fraction plots.

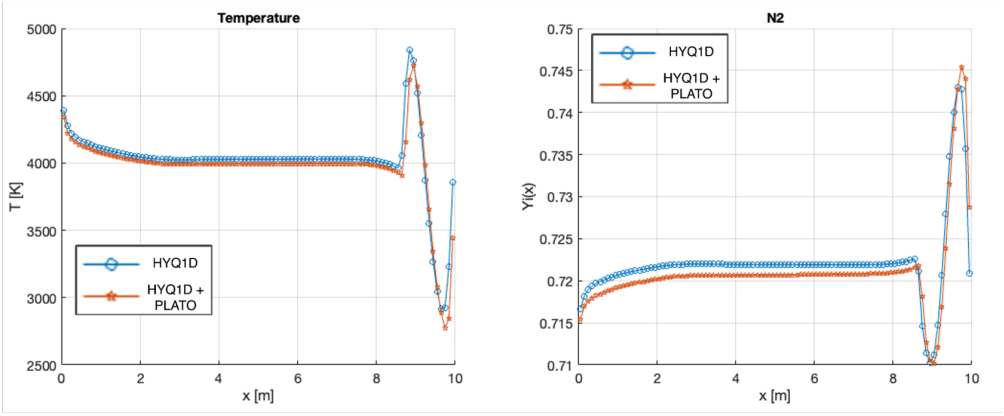


(a)

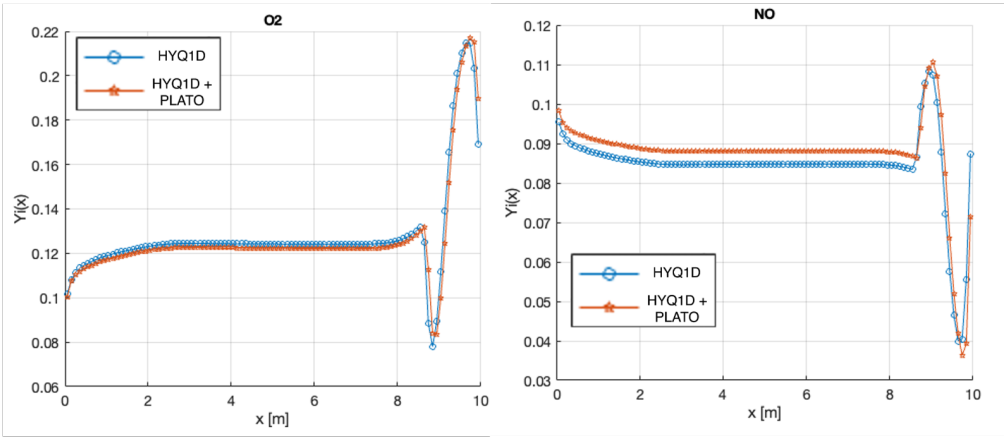


(b)

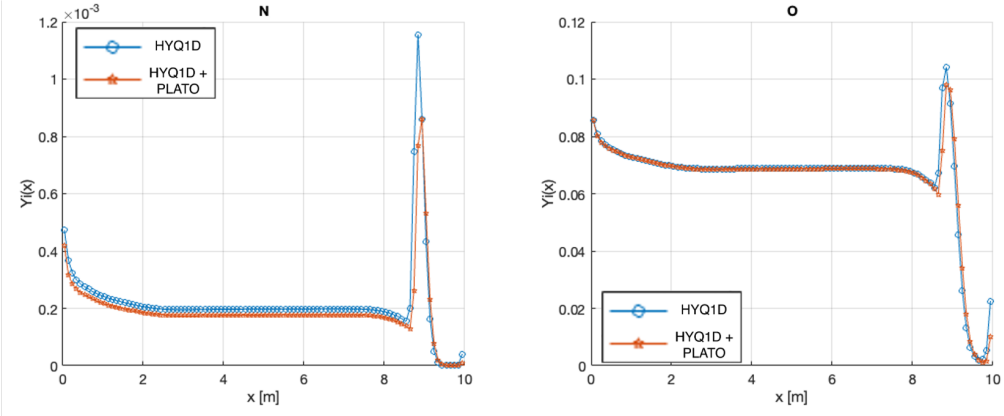
Figure 9.5: HYQ1D vs HYQ1D + PLATO, pressure and temperature comparison



(a)



(b)



(c)

Figure 9.6: HYQ1D vs HYQ1D + PLATO, mass fraction comparison at $t \simeq 0.007s$

In figure 9.6, the molar fractions are plotted at the time mentioned above. It is evident that the two results are consistent with each other. We can highlight a small difference in the plot where nitrogen atoms are present in the species. It seems that the PLATO libraries overpredict the formation of NO compared to the HYQ1D stand-alone model, while underpredicting N_2 dissociation, leading to differences in N and O_2. This is probably due to more accurate chemical coefficients used in the PLATO kinetic model. However, this leads to no substantial difference in the temperature plot, which is one of the fluid dynamic variables most sensitive to chemical phenomena.

9.2 Nozzle Only Test

In this section, we move forward in the UQ-T4 tunnel, analyzing the nozzle. The nozzle shape shown in figure 9.7 was provided by the QU research group, which works on the facility. Here, some comparisons are made between the various models implemented in the code, as done in the previous section, including a comparison with the data provided by QU.

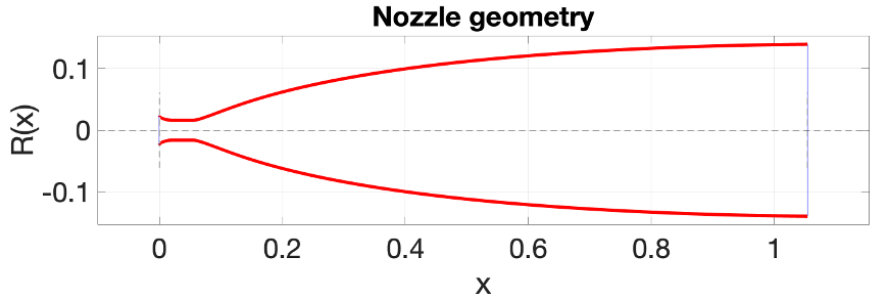


Figure 9.7: UQ-T4 Nozzle geometry

9.2.1 HYQ1D + PLATO simulation, 5 species model vs. 11 species model

Here, a comparison was made using first a 5-species chemical model (without ionization) and then an 11-species model (with ions). The PLATO libraries were used to calculate the chemical source term in both cases.

