POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell'Informazione
Corso di Laurea in Ingegneria Aeronautica

Development of a Lagrangian solver for thermochemical nonequilibrium flows

Relatore: Prof. Aldo Frezzotti
Co-relatore: Prof. Thierry Magin

Tesi di laurea di:
Stefano Boccelli - Matr. 820652

Anno Accademico 2015-2016

To Ermanno Bazzocchi,
designer of the plane that inspired me
since my very first day at PoliMi.

Abstract

In this work, a Lagrangian solver for nonequilibrium reacting flows is developed. The solver acts as a multi-temperatures chemical reactor following a fluid particle along pre-computed streamlines, integrating the governing equations with initial conditions picked from the first point of the streamline. The result is obtained in terms of temperatures and mass fractions of chemical species.

The developed Lagrangian reactor might be used to refine a previous rough numerical computation, introducing more sophisticated chemical mechanisms and optionally thermal nonequilibrium.

The Lagrangian solver is applied to the study of thermochemical relaxation past a 1D shock, to hypersonic argon flows over a cylinder at various degrees of rarefaction and to the problem of computing the recombination of free electrons in the trail of a meteoroid entering the Earth's atmosphere at hypersonic velocities.

Sommario

Il presente lavoro tratta lo sviluppo di un solutore Lagrangiano per correnti fuori equilibrio termico e chimico. Data una streamline proveniente da un calcolo eseguito tramite qualsivoglia metodo numerico, il solutore si comporta come un reattore chimico che integra le equazioni di governo seguendo la particella fluida. Le condizioni iniziali per l'integrazione vengono prese dal punto iniziale della streamline e la soluzione è fornita in termini di temperature e frazioni massiche delle specie chimiche.

Il solutore sviluppato può essere utilizzato per raffinare una precedente simulazione numerica semplificata, introducendo meccanismi chimici più elaborati ed opzionalmente nonequilibrio termico.

Nel presente lavoro, il solutore Lagrangiano viene applicato ad alcuni casi test: lo studio del rilassamento termochimico nella regione post-shock di un'urto normale, una corrente ipersonica di argon attorno ad un cilindro a vari gradi di rarefazione ed infine al calcolo della ricombinazione di elettroni liberi nella scia di un meteorite durante il suo ingresso atmosferico a velocità ipersonica.

Ringraziamenti

Ormai giunto al termine della mia avventura Politecnica, guardando indietro vedo un mare di persone che hanno incrociato la mia via. È grazie a tutti voi, grazie agli anni, ai mesi, o a volte anche solo alle poche ore che ho potuto passare con voi, che son diventato quel che sono e son potuto giungere fin qui.

Innanzitutto vorrei ringraziare il prof. Frezzotti sia per la grande disponibilità mostrata pur a vari kilometri di distanza, sia perchè suggerendomi di "farmi un giro" al von Karman Institute a lavorare su tematiche di rientro atmosferico mi ha veramente aperto un mondo. Ringrazio poi il prof. Magin per tutta la pazienza ed il tempo dedicato allo studentello completamente digiuno di tematiche ipersoniche quale ero fino a qualche mese fa.

Sono profondamente in debito verso tutti i professori che in questi anni di PoliMi ci hanno donato gioie e dolori, ma tre in particolare, i proff. Quartapelle, Boffadossi e Masarati meritano i miei ringraziamenti più sentiti. Ognuno a modo suo, hanno saputo marcare indelebilmente la mia carriera universitaria.

Vorrei ora ringraziare la mia famiglia per l'incessante supporto durato tutti questi anni, e Cecilia, alla quale devo anche riconoscere una dose di pazienza assolutamente eroica (hai davvero preso dalla nonna Giuseppina!)

Adesso è il momento di ringraziare tutti voi, valorosi compagni Aerospaziali, che con me avete ridotto l'universo intero ad un sistema "massa - molla - smorzatore" e con me avete sognato un futuro luminoso in cui l'amore e la passione per quel che studiamo spingano sempre più in là i nostri limiti. Un immenso grazie a Davide D.C.B, al Da' ed a Davide B. Grazie poi a (Francesco)Dario C, a Dario C, ed a Dario B. Anche se sfuggono alla regola dei tre nomi, grazie anche al grande Gigi ed al mio consulente personale e grillo parlante, il super-Gale.

Ancora qualche riga per mandare il mio pensiero a tutti i compagni stageurs al VKI. Grazie a tutti voi ogni singola goccia di sudore spesa su questa tesi è stata accompagnata da un sorriso. Grazie ai compagni di Little Italy, in particolare a Soudri il fiorentino, per tutte le nottate a lavorare "come se non ci fosse un domani". Ma quel domani poi arrivava sempre, insieme con due belle occhiaie... Grazie poi alla combriccola del Computer Center. Rahand, Maria.. e soprattutto a te Nishant, che tra l'altro mi hai fatto scoprire la costellazione di Perseo, da cui arriva il meteorite simulato in questa tesi! E grazie a te, principessa Kamila Bubblegum, compagna di esperimenti improbabili e futura coautrice di esilaranti articoli. Grazie infine ai PhD Georgios, Bruno ed a Tamás, per il vostro supporto sia pratico che morale.

Riservo l'ultimo ringraziamento, il più grande di tutti, al mio PhD advisor Federico, vero angelo custode dal primo giorno al VKI fino all'ultimo.

Milano, Settembre 2016

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

This thesis was developed during an internship at von Karman Institute for Fluid Dynamics, Rhode-Saint-Genèse, Belgium.

1.1 Motivation

Studying hypersonic flows requires dealing with a variety of physical phenomena experienced at the microscopic level by atoms and molecules, often leading to results that are very different from the classical solutions of supersonic aerodynamics. Those phenomena include chemical reactions, excitation of vibrational degrees of freedom, ionization, radiation and the challenges are twofold:

i) modeling phase presents non-negligible difficulties, also due to the fact that experimental data are highly inaccurate due to the extreme conditions that should be reproduced in experiments

ii) simulation phase currently requires way too many computational efforts to perform a complete modeling of the physics.

Nowadays, the mostly used numerical tools are the Direct Simulation Monte Carlo method (DSMC), for rarefied flows, and CFD methods, to be applied in the continuum regime. Both methods suffer from the limitations mentioned above: for instance, the computational cost of the DSMC method becomes prohibitive when interactions among particles involve too many chemical reactions. Simulating many chemical species decreases the accuracy of DSMC simulations, in that longer time averages or a higher number of simulated particles are needed to obtain a smooth-enough solution. The second point is

also the Achilles' heel of CFD methods, that are forced to drastically simplify the physics of the problem since simulating too many species or internal temperatures would lead to large algebraic systems to be solved.

Finally, DSMC simulations suffer from difficulties in modeling three body collisions, that also leads to difficulties in implementing reassociation reactions.

This thesis is aimed at developing a solver that might correct the mentioned modeling deficiencies of numerical simulations, by partially recomputing results introducing more elaborate chemical or thermal modeling.

1.2 Aim of the work

This work is aimed at developing a Lagrangian solver for hypersonic flows out of equilibrium, able to implement arbitrarily complicated chemical models and various degrees of internal nonequilibrium. In order to do that, the solver does not provide the solution of the governing equations on a whole multidimensional domain, but starts from a baseline simulation and recomputes the results along a given streamline by introducing new chemical mechanisms and internal temperatures.

The governing equations along the streamline are a set of ODEs and the solver acquires a particularly lightweight structure, so that all the saved computational time can be invested in solving a more complicated thermochemical model.

This solver is based on the hypothesis that the velocity field may be assumed as given from the baseline reference simulation and as a result, also the density field will be taken as given, as explained in section 3.2. This hypothesis is quite strong, in that modifications in the thermochemical model actually *do* reflect into changes in the flow field. However, the improvements to the baseline flow obtained with the Lagrangian solver proved to yield sound results, indicating that the velocity and density fields are not-so-tightly coupled with the thermochemical description of the flow.

To sum up, the goal of the work is developing a solver that, at the price of decoupling the velocity field, is able to refine a baseline solution along a streamline, by including the chemistry of previously neglected species and/or internal nonequilibrium.

1.3 Atmospheric entry of meteoroids

Among the bodies entering the Earth's atmosphere, meteoroids are probably those experiencing the most extreme conditions. Modeling the air flow around them is a non-trivial task using the standard tools used in the Aerospace field and requires taking into account virtually all the phenomena reviewed in the first part of this introduction. In this work, a first attempt of modeling this problem is done with the developed Lagrangian solver.

To give an idea of the quantities involved, should be noted that the typical entry velocity of meteoroids is higher than manned re-entry vehicles (around 11 km/s for Lunar missions), reaching up to 72 km/s for the Perseids meteor shower and their ablation is very intense, usually completely destroying the meteoroid before it can reach the ground. The quantity of meteoroids entering the atmosphere is also massive, estimated to be around 50 tonnes a day, the size of incoming bodies ranging from micrometers to some centimeters in diameter.

Many numerical and experimental efforts[1] are currently underway to characterize the plasma flow that develops around meteoroids entering the atmosphere, the main goal being correlating the parameters of the incoming body (size, velocity, composition etc) to ground-based radar observations.[2] The high velocities, altogether with strong ablation, may lead to the creation of a possibly very long ionized trail past the meteor, deeply altering the radio echo received by ground stations.

A complete approach in simulating the ionized trail of the meteor at high altitudes would be based on methods coupling Particle In Cell simulations and Monte Carlo methods able to treat collisions, including some ablation modeling of the surface. However those methods are very computationally expensive and more efficient numerical schemes might be desirable. The Direct Simulation Monte Carlo method, widely used in the Aerospace field for simulating rarefied atmospheric entry flows, is a natural candidate for studying the problem. This method suffers from some limitations, the most severe of which is the numerical difficulty in implementing recombination of free electrons, that might be a driving factor in the developing trail.

[1] At von Karman Institute for Fluid Dynamics, characterization of ablating chondrites is being done in the Plasmatron facility.

[2] One effort in radar detection of meteors is the BRAMS network, based in Belgium, currently counting around 30 antennas spread all over the country. The BRAMS network relies on forward-scattering techniques and detects around 1500 meteors echos per day.

In this work, a DSMC simulation of a non-ablating meteor at high altitudes is performed with the software *SPARTA*[3] and the results are then refined with the developed Lagrangian solver to artificially introduce recombination.

1.4 Structure of the thesis

This thesis develops as follows: first of all some physical models are reviewed in chapter 2, introducing the Boltzmann equation, internal degrees of freedom of atoms and molecules, the N-temperatures approximation and collisional models used in Direct Simulation Monte Carlo codes.

In chapter 3, constitutive equations for nonequilibrium flows are introduced and recast in Lagrangian form. The obtained ODE system constitutes the set of equations that are solved by the developed Lagrangian solver.

Chapter 4 reviews numerical tools exploited in this work: a thermodynamic library, a solver for relaxation past a 1D shockwave and a DSMC code.

The core of this thesis, the Lagrangian solver, is introduced in chapter 5 first of all reviewing its numerical implementation and then verifying it with simple testcases.

After the implementation and verification step, the solver is applied to practical cases (chapter 6), starting with the relaxation past a 1D shock, where more detailed chemistry is introduced, as well as thermal nonequilibrium using the two-temperatures model. The solver is then applied to some simple DSMC simulations of argon flows at various Knudsen numbers, and finally to the problem of computing the recombination of free electrons in the ionized trail of a meteoroid entering the Earth's atmosphere at high altitude and velocity.

In the end, conclusions are drawn in chapter 7, and some directions for further improvements and applications are indicated.

[3] `http://sparta.sandia.gov`

CHAPTER 2

Physico-chemical models

2.1 Modeling rarefied flows

When dealing with rarefied flows, the fluid dynamics equations based on continuum approach progressively lose their validity and a more general model is to be adopted. A practical way to characterize the degree of rarefaction of a flow is provided by the *Knudsen number*, ratio between the mean free path of the gas molecules and a characteristic dimension of the flow:

$$\mathrm{Kn} = \frac{\lambda}{L} \tag{2.1}$$

Flows with Knudsen numbers $\mathrm{Kn} < 0.01$ can be thought as following the continuum equations, while flows with $\mathrm{Kn} > 10$ are usually called free molecular flows and are characterized by a very low number of collisions among particles. In the middle, the actual behavior of the fluid gradually deviates from the classical gas dynamics equations based on the continuum hypothesis.

The Knudsen number as defined in equation 2.1 is a little too coarse to fully describe a flowfield, since it does not take into account that the fluid might *locally* break the continuum hypothesis. A local version of the Knudsen number may then be written, based on gradients of some properties Q of the flow:

$$\mathrm{Kn} = \frac{\lambda}{Q}\left|\frac{\mathrm{d}Q}{\mathrm{d}x}\right| \tag{2.2}$$

If dealing with rarefied flows, a new approach to the problem is needed. Macroscopic constitutive laws can still be obtained, but starting from statistical mechanics and working in the framework of the phase space. An evolutive equation can be obtained for the distribution function $f(\boldsymbol{r}, \boldsymbol{c}, t)$, describing

the probability density that a particle having a velocity between $\boldsymbol{c}$ and $\boldsymbol{c} + \mathrm{d}\boldsymbol{c}$ is located between $\boldsymbol{r}$ and $\boldsymbol{r} + \mathrm{d}\boldsymbol{r}$ at the time t. This equation is the *Boltzmann equation* and is valid at whatever Knudsen number for gases in or out of equilibrium:

$$\frac{\partial}{\partial t}(nf) \; + \; c_j \; \frac{\partial}{\partial r_j}(nf) \; + \; F_j \; \frac{\partial}{\partial c_j}(nf) = \Gamma(nf) \tag{2.3}$$

This equation is a nonlinear integro-differential equation, its integral and nonlinear part being defined inside the term $\Gamma(nf)$, called *collision integral*. By describing the evolution for the distribution function, the Boltzmann equation describes the state of a system of particles even out of equilibrium.

The macroscopic variables used in conservation equations (density, momentum and energy of the fluid particle) can be derived as *moments* of the distribution function, and the conservation laws itself can be obtained by computing moments of the Boltzmann equation.

2.2 Thermodynamics

Due to the high velocities and very steep gradients involved, hypersonic flows are characterized by a variety of phenomena that do not show in supersonic flows at Mach numbers lower than 5. The high energies involved are enough to excite internal degrees of freedom such as vibrational and electronic, that would be frozen or very poorly excited at standard temperatures. In many cases, since the process of excitation requires a certain number of collisions, a new dynamics emerge, whose times are comparable to the characteristic times of the flow, thus generating in the domain regions of thermal *nonequilibrium*. Translational equilibrium is the quickest to be reached, requiring only a few molecular collisions, rotational is a little slower, while vibrational is much slower and electronic degrees of freedom take even more time.

The energy of a generic molecule can be written as the sum of its translational, rotational, vibrational and electronic energies and the fact that they can be in a state of nonequilibrium among each other induces us to introduce different temperatures for each one of them. It should be noted that atoms do not have rotational or vibrational energies and that free electrons can be seen as a particular chemical species possessing only kinetic (translational) energy. In general we could attribute different internal temperatures to different chemical species, so that by denoting atoms with the set of indices $\mathcal{A}$ and molecules

with $\mathcal{M}$:

$$\begin{array}{rcccl} i \in \mathcal{M} & \rightarrow & e_i & = & e_i^{tr}(T) + e_i^{rot}(T_i^r) + e_i^{vib}(T_i^v) + e_i^{el}(T_i^e) + e_i^{form} \\ i \in \mathcal{A} & \rightarrow & e_i & = & e_i^{tr}(T) + e_i^{el}(T_i^e) + e_i^{form} \\ \text{free electrons} & \rightarrow & e_e & = & e_e^{tr}(T_e) + e_e^{form} \end{array}$$

where the translational temperature T was supposed common among all the heavy species (atoms and molecules), since translational equilibrium is extremely quickly reached. Note that although electrons have only translational energy, their temperature is better supposed out of equilibrium since they are very coupled to the most internal degrees of freedom such as electronic and vibrational. Such a model is very general, too much to be used in many practical CFD computations, so that it is customary to merge together some internal temperatures, giving rise to the so called N-temperatures models, that retain the dependency on N temperatures only.