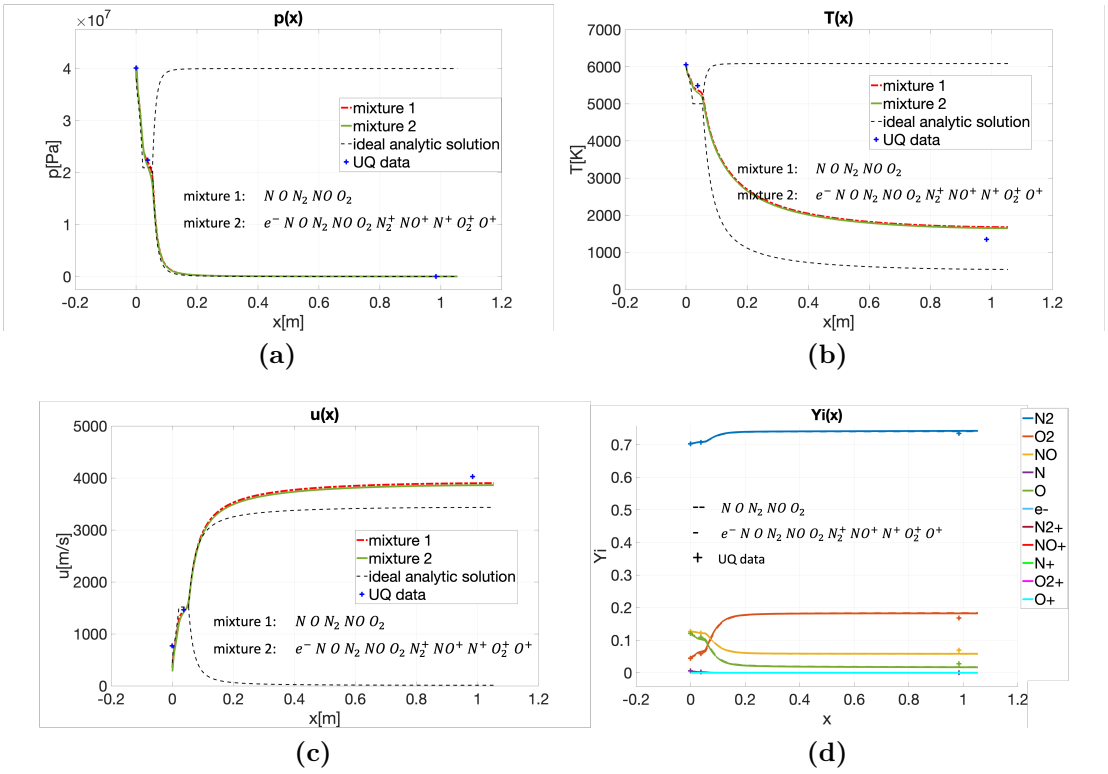


Figure 9.8: 5 species vs 11 species model comparison

In figure 9.8, the solution is plotted for pressure, temperature, velocity, and mass fractions. The results from the two compared models are plotted alongside the ideal nozzle solution and the data provided by UQ.

It is evident that the nozzle solution follows the expected trend of a subsonic inlet and supersonic outlet in a convergent-divergent nozzle. The results differ significantly from the ideal solution due to real gas effects, which are fundamental at high temperatures. In the divergent part of the nozzle, it becomes clear that the chemistry can be considered frozen due to the high Mach number of the flow. The results from the two different mixtures are almost identical, with no substantial differences, and both closely match the data provided by QU. Even better results could likely be achieved by considering thermodynamic non-equilibrium as well.

9.2.2 HYQ1D vs. HYQ1D + PLATO

In this section, using a 5-species mixture, the results obtained with the stand-alone HYQ1D chemistry are compared to those from a calculation where the chemical source term is modeled using the PLATO libraries.