Two-temperatures model

Among the N-temperatures models, the "two-temperatures" is perhaps the most famous, consisting in assuming that translational and rotational degrees of freedom are in equilibrium among each other at temperature T, whereas vibrational, electronic and the kinetic energy of free electrons are in equilibrium among each others at temperature T_v:

$$T = T_r \quad , \quad T_v = T_{el} = T_e \tag{2.4}$$

Apart from its simplicity, this model gained wide usage since it is based on the fact that rotational degrees of freedom reach equilibrium much more easily than the vibrational ones.

Relaxation towards equilibrium

Molecular collisions provide a mean to equilibrate the internal degrees of freedom, so that equilibrium is eventually reached. Describing the energy transfer among internal modes on a rigorous basis is a difficult task, usually addressed through some modeling. A commonly used model is from Landau and Teller, modeling the approach to equilibrium as a first order system. For vibrational energy for example:

$$\frac{\mathrm{d}e_v}{\mathrm{d}t} \sim \frac{e_v(T) - e_v(T_v)}{\tau_v} \tag{2.5}$$

where τ is a relaxation time, that may be expressed with the Millikan-White approach with high-temperature correction by Park. A nice review of the Landau-Teller model is given by Nikitin and Troe [10].

2.3 Physical models for DSMC

This sections briefly overviews models employed in the DSMC simulations performed in this thesis.

Elastic collisions

Collisions in the DSMC model are performed among particles belonging to the same cell, testing a certain number of pairs and actually performing the collisions if a condition over the relative velocities and the cross sections is met. Modeling the cross sections is the main issue and some models have been developed in the years, the simplest being the Hard Sphere model (named "HS"), that treating the particles as rigid spheres predicts a total cross section $\sigma = \pi d_{12}^2$, with d_{12} being the average of the spheres diameters. The hypothesis of constant cross section reveals to be quite heavy and leads to wrong results on macroscopic transport properties. The cross section is actually found to *decrease* with the relative velocity of the colliding pair and this results yields to the formulation of a more accurate model, named Variable Hard Sphere (VHS). This model includes the dependency on the relative velocity of the colliding pair in the definition of the sphere diameter: $d = d_{ref} \left(c_{r,ref}/c_r\right)^{\nu}$. The scattering law of this model is still isotropic, as for the simpler HS model.

The model used in this work is the Variable Soft Sphere (VSS), a modification of the VHS approximation that models the deflection angle with the law: $\chi = 2\cos^{-1}\left[(b/d)^{1/\alpha}\right]$. For more details, the reader can refer to Bird [4].

From a practical point of view, the VSS model requires four parameters to be specified for each species:

- reference diameter
- reference temperature
- viscosity coefficient ω
- coefficient α

the VHS model being the subcase with $\alpha = 1$. The following input file for the DSMC code *SPARTA* used in this work (`http://sparta.sandia.gov`) resumes all the VSS parameters used in the simulations of this work:

```
# Syntax:
# -   diameter [m]   omega   Tref    alpha
# Argon:
Ar    3.595E-10      0.734   1000    1
# Air11:
O2    3.96E-10       0.77    273.15  1.4
N2    4.07E-10       0.74    273.15  1.6
O     3.0E-10        0.80    273.15  1.0
N     3.0E-10        0.80    273.15  1.0
NO    4.0E-10        0.80    273.15  1.0
O2+   3.96E-10       0.77    273.15  1.4
N2+   4.07E-10       0.74    273.15  1.6
O+    3.0E-10        0.80    273.15  1.0
N+    3.0E-10        0.80    273.15  1.0
NO+   4.0E-10        0.80    273.15  1.0
e     7.0E-13        0.50    273.15  1.0
```

Inelastic collisions and chemistry

During collisions, part or all the energy of the colliding pairs might be transferred to internal energy as excitation of rotational, vibrational or electronic modes. The most common method implemented in DSMC is the Larsen-Borgnakke method (see Bird [4]), a phenomenological approach that matches experimental relaxation times by tuning the energy transfer from translational to rotational or vibrational modes. Taking into account exchanges of energy among internal modes is not straightforward and is neglected in this approach, as well as the excitation of electronic levels.

A more detailed description might be needed if accurate results are to be obtained for high enthalpy flows such as those encountered in atmospheric entry conditions. The so called "state-to-state" models provide such a level of accuracy, see for example Bruno [14]. Those models have been developed for some mixtures and describe internal modes as pseudo-species and the energy exchange is provided via pseudo-chemical reactions.

Another type of inelastic collisions are those that involve chemical reactions, in which part of the colliding energy is used to destroy (or create) chem-

ical bonds. There are mainly two models used in DSMC simulations: the TCE (Total Collision Energy, see Bird [4]) and the Q-K (Quantum-Kinetic, see Bird [21]) models. This work uses the TCE model, that computes collisions on the basis of the kinetic energy associated to the relative velocity among the colliding pair. In the hypothesis of a Maxwellian velocity distribution, cross-sections are recast into rate coefficients in Arrhenius form.

The set of rates adopted in this work for air chemistry is reported in appendix B.

Ambipolar assumption

DSMC methods are able to treat ionizing collisions among chemical species. The resulting electrons can be treated just like any other chemical species, moving in the domain, colliding and experiencing chemical reactions. However, the simulation of ionized flows poses a serious difficulty: since the mass of electrons is way smaller than that of heavy species, the mean velocity is much higher and as a result the timestep required to track electrons becomes extremely small, too much for common simulations to be carried out.

Fortunately, weakly ionized plasmas have the tendency of being neutral over a length that exceed the Debye length, since the electric field that generates from a net charge distribution in space tend to keep electrons and ions close to each other. The assumption that ions and electrons stay relatively close together is called "ambipolar assumption", see Bellan [5].

The ambipolar assumption is implemented into DSMC codes by tying dissociated electrons to their parent ions during the *moving phase* of the algorithm. This does not affect the energy of free electrons (and their temperature), that is stored and used to compute the effect of collisions.

CHAPTER 3

Governing equations for the fluid model

In this chapter, governing equations for nonequilibrium chemically reacting flows are introduced. After showing the general set of conservation equations for nonequilibrium flows in PDE form (section 3.1), the set of equations used by the Lagrangian solver developed in this thesis is introduced in section 3.2. Finally, the inviscid set of equations to solve the relaxation past a 1D shock (section 3.3) is quickly introduced. This last set of equations is used to obtain a reference solution over which verifying the Lagrangian solver implementation.

The derivation of the basic conservation equations will not be shown. The reader can refer to classical textbooks on the subject, such as Anderson [1], Vincenti and Kruger [2] and Park [3]. The derivation of Lagrangian equations is only sketched in this chapter, while more details can be found in appendix A.

3.1 Conservation equations

The description of a mixture in thermochemical nonequilibrium can be obtained starting from the Maxwell transfer equations. Those equations are valid both in the continuum and rarefied regimes, the main difficulty being represented by the need of an accurate closure for the transport fluxes. The well known set of Navier-Stokes equations is a particular case of the Maxwell transfer equations, with a specific closure.