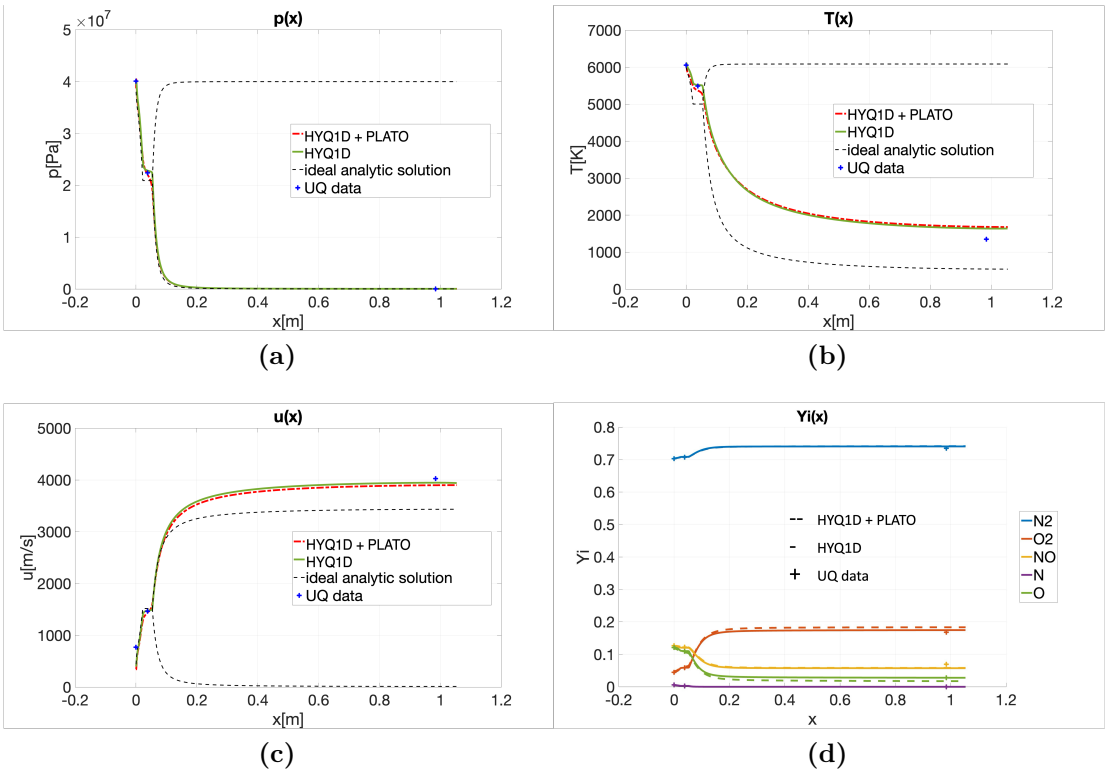


Figure 9.9: HYQ1D vs HYQ1D + PLATO comparison

In figure 9.9, in addition to the general considerations made in the previous section, we can observe that the two calculations follow the same trend, with small differences, for example in the mass fractions, which are comparable to those encountered in section 9.1.3. The stand-alone HYQ1D case shows better agreement with the UQ-provided data, possibly because the PLATO library coefficients are more accurate than those used in the UQ calculations. Probably, HYQ1D uses a chemical model more similar to that employed in the UQ model.

9.3 UQ-T4 complete facility simulation

Here, the results and comparisons for the complete QU-T4 facility are presented, representing the facility discretization as implemented in the code is presented in Figure 9.10.

It is notable how from the logarithm (base 10) of pressure in Figure 9.12 and 9.11, all the peculiar characteristics of the QU-T4 functioning become apparent. Taking into account the spatial distribution of the entire wind tunnel section, as reported in Table 9.1, we can observe how the pressure differential across the piston

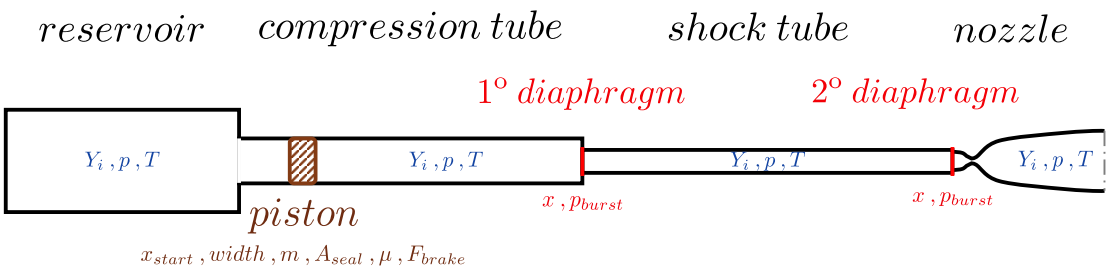
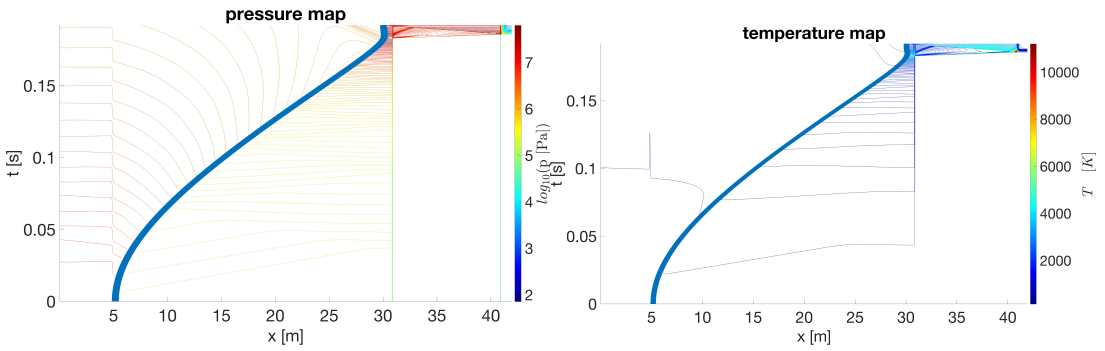


Figure 9.10: Typical piston driven wind tunnel configuration.

	$I.C.: \quad Y_i \quad [/]$	$I.C.: \quad p \quad [kPa]$	$I.C.: \quad T \quad [K]$	length [m]	diameter [m]
Reservoir	Air	5600	300	4.87	0.4414
Compression	0.9He-0.1Ar	115	300	26	0.229
Shok Tube	Air	40	300	10	0.076
Nozzle	Air	0.067	300	1	0.279
	$x_{position} \quad [m]$	$width \quad [m]$	$m \quad [Kg]$	$\mu \quad [/]$	$p_{burst} [MPa]$
Piston	0.235	0.37	90.05	0.2	
Diaphragm 1	26				60
Diaphragm 2	36				1

Table 9.1: UQ-T4 wind tunnel data [4]-[71]

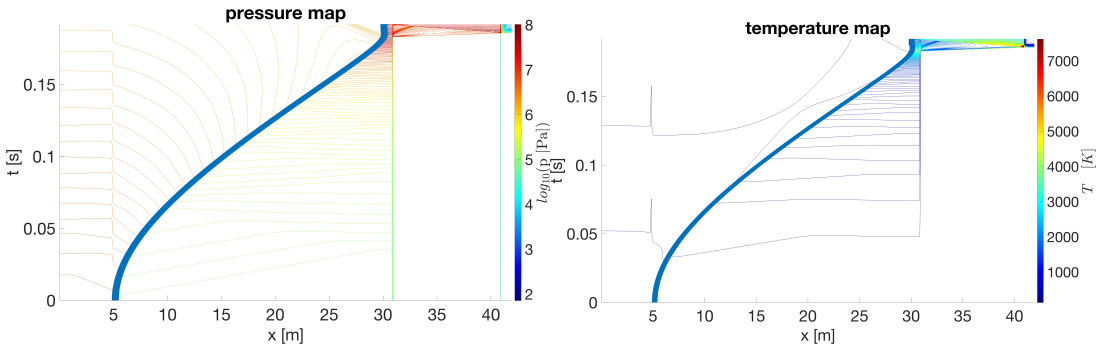


(a) base 10 logarithm of pressure **(b)** temperature

Figure 9.11: x-t diagram showing wave phenomena for the data in table 9.1. Test case without reactive chemistry model.

causes its movement and how this generates an expansion wave behind it and a compression wave in front of it. These waves reflect on the left wall (for the expansion wave) and on the first diaphragm (for the compression wave), and this mechanism continues until the pressure becomes high enough to burst the first diaphragm.

It is also notable that, at each area reduction—for example, from the reservoir



(a) base 10 logarithm of pressure **(b)** temperature

Figure 9.12: x-t diagram showing wave phenomena for the data in table 9.1. Test case with reactive chemistry model.

to the compression tube—there is a rapid expansion of the fluid, as expected. As the piston approaches the first diaphragm, the fluid on its right side becomes increasingly compressed, and at a certain point, the piston's acceleration starts to decrease, eventually becoming negative.

When the pressure is sufficient to burst the diaphragm, the piston is almost stationary, experiencing high negative acceleration, which initiates a backward motion. Once the diaphragm bursts, a shock wave appears at the interface between the highly compressed gas in the compression tube and the gas in the shock tube. This generates a very fast shock that reaches the second diaphragm in almost 2 ms, causing it to burst. The shock first reflects off the diaphragm, which instantly ruptures due to the high fluid pressure, reaching its peak value, and initiating the expansion into the convergent-divergent nozzle.

It is important to note an oscillation in the shock tube that begins after the rupture of the first diaphragm and is convected forward until the reflected shock clears it upon returning. This oscillation is likely the result of the instantaneous contact between two completely different gases: the driver gas and the air filling the shock tube. This issue appears to be a common problem, as it has been studied in the literature, and a solution has been proposed using a modified form of the AUSM solver [72, 73]. Implementing this solution will be the next step for the HYQ1D code.

Looking at subfigure *b* in Figure 9.11 , it is evident that the temperature resulting from the post-shock conditions in the shock tube massively triggers chemical reactions, particularly when the shock impacts the second diaphragm, so it is essential to consider the fluid chemical reactive (A comparison between non reacting result and reacting fluid result could be done comparing, for example, Figure 9.12 and Figure 9.11, results are substantially different from the shock tube

and after).

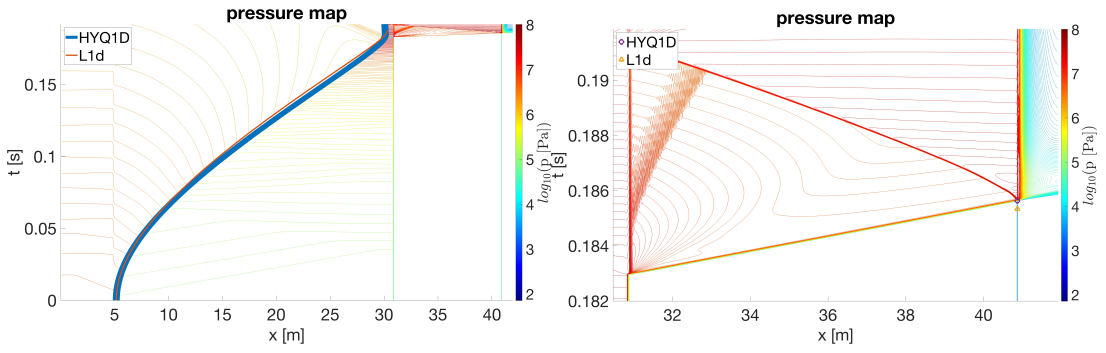


Figure 9.13: model comparisons with QU-T4 data

In Figure 9.13, a comparison is shown between the results obtained in this work and those from the research code *L1d* [34] for the same test case.

On the left side, in subfigure *a*, overlaid on subfigure *a* of Figure 9.12 is the piston midpoint trajectory calculated using the *L1d* code. The results from this work seem to slightly anticipate the diaphragm burst compared to the Australian ones, but overall, there is a good agreement and consistency between the two sets of results.

On the right side, in subfigure *b*, a comparison of shock velocity is presented. Using the first diaphragm rupture obtained in this work as a common reference point, the position of the second diaphragm rupture is plotted on the x-t diagram for two research codes, using different markers. In this way, the burst time of the second diaphragm is directly related to the shock velocity. From the plotted markers, we can observe that the result obtained in this work slightly delays the second diaphragm rupture compared to the *L1d* code. This indicates that a slightly lower shock velocity was predicted in this work. However, overall, the results appear consistent with those from the previous comparison.

It is important to consider that the HYQ1D code is a quasi-one-dimensional code, so it doesn't account for 3D effects like separation and reattachment at abrupt area changes in the facility. This leads to a systematic error in all the results, due to the physics of the problem, which cannot be resolved by this approximation. In this code, the area changes are smoothed to ensure stability and to qualitatively mimic the fluid dynamic shape created by the recirculation bubbles that would form in the 3D detached flow.

Chapter 10

DLR-HEG wind tunnel simulation results

Here, the results and comparisons for the DLR-HEG facility are presented. The facility was discretized in the same way as the QU-T4 facility, as illustrated in the schematic in Figure 9.10. Regarding the reservoir, since the volume was known, it was decided to discretize it as a cylinder preceding the compression tube, with a length of 5 meters. The diameter was derived from this assumption and the provided volume.

In this simulation, the nozzle was not considered due to insufficient data. The remaining facility data are grouped in Table 10.1, with the black data taken from the references cited in the table caption and the gray data being assumptions used to run the test case.

	$I.C.: \quad Y_i \quad [/]$	$I.C.: \quad p \quad [kPa]$	$I.C.: \quad T \quad [K]$	length [m]	diameter/volume
Reservoir	Air	5000	300	5	5 $[m^3]$
Compression	He	72.5	300	33	0.55 $[m]$
Shok Tube	N_2	23.8	300	17	0.15 $[m]$
	$x_{position} \quad [m]$	$m \quad [Kg]$	$p_{burst}[MPa]$		
Piston	5	268			
Diaphragm 1	38		49.3		

Table 10.1: DLR-HEG wind tunnel data for shot51 [74, 75]

Here, we present only the results of the chemically reactive model because, as in the previous case, the resulting temperature in the shock tube causes chemistry to be massively triggered.