Set of Maxwell transfer equations

The Maxwell transfer equations may be expressed as the set of the density ρ,

momentum $\rho\boldsymbol{u}$ and the total energy E conservation equations:

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{u}) = 0 \tag{3.1}$$

$$\frac{\partial \rho \boldsymbol{u}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u} + P\mathbb{I}) = \boldsymbol{\nabla} \cdot \boldsymbol{\varpi} \tag{3.2}$$

$$\frac{\partial \rho E}{\partial t} + \boldsymbol{\nabla} \cdot [\boldsymbol{u}\,(P + \rho E)] = -\boldsymbol{\nabla} \cdot \boldsymbol{q} + \boldsymbol{\nabla} \cdot (\boldsymbol{u} \cdot \boldsymbol{\varpi}) - Q^{rad} \tag{3.3}$$

where Q^{rad} is the energy lost (or gained) as radiation and E is the total energy per unit mass, sum of the internal and kinetic energy per unit mass of the flow: $E = e + u^2/2$. The quantity P is the pressure, $\mathbb{I}$ the identity matrix, $\boldsymbol{\varpi}$ the viscous shear stress tensor and $\boldsymbol{q}$ the heat flux vector.

Mass conservation for chemical species

To the set of Maxwell transfer equations, a balance equation for each chemical species is added:

$$\frac{\partial \rho_i}{\partial t} + \boldsymbol{\nabla} \cdot (\rho_i \boldsymbol{u} + \boldsymbol{J}_i) = \dot{\omega}_i \tag{3.4}$$

where $\dot{\omega}_i$ is the rate of production for the *i-th* chemical species and can be expressed with the *law of mass action*. The term J_i represents the mass flow due to Fick's diffusion, arising from the decomposition $\rho\boldsymbol{u}_i = \rho\boldsymbol{u} + \boldsymbol{J}_i$.

Vibrational, rotational and electronic energy

The balance equations for the vibrational energy of the *i-th* chemical species is easily obtained from conservation principles, reading:

$$\frac{\partial \rho_i e_i^v}{\partial t} + \boldsymbol{\nabla} \cdot (\rho_i e_i^v \boldsymbol{u}) = -\boldsymbol{\nabla} \cdot (e_i^v \boldsymbol{J}_i + \boldsymbol{q}_i^v) + \Omega_i^v \ , \qquad i \in \mathcal{M} \tag{3.5}$$

where:

$\boldsymbol{\nabla} \cdot (e_i^v \boldsymbol{J}_i)$ is the flux of vibrational energy due to mass flow

$\boldsymbol{\nabla} \cdot \boldsymbol{q}_i^v$ is the conductive flux of vibrational energy

Ω_i^v the rate of exchange of energy between vibrational and the other degrees of freedom.[1] This term requires modeling, see section 2.2

[1] Chemistry included, since the vibrational level of a species may change after the occurrence of a chemical reaction.

$\mathcal{M}$ is the set of *molecular* species

For the rotational and electronic internal energies, the equation is formally the same as for the vibrational case, with the only exception that the electronic energy applies to *any* species, not only to molecular ones:

$$\frac{\partial \rho_i e_i^r}{\partial t} + \boldsymbol{\nabla} \cdot (\rho_i e_i^r \boldsymbol{u}) = -\boldsymbol{\nabla} \cdot (e_i^r \boldsymbol{J}_i + \boldsymbol{q}_i^r) + \Omega_i^r \ , \qquad i \in \mathcal{M} \tag{3.6}$$

$$\frac{\partial \rho_i e_i^{el}}{\partial t} + \boldsymbol{\nabla} \cdot \left(\rho_i e_i^{el} \boldsymbol{u}\right) = -\boldsymbol{\nabla} \cdot \left(e_i^{el} \boldsymbol{J}_i + \boldsymbol{q}_i^{el}\right) + \Omega_i^{el} \ , \qquad i \in \mathcal{S} \tag{3.7}$$

with obvious meaning of the superscripts.

Free electrons translational energy

Obtaining an equation for the free electrons translational energy e_e^t is less trivial than the other internal energies and relies on some strong hypothesis. It should be noted that a charge distribution in space generates an electric field and thus a force on the electrons, that will produce work. Also, charges experience bremsstrahlung when accelerated, but this effect will be neglected.

Moreover, textbooks usually write an equation for the "electron-electronic" energy, by merging the kinetic energy of free electrons and the electronic energy of atoms and molecules, while for our goals the two contributions should be kept separated. The equation will thus be here introduced in a slightly more detailed manner.

Following Park [3], first of all the momentum equation is written by neglecting viscosity:

$$\rho_e \frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} = -\boldsymbol{\nabla} P_e - \sum_{i \in \mathcal{H}} \frac{m_e}{m_i} N_i \nu_i \left(\boldsymbol{U}_i - \boldsymbol{U}_e\right) - e N_e \boldsymbol{E} \tag{3.8}$$

where $\boldsymbol{u}$ is the mixture velocity, $\boldsymbol{U}_i - \boldsymbol{U}_e$ is the difference in diffusion velocity of the *i-th* heavy species and free electrons, ν_i is the collision rate, e the electron charge, N are number densities (number of particles per cubic meter) and $\boldsymbol{E}$ is the electric field (self-generated plus externally applied). Here, the syntax $\mathrm{D}/\mathrm{D}t$ represents the *material derivative*.

In the hypothesis of weakly ionized plasma, some simplifications are made. First of all, the term ρ_e is supposed small, so that the left-hand term vanishes, then the difference in diffusion velocities is also supposed small since it would

represent an electric current. A simple equation for the electric field is then obtained:

$$eN_e\boldsymbol{E} = -\boldsymbol{\nabla}P_e \tag{3.9}$$

The equation for free electrons kinetic energy is then written by applying the conservation to a control volume $V(t)$ that is moving with the mean velocity of free electrons $\boldsymbol{u}_e$:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_V \rho_e e_e^t \,\mathrm{d}V = -\oint_{\partial V} P\hat{\boldsymbol{n}}\cdot\boldsymbol{u}_e \mathrm{d}S - \oint_{\partial V}\boldsymbol{q}_e\cdot\hat{\boldsymbol{n}}\,\mathrm{d}S - \int_V eN_e\boldsymbol{E}\cdot\boldsymbol{u}_e\,\mathrm{d}V + \int_V \Omega_e\,\mathrm{d}V \tag{3.10}$$

where the power of pressure forces has been taken into account since the volume is moving. From here, the approximated result for the electric field can be used, as well as simplifying hypothesis based on the weak ionization of the plasma, finally leading to an equation for the free electrons translational energy:

$$\frac{\partial \rho_e e_e^t}{\partial t} + \boldsymbol{\nabla}\cdot\left(\rho_e e_e^t \boldsymbol{u}\right) = -P_e\boldsymbol{\nabla}\cdot\boldsymbol{u} - \boldsymbol{\nabla}\cdot\boldsymbol{q}_e + \Omega_e \tag{3.11}$$

Equation of state

Finally, the equation of state for a multispecies mixture of perfect gases reads:

$$P = \sum_{i\in\mathcal{H}} \rho_i \frac{\mathcal{R}}{\mathcal{M}_i}T + \rho_e\frac{\mathcal{R}}{\mathcal{M}_e}T_e \tag{3.12}$$

Because of the high temperatures involved in atmospheric entry flows and the high degree of rarefaction, the hypothesis of perfect gas holds with great precision.

3.2 Lagrangian approach

As seen in section 1.2, this work deals with the creation of a Lagrangian solver that starting from a computed flow field may recompute the chemistry and the temperatures in the framework of a multi-temperature fluid model. Since test cases are carried out using the equilibrium assumption or the two-temperatures model, equations here are shown only for those two cases, while the Lagrangian equations for the more general N-temperatures model are left to appendix A, altogether with a more detailed derivation.