Figure 10.1 provides an overview of the wave phenomena. The physical description is qualitatively the same as that provided for the QU-T4 facility in section 9.

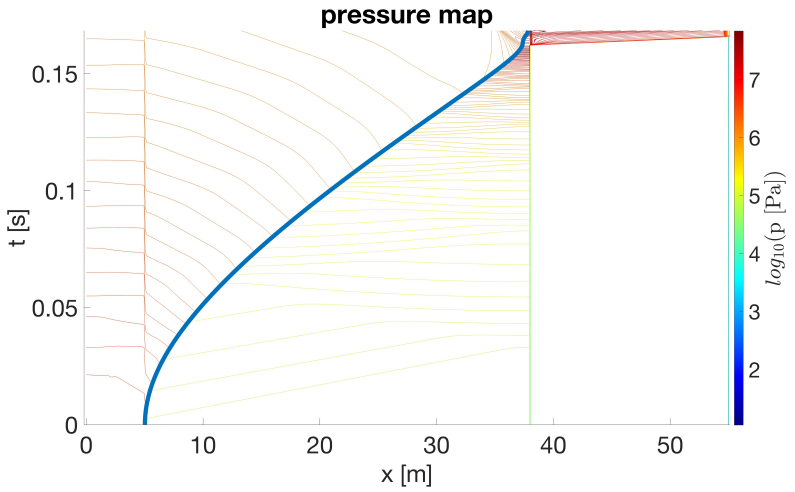


Figure 10.1: x-t diagram showing wave phenomena for the data in table 10.1, base 10 logarithm of pressure is shown. Test case with reactive chemistry model.

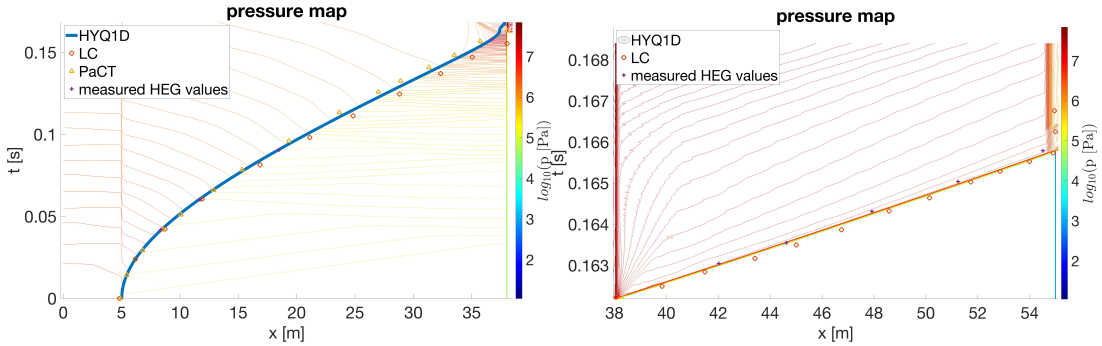


Figure 10.2: model comparisons with DLR-HEG data

In Figure 10.2, two comparisons are made, as done for the QU-T4 facility in Figure 9.13. All the data for the comparison are extrapolated from [74, 75]. The first comparison, in subfigure *a*, focuses on the piston trajectory between the results obtained in this study, two research codes (*LC* and *PaCT*), and HEG measurements. The plot clearly shows that the developed model produces results consistent with the other methods used for comparison. Specifically, the results fall between those of the two other codes and closely agree with the experimental measurements.

For the second comparison, in subfigure *b*, the results obtained in this work, the *LC* results, and the experimental measurements for the shock front on an x-t plane are shown. From this graph, we can see that the shock velocity predicted in this

work is slightly closer to the experimental measurement than that predicted by the LC code, though it remains faster than the experimental data. Further tuning of the loss model may be required to improve the results. Overall, the results obtained in this comparison are comparable to alternative code formulations and consistent with them.

Conclusion

The HYQ1D research code developed in this work includes a wide range of numerical methods for time integration and spatial reconstruction, from first-order to high-order methods. Additionally, various Riemann solvers are available for flux evaluation. All the numerical methods were tested, and their solutions were verified in section 8.1 using different tabulated 1D shock tube tests.

The area variation capabilities of the quasi-one-dimensional formulation were verified in section 8.2, where the code was tested for convergence to the ideal nozzle solution by varying the inlet/outlet pressure ratio. The chemical and thermal non-equilibrium capabilities were also tested in this section, showing the expected trends and results. The code was further extensively tested on the QU-T4 and DLR-HEG facilities. Some tests on the former were performed using the chemical non-equilibrium formulation and the PLATO library to calculate the chemical source terms. The results showed good agreement, verifying the chemical non-equilibrium capabilities in sections 9.1.3 and 9.9.

In conclusion, from the results reported in sections 9.3 and 10, an assessment of the overall model's accuracy and consistency was conducted. It was observed that the overall results of the simulated facilities show good agreement with the comparison data. Furthermore, when isolating the compression tube and focusing on the free-piston behavior simulation, the comparison showed even better results. The free-piston model developed in this work produced results comparable to other models designed for the same purpose and closely matched experimental measurements (section 10).

Therefore, we can conclude that the main objective of this work has been achieved. The free-piston model developed, and the HYQ1D code in general, effectively describe the physics of free-piston-driven hypersonic wind tunnels, showing good agreement with experimental data and simulations from similar research codes. However, further improvements could be made to the overall code, such as incorporating the AUSM solver to avoid oscillations at the interface between gases with completely different properties.

Bibliography

[1] Alessandro Munafò and Marco Panesi. «Plato: a high-fidelity tool for multi-component plasmas». In: *AIAA Aviation 2023 Forum*. 2023, p. 3490 (cit. on pp. i, 45–50).

[2] John J Bertin, Jacques Periaux, and Josef Ballmann. *Advances in Hypersonics: Modeling Hypersonic Flows*. Springer, 1992 (cit. on p. 1).

[3] Graham V Candler. «Nonequilibrium hypersonic flows and hypersonic nozzle flow modeling». In: *NATO STO Lecture Series: Flow Characterization and Modeling of Hypersonic Wind Tunnels, STO-AVT-352-VKI* (2018) (cit. on pp. 1, 3, 5, 21).