The basic assumption underlying this work is that the velocity field $\boldsymbol{u}(\boldsymbol{r}, t)$ can be taken as given. From the mass conservation written for the whole mixture (eq. 3.1), for a given velocity field it's possible to compute the density field directly, with no need of taking into consideration any more equation. For this reason, also the density field is chosen to be "externally given", in that recomputing it with equation 3.1 would return the same result, altogether with additional numerical error.[2] This implies that from the set of conservation equations shown in section 3.1, the mass and momentum equations for the whole mixture are not necessary anymore. The governing equations for the Lagrangian solver are then:

i) one mass conservation equation for each chemical species

ii) the total energy conservation equation

iii) one energy equation for each internal energy

In the special cases of thermal equilibrium, equations in *iii)* are not needed, while in the case of two-temperatures model they are merged into one only equation.

The first step is reformulating the required equations in a Lagrangian framework following the fluid particle, by introducing the material derivative and thus expressing the rate of variation of quantities *along* the streamline. Additionally, for energy equations the chain rule is applied to obtain an equation for each internal temperature and for the translational one.

It should be noted that in a system of reference moving with the particle, quantities are function only of the *Lagrangian time* τ, or function of the *curvilinear abscissa* s along the streamline. With a slight notational abuse[3] we can write for example:

$$\frac{\mathrm{D}y_i(\boldsymbol{r}, t)}{\mathrm{D}t} \equiv \frac{\mathrm{d}y_i(s(\tau))}{\mathrm{d}s} \tag{3.13}$$

[2] One should recall that even considering high Knudsen numbers, the mass conservation equation retains the same shape. Even for rarefied flows then, the density field from a well-converged DSMC simulation will satisfy equation 3.1.

[3] Formally, we should denote differently quantities in *Eulerian* and *Lagrangian* formulation. In the example, we should have written $y_i(\boldsymbol{r}, t) = \tilde{y}_i(s(\tau))$, the dash denoting that we are handling a different mathematical function.

and the same holds for any other quantity. The hypothesis that the velocity field is given, thus, turns the conservation equations into a system of ODEs.

Mass conservation for chemical species

The mass fraction $y_i = \rho_i/\rho$ is introduced into eq.3.4, leading to:

$$\frac{\mathrm{D}y_i}{\mathrm{D}t} = \frac{1}{\rho}\left(\dot{\omega}_i - \boldsymbol{\nabla}\cdot\boldsymbol{J}_i\right) \tag{3.14}$$

Internal temperature T_v

In the framework of a two-temperatures model (see section 2.2), an equation for the internal temperature T_v can be obtained by merging together the vibrational, electronic and free-electrons enthalpies, each function of T_v only.

On conservational grounds, an equation for the enthalpy of internal degrees of freedom h^{in} can be written for each chemical species i, reading:

$$\frac{\mathrm{D}h_i^{in}}{\mathrm{D}t} = \frac{\boldsymbol{\nabla}\cdot(\boldsymbol{q}_i^{in} + h_i^{in}\boldsymbol{J}_i) + \Omega_i^{in} - h_i^{in}\dot{\omega}_i}{\rho y_i} \tag{3.15}$$

Note that for molecular species, h_i^{in} is the *sum* of the vibrational and electronic internal enthalpies, while for free electrons it is represented by their translational enthalpy.
The internal temperature T_v is shown to follow the following relation (see appendix A):

$$\frac{\mathrm{D}T_v}{\mathrm{D}t} = \left(\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{D}h_i^{in}}{\mathrm{D}t}\right) \Big/ \left(\sum_{i\in\mathcal{S}} y_i c_{p,i}^{in}\right) \tag{3.16}$$

that, by exploiting the previous equation in the hypothesis of null heat flux and diffusion, leads to the *internal temperature equation*:

$$\frac{\mathrm{D}T_v}{\mathrm{D}t} = \left(\frac{\Omega^{in} - \sum_{i\in\mathcal{S}} \dot{\omega}_i h_i^{in}}{\rho}\right) \Big/ \left(\sum_{i\in\mathcal{S}} y_i c_{p,i}^{in}\right) \tag{3.17}$$

Translational temperature T

An equation for the translational temperature T of the mixture is obtained starting from the total energy equation. First of all equation 3.3 is written

by introducing the per unit mass enthalpy $H = E + P/\rho$, then the material derivative is introduced. The hypothesis of *stationary* flow is then made, leading to:

$$\frac{\mathrm{D}H}{\mathrm{D}t} = -\boldsymbol{\nabla}\cdot\boldsymbol{q} + \boldsymbol{\nabla}\cdot(\boldsymbol{u}\cdot\boldsymbol{\tau}) - Q^{rad} = \mathcal{Q} \tag{3.18}$$

the term $\mathcal{Q}$ being the amount of energy entering or leaving the fluid particle because of heat flux, viscous stresses or radiation.[4] The hypothesis of *adiabatic* fluid particle may be useful in some situations and would lead to the simple equation $\mathrm{D}H/\mathrm{D}t = 0$, however such a case heavily fails in many practical circumstances, such as near surfaces, in expansion regions and inside shock regions.

Since the heat flux is a function of the temperature gradients, computing it in the framework of a Lagrangian solver is not trivial at all. In this work, the energy flux $\mathcal{Q}$ is taken "as given", just as the velocity and density fields, from the reference simulation. The hypothesis of taking the energy flux from outside might at a first glance seem unrealistic, in that the temperature gradients are by definition based on the temperature itself, that is an unknown of the Lagrangian problem. However in section 6.2 the hypothesis is shown to be a *great* improvement with respect to considering the particle as adiabatic.

First of all the total enthalpy is computed from the reference simulation, then its derivative is computed along the streamline, leading to an estimation for $\mathcal{Q}$:

$$\mathcal{Q} = \left(\frac{\mathrm{D}H}{\mathrm{D}t}\right)_{ref} \tag{3.19}$$

the subscript *ref* indicating that the value comes from the reference simulation. In order to obtain an equation for the translational temperature T, the total enthalpy is expanded into its contributions. In the context of a two-temperatures model:

$$\frac{\mathrm{D}H}{\mathrm{D}t} = \frac{\mathrm{D}h^t}{\mathrm{D}t} + \frac{\mathrm{D}h^{in}}{\mathrm{D}t} + \frac{\mathrm{D}u^2/2}{\mathrm{D}t} = \mathcal{Q} \tag{3.20}$$

The enthalpies can then be written as a sum over the chemical species and after some manipulations (see appendix A), the equation for the translational

[4] Radiation is not explicitly considered in this work, however the term $\mathcal{Q}$ is introduced in such a way that it's not necessary to specify the physical origin of the energy flux.

temperature in the framework of a two-temperature model is obtained:

$$\frac{\mathrm{D}T}{\mathrm{D}t} = \left[\mathcal{Q} - \frac{\mathrm{D}u^2/2}{\mathrm{D}t} - \sum_{i\in\mathcal{S}} \frac{h_i\dot{\omega}_i}{\rho} - \left(\Omega^{in} - \sum_{i\in\mathcal{S}} \frac{h_i^{in}\dot{\omega}_i}{\rho}\right)\right] \Big/ \left[\sum_{i\in\mathcal{S}} y_i c_{p,i}^t\right] \quad (3.21)$$

Care should be taken when computing enthalpies (and specific heats) of internal degrees of freedom, recalling that atoms have zero vibrational enthalpy and that the translational enthalpy of free electrons is to be inserted in h^{in} and not in h.