[4] PA Jacobs. «Quasi-one-dimensional modeling of a free-piston shock tunnel». In: *AIAA journal* 32.1 (1994), pp. 137–145 (cit. on pp. 1, 13, 15, 31, 85).

[5] R. J. Stalker. «A Study of the Free-Piston Shock Tunnel». In: *AIAA Journal* 5.12 (1967), pp. 2160–2165. DOI: 10.2514/3.4402. URL: https://doi.org/10.2514/3.4402 (cit. on pp. 1, 31).

[6] Katsuhiro Itoh, Kouichiro Tani, Hideyuki Tanno, Masahiro Takahashi, Hiroshi Miyajima, Takahisa Asano, Akihiro Sasoh, and Kazuyoshi Takayama. «A Numerical and Experimental Study of the Free Piston Shock Tunnel». In: *Shock Waves @ Marseille I*. Ed. by Raymond Brun and Lucien Z. Dumitrescu. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 257–262. ISBN: 978-3-642-78829-1 (cit. on pp. 1, 31).

[7] W. Y. K. Chan, R. W. Whitside, M. K. Smart, D. E. Gildfind, P. A. Jacobs, and T. Sopek. «Nitrogen driver for low-enthalpy testing in free-piston-driven shock tunnels». In: *Shock Waves* 31 (2021), pp. 541–550. DOI: 10.1007/s00193-021-01002-0 (cit. on p. 1).

[8] Klaus Hannemann, Jan Martinez Schramm, Alexander Wagner, and Giannino Ponchio Camillo. «The high enthalpy shock tunnel Göttingen of the German aerospace center (DLR)». In: *Journal of large-scale research facilities JLSRF* 4.A133 (2018), pp. 1–14 (cit. on p. 1).

[9] Klaus Hannemann. «High enthalpy flows in the HEG shock tunnel: experiment and numerical rebuilding». In: *41st Aerospace Sciences Meeting and Exhibit*. 2003, p. 978 (cit. on p. 1).

[10] JAXA Business Development and Industrial Relations Department. *High Enthalpy Shock Tunnel (HIEST)*. `https://aerospacebiz.jaxa.jp/en/solution/facility/facility64/`. Accessed: 2024-10-15. 2024 (cit. on p. 1).

[11] California Institute of Technology. *T5 Hypervelocity Shock Tunnel*. `https://www.austin.caltech.edu/hypersonics/t5.html`. Accessed: 2024-10-15. 2024 (cit. on p. 1).

[12] Purdue University Hypersonics Program. *Hypersonics Facilities at Purdue University*. `https://engineering.purdue.edu/hypersonics/hypersonics-facilities`. Accessed: 2024-10-15. 2024 (cit. on p. 1).

[13] C.R. Bagshaw. *Biomolecular Kinetics: A Step-by-Step Guide*. 1st. CRC Press, 2017. DOI: `10.1201/9781315120355` (cit. on p. 2).

[14] H.L. Otálvaro-Marín and F. Machuca-Martínez. «Sizing of reactors by charts of Damköhler's number for solutions of dimensionless design equations». In: *Heliyon* 6.11 (2020), e05386. DOI: `10.1016/j.heliyon.2020.e05386` (cit. on p. 2).

[15] Peter A Gnoffo. *Conservation equations and physical models for hypersonic air flows in thermal and chemical nonequilibrium*. Vol. 2867. National Aeronautics and Space Administration, Office of Management ..., 1989 (cit. on pp. 3, 4, 6, 19).

[16] Thierry Poinsot and Denis Veynante. *Theoretical and Numerical Combustion*. 3rd. Philadelphia, PA: Edwards, 2011. ISBN: 978-1-930217-91-6 (cit. on p. 3).

[17] Navin Singh and Thomas Schwartzentruber. «Nonequilibrium internal energy distributions during dissociation». In: *Proceedings of the National Academy of Sciences* 115.1 (2018), pp. 47–52. DOI: `10.1073/pnas.1713840115` (cit. on p. 5).

[18] Peter Atkins and Ronald Friedman. *Molecular Quantum Mechanics*. 5th. Oxford: Oxford University Press, 2011 (cit. on pp. 5, 6).

[19] Iain D Boyd and Thomas E Schwartzentruber. *Nonequilibrium gas dynamics and molecular simulation*. Vol. 42. Cambridge University Press, 2017 (cit. on p. 5).

[20] Bonnie J McBride. *NASA Glenn coefficients for calculating thermodynamic properties of individual species*. National Aeronautics and Space Administration, John H. Glenn Research Center ..., 2002 (cit. on pp. 5, 20).

[21] Alessandro Munafò, Andrea Alberti, Carlos Pantano, Jonathan B Freund, and Marco Panesi. «A computational model for nanosecond pulse laser-plasma interactions». In: *Journal of Computational Physics* 406 (2020), p. 109190 (cit. on p. 5).

[22] AW Ali. «The Harmonic and Anharmanic Models for Vibrational Relaxation and Dissociation of the Nitrogen Molecule». In: *Naval Research'Laboratory, Washington, DC* (1986) (cit. on p. 6).

[23] John Smith and Jane Doe. «Nonequilibrium Vibrational, Rotational, and Electronic Energy Distributions in High-Speed Flows». In: *Journal of Fluid Mechanics* 900 (2020), pp. 123–145. DOI: 10.1017/jfm.2020.1234 (cit. on p. 7).

[24] Roger C Millikan and Donald R White. «Systematics of vibrational relaxation». In: *The Journal of chemical physics* 39.12 (1963), pp. 3209–3213 (cit. on p. 7).

[25] Chul Park. «Problems of rate chemistry in the flight regimes of aeroassisted orbital transfer vehicles». In: *19th Thermophysics Conference*. 1984, p. 1730 (cit. on pp. 7, 19).

[26] Chul Park. «Review of chemical-kinetic problems of future NASA missions. I-Earth entries». In: *Journal of Thermophysics and Heat transfer* 7.3 (1993), pp. 385–398 (cit. on pp. 7, 19).

[27] Frank E Peñaranda and M Shannon Freda. *Aeronautical Facilities Catalogue. Volume 1: Wind Tunnels*. Tech. rep. 1985 (cit. on p. 8).