A temperature equation for the case of equilibrium flows can be obtained removing the term in round brackets from equation 3.21 and substituting the specific heats $c_{p,i}^t$ for translational DOFs with the total specific heats $c_{p,i}$:

$$\frac{\mathrm{D}T}{\mathrm{D}t} = \left[\mathcal{Q} - \frac{\mathrm{D}u^2/2}{\mathrm{D}t} - \sum_{i\in\mathcal{S}} \frac{h_i\dot{\omega}_i}{\rho}\right] \Big/ \left[\sum_{i\in\mathcal{S}} y_i c_{p,i}\right] \quad (3.22)$$

3.3 One-dimensional shock tube

In sections 5.2 and 6.1 the Lagrangian solver is applied to the case of thermochemical relaxation past a 1-D shockwave. The reference solution is provided by the software *Shocking*, developed at von Karman Institute, that in its most general implementation is able to solve the set of inviscid conservation equations here shown:

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\rho_i u\right) = \dot{\omega}_i \quad (3.23)$$

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\rho u^2 + p\right) = 0 \quad (3.24)$$

$$\frac{\mathrm{d}}{\mathrm{d}x}\left[\rho u\left(h + \tfrac{1}{2}u^2\right)\right] = 0 \quad (3.25)$$

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\rho u y_i e_i^v\right) = \dot{\omega}_i e_i^v + \Omega_i^v \quad , \quad i \in \mathcal{M} \quad (3.26)$$

$$\frac{\mathrm{d}}{\mathrm{d}x}[\rho u(y_e e_e + \sum_{i\in\mathcal{H}} y_i e_i^{el})] = -P_e\frac{\mathrm{d}u}{\mathrm{d}x} + \dot{\omega}_e e_e + \sum_{i\in\mathcal{H}} \dot{\omega}_i e_i^{el} + \Omega^{el} \quad (3.27)$$

By following the derivation of Thivet [17], the system can then be transformed into a system of ODEs.

CHAPTER 4

Numerical Tools

This chapter reviews the numerical tools used in this work, namely the *Mutation++* thermodynamic library for ionized gases, the *Shocking* code used to compute thermochemical relaxation past 1D shockwaves and the *SPARTA* Direct Simulation Monte Carlo code.

4.1 Mutation++: thermodynamic library

The *Mutation++* library (Multicomponent Thermodynamic And Transport properties for Ionized gases in C++) is a thermodynamic library developed at von Karman Institute for Fluid Dynamics, with the goal of providing efficient algorithms for obtaining thermodynamic and transport properties of nonequilibrium mixtures. A beta version can be obtained at `www.mutationpp.org`.

The library is repeatedly called by the Lagrangian solver implemented in this work, to obtain:

- Thermodynamic properties of mixture constituents such as enthalpies, energies and specific heats
- Chemical production rates
- Terms of energy transfer among degrees of freedom

The thermodynamic model chosen in this work is that of the "Rigid Rotor - Harmonic Oscillator", whereas *Mutation++* also implements two NASA polynomial databases. Chemical rates are given to *Mutation++* in Arrhenius form and the energy transfer terms are based on a Landau-Teller approach, with relaxation time given by the Millikan-White model including the Park correction.

From a practical point of view, the usage of the library starts by specifying the atomic composition of the mixture (number of nuclei of a certain element), then the current state of the mixture can be set in terms of partial densities of constituents and temperatures or energy. At this point, thermodynamic and transport properties are readily extracted.

4.2 Shocking: relaxation past 1D shockwave

The *Shocking* code, developed at von Karman Institute for Fluid Dynamics, is a 1D steady-state solver for Euler equations, aimed at computing the thermochemical relaxation past a shockwave.

The code implements the set of equations shown in section 3.3, the reference thermodynamic library being *Mutation++* and the numerical method for integrating the ODEs system is provided by the package *Odeint* from the *Boost* C++ libraries (v1.53 or higher).

Shocking takes as input the pressure, velocity and temperature of the free-stream mixture. To these values, Rankine-Hugoniot jump relations are applied to find the post-shock state (applying a Newton-Rhapson method if internal degrees of freedom are supposed in equilibrium). The chemical composition is supposed frozen during the shock, the free-stream composition is thus the initial composition for the integration process. Constitutive equations are then integrated starting from this composition and from the temperatures obtained by the Rankine-Hugoniot relations, up to the specified distance from the shock.

The code is implemented in FORTRAN 95, but a newer version is currently being developed in C++, following an object-oriented philosophy. It must be noted that the Lagrangian solver implemented in this thesis has a structure very similar to the C++ version of *Shocking*: both *Shocking* and the implemented solver deal with an ODE system along a streamline, sharing the numerical *Odeint* integrators and the *Mutation++* thermodynamic library. Also the integration step is quite similar, solving slightly different equations but interrogating *Mutation++* more or less in the same fashion. For this reason, *Shocking* was the natural choice for the verification step of the Lagrangian solver, shown in section 5.2. The *Shocking* code is also used in section 6.1 to construct input flows to be subsequently refined by the Lagrangian solver to provide a (hopefully) better solution.

Two versions of the *Shocking* solver were used, implementing the "Thermodynamic equilibrium" and the "Two-temperatures" models.

4.3 SPARTA: Direct Simulation Monte Carlo

SPARTA, Stochastic PArallel Rarefied-gas Time-accurate Analyzer, is a Direct Simulation Monte Carlo software developed at Sandia National Laboratories,[1] written in C++. This code was used to provide reference flow fields, on which the temperature or chemistry (or both) can be recomputed (and possibly refined) by the implemented Lagrangian solver. In section 6.2, a DSMC solution is obtained for hypersonic argon flows at different Knudsen numbers, while in section 6.3, *SPARTA* is used to compute the flow field around a meteoroid entering the atmosphere and the Lagrangian solver is then applied to refine the electrons chemistry.

In the following, the DSMC algorithm is quickly overviewed and a couple of testcases are performed with *SPARTA*, showing good agreement with literature results.

4.3.1 The DSMC method

The DSMC method is a statistical method aimed at solving the Boltzmann equation by simulating particles *directly*, see Bird [4]. Basically, the method takes a set of simulator-particles and alternates a *displacement step* in which the particles ballistically move in the domain to a *collision step*, in which some neighboring particles are chosen to collide.

For a number of simulator-particles approaching the actual number of physical particles in the domain and the timestep being small enough, the solution of the standard DSMC method[2] was shown to approach the exact solution of the Boltzmann equation, see Wagner [13]. Of course, the number of simulated particles is usually extremely small with respect to the actual number of physical particles, the ratio usually ranging around $10^{13} - 10^{17}$ and highly depending on the type of considered flow, but this is enough to reach good results for rarefied flows, while for flows in the continuum regime the method quickly becomes too computationally demanding.

The computational domain is discretized in "cells" and collisions are performed only among particles belonging to the same cell, the number of collision being obtained from the kinetic theory. *SPARTA* implements elastic collisions

[1] See `http://sparta.sandia.gov`.

[2] There are many variations of the DSMC method, mainly aimed at reducing computation times and required resources.

with the VSS model, the VHS being just a subcase. Inelastic collisions can be implemented with a continuous Larsen-Borgnakke method with variable relaxation times or with a quantized method (for diatomics molecules only). Chemical reactions are also implemented, the user having the choice among the TCE and the Q-K method. In this work are used the Larsen-Borgnakke continuous method for inelastic collisions and the TCE method for chemical reactions.

Ionizing reactions can be treated by *SPARTA* using the ambipolar assumption. It should be noted however that ambipolar recombinations are not currently implemented. In section 6.3 the developed Lagrangian solver is applied to the computation of electrons recombination in the wake of a meteoroid, the baseline simulation being performed with *SPARTA*. See section 2.3 for an overview of the TCE and VSS models, and for the ambipolar assumption.

There are some checks to be done to test whether the numerical solution obtained with DSMC is physically acceptable:

1. The number of particles in the domain (or the energies) must have reached a steady state. From this point on, cumulative averages can be run to obtain a final solution

2. Each cell should contain at least 10 particles, 20 being a good compromise between accuracy and computational overhead

3. The dimension of cells should be *smaller* than the mean free path everywhere in the domain

4. The time step should be smaller than the mean time between collisions

The grid generated by *SPARTA* is cartesian and cells can be refined at user's wish. An "on-the-fly" grid refinement is also provided by *SPARTA*, adjusting the dimension of cells during runtime following criteria chosen by the user such as local values for some parameter.