[28] S.L. Gai. «Free piston shock tunnels: developments and capabilities». In: *Progress in Aerospace Sciences* 29.1 (1992), pp. 1–41. ISSN: 0376-0421. DOI: https://doi.org/10.1016/0376-0421(92)90002-Y. URL: https://www.sciencedirect.com/science/article/pii/037604219290002Y (cit. on p. 8).

[29] DLR. *Arc Heated Wind Tunnel L2K*. https://www.dlr.de/en/research-and-transfer/research-infrastructure/arc-heated-wind-tunnel-l2k. Accessed: 21-Oct-2024. 2024 (cit. on p. 8).

[30] Zonglin Jiang and Randy S. M. Chue. «Detonation-Driven High-Enthalpy Shock Tunnels». In: *Theories and Technologies of Hypervelocity Shock Tunnels*. Cambridge University Press, 2023, pp. 119–164 (cit. on p. 8).

[31] NASA Glenn Research Center. *Blowdown Wind Tunnels*. https://www.grc.nasa.gov/www/k-12/airplane/tunblow.html. Accessed: 21-Oct-2024. 2024 (cit. on p. 8).

[32] John D. Anderson. *Fundamentals of aerodynamics*. 5th. McGraw-Hill, Feb. 2011. ISBN: 9780073398105. URL: http://www.worldcat.org/isbn/9780073398105 (cit. on pp. 10, 34).

[33] Andrea Maranzoni. «Modellazione numerica e fisica di moti bidimensionali a pelo libero». PhD thesis. Università degli Studi di Parma: Università degli Studi di Parma, 2004 (cit. on p. 13).

[34] Peter A Jacobs. «Shock tube modelling with L1d». In: (1998) (cit. on pp. 13, 15, 74, 87).

[35] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 3rd. Berlin, Heidelberg: Springer, 2009. ISBN: 978-3-540-25202-3. DOI: 10.1007/978-3-540-49834-6 (cit. on pp. 16, 22, 25, 69).

[36] Dale R. Durran. *Numerical Methods for Fluid Dynamics: With Applications to Geophysics*. 2nd. Springer, 1999. DOI: 10.1007/978-1-4757-3121-5 (cit. on p. 16).

[37] Joel H. Ferziger and Milovan Peric. *Computational Methods for Fluid Dynamics*. 3rd. Springer, 1996. DOI: 10.1007/978-3-540-72159-8 (cit. on p. 16).

[38] Randall J. LeVeque. *Finite-Volume Methods for Hyperbolic Problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002. DOI: 10.1017/CBO9780511791253 (cit. on p. 16).

[39] Philip L. Roe. «Riemann Problem». In: *Encyclopedia of Applied and Computational Mathematics*. Ed. by Björn Engquist. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 1257–1260. ISBN: 978-3-540-70529-1. DOI: 10.1007/978-3-540-70529-1_357. URL: https://doi.org/10.1007/978-3-540-70529-1_357 (cit. on p. 16).

[40] Christian Di Lizia. «Ottimizazione di schemi alla Godunov per la simulaione di flussi quasi-unidimensionali non stazionari». Tesi in formato PDF. PhD thesis. Università Degli Studi, 2024 (cit. on p. 16).

[41] Stanley Osher and Fredric Solomon. «Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws». In: *Mathematics of Computation* 38.158 (1982), pp. 339–374. DOI: 10.2307/2007475 (cit. on p. 17).

[42] Jesse W Streicher, Ajay Krish, Ronald K Hanson, Kyle M Hanquist, Ross S Chaudhry, and Iain D Boyd. «Shock-tube measurements of coupled vibration–dissociation time-histories and rate parameters in oxygen and argon mixtures from 5000 K to 10 000 K». In: *Physics of Fluids* 32.7 (2020) (cit. on p. 19).

[43] Steven C. Chapra and Raymond P. Canale. *Numerical Methods for Engineers*. McGraw-Hill, 2002 (cit. on pp. 20, 21).

[44] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992 (cit. on pp. 20, 21).

[45] Richard Courant. «Courant-Friedrichs-Lewy condition». In: (2018) (cit. on p. 22).

[46] Jonathan Weiss. «Calculation of reacting flowfields involving stiff chemical kinetics». In: *14th Computational Fluid Dynamics Conference*. 1999, p. 3369 (cit. on p. 22).

[47] Chi-Wang Shu and Stanley Osher. «Efficient implementation of essentially non-oscillatory shock-capturing schemes». In: *Journal of computational physics* 77.2 (1988), pp. 439–471 (cit. on p. 25).

[48] Sigal Gottlieb and Chi-Wang Shu. «Total variation diminishing Runge-Kutta schemes». In: *Mathematics of computation* 67.221 (1998), pp. 73–85 (cit. on p. 25).

[49] David I Ketcheson, Colin B Macdonald, and Sigal Gottlieb. «Optimal implicit strong stability preserving Runge–Kutta methods». In: *Applied Numerical Mathematics* 59.2 (2009), pp. 373–392 (cit. on p. 26).

[50] P Batten, DM Ingram, R Saunders, and DM Causon. «A time-splitting approach to solving the Navier-Stokes equations». In: *Computers & fluids* 25.4 (1996), pp. 421–431 (cit. on p. 26).

[51] Axel Ariaan Lukassen and Martin Kiehl. «Operator splitting for chemical reaction systems with fast chemistry». In: *Journal of Computational and Applied Mathematics* 344 (2018), pp. 495–511 (cit. on p. 26).

[52] Y-T Zhang and C-W Shu. «ENO and WENO schemes». In: *Handbook of numerical analysis*. Vol. 17. Elsevier, 2016, pp. 103–122 (cit. on p. 27).

[53] Michael Dumbser, Dinshaw S. Balsara, Eleuterio F. Toro, and Claus-Dieter Munz. «A Finite Volume Method Based on a WENO Reconstruction for Compressible Flows on Hybrid Grids». In: *Journal of Computational Physics* 227.8 (2007), pp. 3971–4001. DOI: 10.1016/j.jcp.2007.01.022 (cit. on p. 27).