4.3.2 Testcases

In this section, two testcases are run with *SPARTA* and results are compared with literature results obtained using different DSMC softwares, namely

MONACO[3] and *dsmcFoam*.[4] First of all, an argon flow at Mach number 10 over a cylinder is shown and a comparison is made with Lofthouse [18], then an air flow at Mach 24.85 over a cylinder is compared to results from Scanlon et al.[19]. Not much emphasis is put in showing those simulations, the goal being mainly testing *SPARTA* capabilities as a preparatory step for the further simulations performed in sections 6.2 and 6.3. The criteria adopted in accepting (or rejecting) a solution are those explained above, namely the convergence in the number of particles, the cell-based Knudsen number, the number of simulated particles per cell and the mean time between collisions. Also a grid-independence study was performed, repeating the simulations with increased number of both cells and simulated particles.

It should be noted that *SPARTA*, *MONACO* and *dsmcFoam* each implement slightly different models and algorithms, so that a small difference among the results has to be accepted. One example is the chemical model: *MONACO* uses the TCE model, while *dsmcFoam* is based on the Q-K approach (see Bird [21]). Also, vibrations are treated in *MONACO* slightly differently than in *SPARTA*. The reader should refer to the manuals of the softwares for more information.

Argon flow

A simulation was performed for an argon flow over a cylinder and compared with results obtained with *MONACO* by Lofthouse [18]. The flow free stream conditions are resumed in the following table:

M_∞	10
Kn_∞	0.05
n_∞	8.494×10^{19} m^{-3}
U_∞	2624 m/s
T_∞	200 K
T_{wall}	500 K
d_{cyl}	0.304 m

Table 4.1: *Free stream conditions for the argon testcase*

The computational domain for this simulation is a rectangle, whose dimen-

[3] *MONACO* is currently developed at University of Michigan by the group of prof. Boyd.

[4] Freely available together with the *OpenFOAM* suite of solvers: `http://www.openfoam.com/`.

sions are shown in Table 4.3 along with the total number of simulated particles and the number of cells, the cylinder being centered in the origin.

The VHS parameters for argon are shown in the following table:

ω_{Ar}	$T_{\mathrm{ref}}[K]$	$d_{\mathrm{ref}}3.595E10[m]$	α
0.734	1000	3.595E10	1

Table 4.2: *VHS parameters for Argon*

x_{min}	-0.457 m
x_{max}	2.5 m
y_{min}	0 m
y_{max}	1.2 m
N_{part}	2.6×10^6 particles
N_{cells}	550×10^3 cells

Table 4.3: *Computational domain for the argon testcase and number of cells and particles*

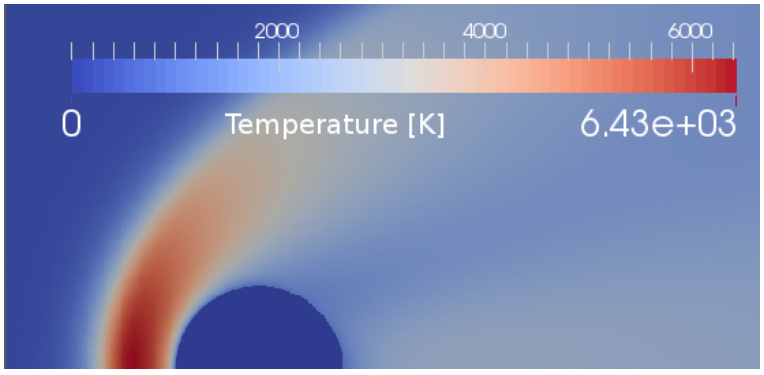

Figure 4.1: *Temperature field for argon flow at M=10*

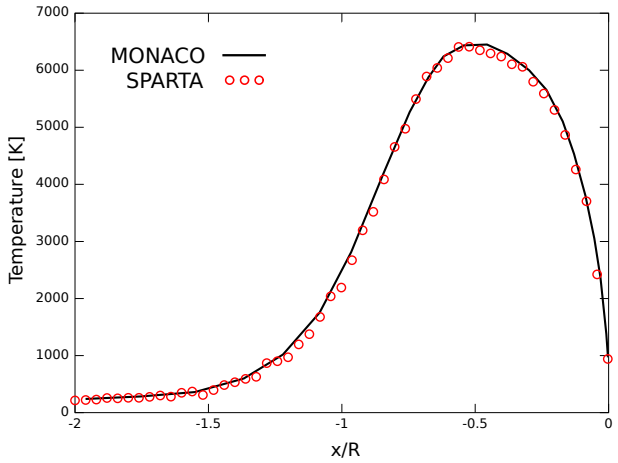

Figure 4.2: *Temperature profile along stagnation line. The surface is located at* $x/R = 0$

Figure 4.1 shows the temperature field in a subset of the computational domain and Figure 4.2 shows the temperature profile along the stagnation line. Values for the temperature are found to be in good agreement with those from Lofthouse [18]. An even better agreement and more smooth results are expected by increasing the averaging time.

Air flow

The last testcase is a cross-flow of chemically reacting air over a cylinder. The air mixture is made of the 5 species: N_2, O_2, NO, N, O. The solution is compared to results from Scanlon et al.[19], obtained with the softwares *MONACO* and *dsmcFoam*.

The following table shows free stream conditions for the flow:

M_∞	24.85
Kn_∞	0.018
n_∞^{air}	1.4331×10^{20} m^{-3}
U_∞	6813 m/s
T_∞	187 K
T_{wall}	1000 K
d_{cyl}	2 m

Table 4.4: *Free stream conditions for the air testcase*

For this testcase, only the forehead region has been analyzed. The computational domain is a rectangle, whose limits are described in the following table and the cylinder is centered in the origin.

x_{min}	-1.5 m
x_{max}	0 m
y_{min}	0 m
y_{max}	2 m
N_{part}	180×10^6 particles
N_{cells}	2.69×10^6 cells

Table 4.5: Domain, number of cells and particles, air
Computational domain for the air flow and number of cells and particles.

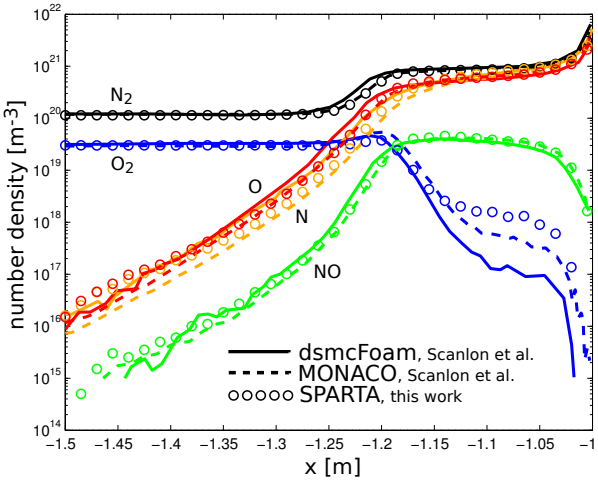

Figure 4.3: *Number densities computed by SPARTA, MONACO and dsmcFoam along the stagnation line. The surface is at* $x = -1$ m

The comparison of *SPARTA* results is fairly good especially with *MONACO*, that like *SPARTA* implements the TCE model. Some deviations near the surface are found but will not be further investigated, the number density of the deviating species being very small with respect to the major species. In Figure 4.4 is shown the translational temperature for molecular nitrogen N_2, the major constituent. The set of chemical reactions used in the simulation is given in appendix B

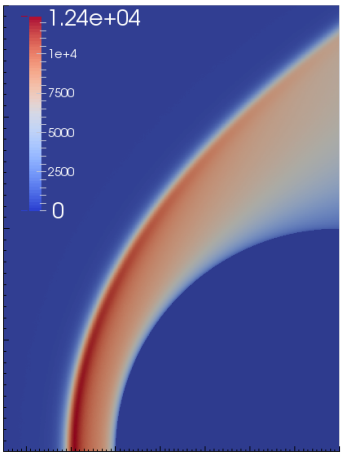

Figure 4.4: *Translational temperature for molecular nitrogen* [K]

CHAPTER 5

Implementation and verification

This chapter shows the implementation of a C++ program developed in this work, able to solve the Lagrangian governing equations introduced in section 3.2. The implemented solver is named *LARSEN*, standing for *LAgrangian Reactor for StrEams in Nonequilibrium* and the logo is shown in Figure 5.1.