[54] Chi-Wang Shu. «Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws». In: *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations: Lectures given at the 2nd Session of the Centro Internazionale Matematico Estivo (C.I.M.E.) held in Cetraro, Italy, June 23–28, 1997*. Ed. by Alfio Quarteroni. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 325–432. ISBN: 978-3-540-49804-9. DOI: 10.1007/BFb0096355. URL: https://doi.org/10.1007/BFb0096355 (cit. on p. 27).

[55] Samala Rathan and G Naga Raju. «A modified fifth-order WENO scheme for hyperbolic conservation laws». In: *Computers & Mathematics with Applications* 75.5 (2018), pp. 1531–1549 (cit. on p. 28).

[56] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 3rd. Non–upwind (centred) schemes and Lax–Friedrichs flux. Springer, 2009, p. 601 (cit. on p. 28).

[57] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 3rd. Related to wave-speed estimates and pressure-based wave-speed estimates. Springer, 2009, p. 329 (cit. on p. 29).

[58] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 3rd. Monotonicity and the Lax-Friedrichs Scheme. Springer, 2009, p. 442 (cit. on p. 29).

[59] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 3rd. Summary of HLLC Fluxes and wave-speed estimates. Springer, 2009, pp. 331–332 (cit. on p. 30).

[60] Marko Čanađija. «2 - Kinematics of deformation». In: *Thermomechanics of Solids and Structures*. Ed. by Marko Čanađija. Elsevier Series on Plasticity of Materials. Elsevier, 2023, pp. 45–73. ISBN: 978-0-12-820121-3. DOI: https://doi.org/10.1016/B978-0-12-820121-3.00011-9. URL: https://www.sciencedirect.com/science/article/pii/B9780128201213000119 (cit. on p. 35).

[61] J.Y. Trépanier, M. Reggio, H. Zhang, and R. Camarero. «A finite-volume method for the Euler equations on arbitrary lagrangian-eulerian grids». In: *Computers Fluids* 20.4 (1991), pp. 399–409. ISSN: 0045-7930. DOI: https://doi.org/10.1016/0045-7930(91)90081-R. URL: https://www.sciencedirect.com/science/article/pii/004579309190081R (cit. on pp. 35, 37).

[62] Paul Dennis Thomas and Charles K Lombard. «Geometric conservation law and its application to flow computations on moving grids». In: *AIAA journal* 17.10 (1979), pp. 1030–1037 (cit. on p. 35).

[63] Alessandro Munafò, Andrea Alberti, Carlos Pantano, Jonathan B Freund, and Marco Panesi. «A computational model for nanosecond pulse laser-plasma interactions». In: *Journal of Computational Physics* 406 (2020), p. 109190 (cit. on p. 45).

[64] Alessandro Munafò and Marco Panesi. «Plato: a high-fidelity tool for multicomponent plasmas». In: *AIAA Aviation 2023 Forum*. 2023, p. 3490 (cit. on p. 45).

[65] David E Gildfind, Peter A Jacobs, Rowan J Gollan, Pierpaolo Toniato, Timothy J McIntyre, Andreas Andrianatos, Christopher M James, and Richard G Morgan. «Flow characterization and modeling of the X2 and X3 expansion tubes». In: (2018) (cit. on p. 46).

[66] *GCC, the GNU Compiler Collection*. `https://gcc.gnu.org`. Accessed: 2023. 2023 (cit. on p. 51).

[67] Apple Inc. *macOS*. `https://www.apple.com/macos/`. 2023 (cit. on p. 51).

[68] Linux Foundation. *Linux Operating System*. `https://www.linux.org/`. 2023 (cit. on p. 51).

[69] E. F. Toro. «Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction». In: 3rd. Berlin, Heidelberg: Springer, 2009. Chap. 10.8, pp. 334–344. ISBN: 978-3-540-25202-3. DOI: `10.1007/b79761`. URL: `https://doi.org/10.1007/b79761` (cit. on p. 56).

[70] John A. Webster. «Nozzle Design». In: *CIRP Encyclopedia of Production Engineering*. Ed. by Sami Chatti, Luc Laperrière, Gunther Reinhart, and Tullio Tolio. Berlin, Heidelberg: Springer Berlin Heidelberg, 2019, pp. 1276–1281. ISBN: 978-3-662-53120-4. DOI: `10.1007/978-3-662-53120-4_6432`. URL: `https://doi.org/10.1007/978-3-662-53120-4_6432` (cit. on p. 69).

[71] R. W. Whitside, W. Y. K. Chan, and M. K. Smart. *Flowpath design and validation of the Mach 6B nozzle for T4*. Research Report 2020/02. School of Mechanical & Mining Engineering, The University of Queensland, 2020 (cit. on p. 85).

[72] «Computational Fluid Dynamics 2000: Proceedings of the First International Conference on Computational Fluid Dynamics, ICCFD, Kyoto, Japan, 10-14 July 2000». In: ed. by Nobuyuki Satofuka. Berlin, Germany: Springer-Verlag Berlin Heidelberg GmbH, 2001, pp. 645–650. ISBN: 978-3-642-62560-2. DOI: `10.1007/978-3-642-56535-9` (cit. on p. 86).

[73] Yasuhiro Wada and Meng-Sing Liou. *A Flux Splitting Scheme With High-Resolution and Robustness for Discontinuities*. NASA Technical Memorandum NASA TM-106452, ICOMP-93-50, AIAA-94-0083. Presented at the 32nd Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January 10-13, 1994. NASA Lewis Research Center, 1994. URL: `https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19940032367.pdf` (cit. on p. 86).

[74] W. H. Beck, G. Eitelberg, T. J. McIntyre, J. P. Baird, J. Lacey, and H. Simon. «The high enthalpy shock tunnel in Göttingen (HEG)». In: *Shock Waves*. Ed. by Kazuyoshi Takayama. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992, pp. 677–682. ISBN: 978-3-642-77648-9 (cit. on pp. 88, 89).

[75] TJ McIntyre, JR Maus, ML Laster, and G Eitelberg. «Comparison of the flow in the High-enthalpy Shock Tunnel in Göttingen with numerical simulations». In: *Shock Waves@ Marseille I: Hypersonics, Shock Tube & Shock Tunnel Flow*. Springer. 1995, pp. 251–256 (cit. on pp. 88, 89).