5.1 The LARSEN solver

As seen in section 1.2, the goal is creating a solver able to integrate equations along a streamline. Picking initial conditions from the beginning of the streamline, the chemistry and optionally internal temperatures are recomputed.

First of all a reference solution is needed and a streamline is to be extracted from it. This streamline is given as an input to *LARSEN*, that imports the velocity and density fields and also computes the enthalpy along the streamline. *LARSEN* then starts the integration of the ODE set of equations of section 3.2, taking as initial conditions the temperature(s) and mass fractions for chemical species *at the beginning* of the streamline.

Results are returned by *LARSEN* in terms of species mass fractions and temperatures and the user can compare them to initial values using standard elaboration and plotting utilities such as *Octave*, *Gnuplot* etc.

To sum up, the user has to perform two operations:

1. Obtain a reference flow field, that might come from CFD, DSMC, or even analytical methods

2. Extract values over a streamline, format them and start *LARSEN*

Then *LARSEN* performs the following operations:

1. reading the input file specifying the mixture and solver parameters
2. reading the file storing values along the streamline
3. saving values for density and velocity along the streamline
4. computing the total enthalpy along the streamline
5. picking starting values for mass fractions and temperatures
6. integrating up to the end of the streamline

Figure 5.1: *LARSEN logo*

The *LARSEN* program was written in an object-oriented fashion, following the same structure of the *Shocking* code (section 4.2). The two programs have been merged into one only software, the long-term goal being developing a versatile multi-purpose software able to treat different situations of interest in hypersonic flows. It's important to remark that *LARSEN* may be seen as a variant of *Shocking*, that in spite of solving the momentum equation, imports the velocity and density fields as a *datum*,[1] thus gaining the ability of analyzing more general problems than the 1D shockwave case treated by *Shocking*, at the price of fixing the velocity and density fields. Also, just like *Shocking*, the computation of thermodynamic properties and chemical and transfer rates is obtained through calls to the *Mutation++* library.

[1] As well as the enthalpy, that gives the ability of estimating energy fluxes.

Input file

The input file for Larsen has the aim of specifying the mixture type (such as argon, air5, air11 or any mixture implemented in *Mutation++*), the state model (thermal equilibrium flow or two-temperatures model) and the thermodynamic database to be used (Rigid Rotor - Harmonic Oscillator, NASA 7 polynomial or NASA 9).

A sample input file is provided here:

```
# ==========   Basic input file for LARSEN   ============

Problem Type:
larsen

Name of the mixture:
air5

State Model:
ChemNonEq1T

Thermodynamic Database:
RRHO
```

Reference solution

The reference solution along the streamline is given as a plain text file where values are separated by *spaces*. Each line stores the values at one point of the streamline, in the following ordering:

1. curvilinear abscissa s along the streamline, in meters. Can have whichever value at the beginning of the streamline (also negative) but must be *strictly increasing*.
2. translational temperature T of the whole mixture, [K]
3. (only if two-temperatures model is selected) internal temperature T_v, [K]
4. pressure P of the mixture (dummy value, kept here only for future developments)
5. density ρ of the mixture, [kg/m^3]

6. module of velocity U of the mixture, [m/s]

7. mass fractions Y_i of all the chemical species, [*adim.*]

It's important to note that from the values along the streamline, the position must be converted from (x, y, z) coordinates to the curvilinear abscissa s along the streamline and the velocity $\boldsymbol{u} = (u, v, w)$ is to be converted to its module U.

Numerical details

Once the reference solution have been imported, the solver starts integrating from one point on the streamline to the next one.

A description of the steps performed by the solver during one integration step is given in Figure 5.2.

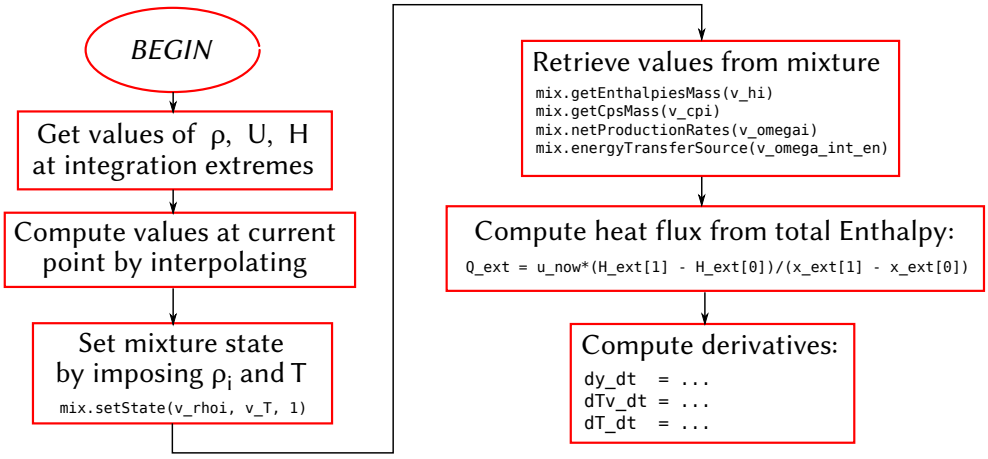

Figure 5.2: *Calculation of the system right hand side.*

The values for the density and velocity at the current position are obtained by linearly interpolating from the values at the extremes. By denoting with the subscripts 1 and 2 the first and second points among which the integration is being done, and by x the current position along the streamline:

$$\begin{cases} \rho = \rho_1 + \frac{\rho_2 - \rho_1}{x_2 - x_1}(x - x_1) \\ U = U_1 + \frac{U_2 - U_1}{x_2 - x_1}(x - x_1) \end{cases} \tag{5.1}$$

The derivative of the kinetic energy per unit mass $\frac{\mathrm{D}U^2/2}{\mathrm{D}t}$ is also needed and is supposed constant during the integration step and computed via finite difference using the extreming values:

$$\frac{\mathrm{D}U^2/2}{\mathrm{D}t} \;=\; \boldsymbol{u}\cdot\boldsymbol{\nabla}U^2/2 \;\approx\; \frac{U}{2}\,\frac{U_2^2-U_1^2}{x_2-x_1} \tag{5.2}$$

where U is the velocity module. This implementation might not seem obvious at a first glance, however this term can be rewritten as:

$$\boldsymbol{u}\cdot\boldsymbol{\nabla}U^2 = U\hat{\boldsymbol{n}}\cdot\boldsymbol{\nabla}U^2 \tag{5.3}$$

where $\hat{\boldsymbol{n}}$ is the direction of the streamline. From this, it's evident that the gradient of U^2 is to be evaluated at two different values along the curvilinear abscissa and to be multiplied by the velocity module U.

Energy fluxes

As explained in section 3.2, energy fluxes changing the enthalpy of the fluid particle are obtained from the reference simulation and plugged directly into *LARSEN*. This can easily be done by computing the total enthalpy H of the reference simulation along the streamline and by noting that this value can change only if diffusion effects are present. Those effects are collectively evaluated as the material derivative of H along the streamline, approximated as constant for each pair of integration points 1 and 2:

$$\mathcal{Q} = \frac{H_2-H_1}{x_2-x_1} \tag{5.4}$$

This approach is advantageous because it does not require any modeling of the dissipative effects such as the heat flux and the shear stresses, that are directly provided through the enthalpy field.

Numerical integrator

The integration is done with the package *Odeint*, part of the well-known C++ *Boost* libraries. *Odeint* provides several different integration schemes, called *steppers*, such as the Runge-Kutta method with various kinds of error estimators. For this work, a *stiff* solver is needed since the chemical rates typically have extremely different values from one reaction to another and the chosen algorithm for this work is the "*rosenbrock4*" method.

5.2 Verification

In this section, some verifications on *LARSEN* are performed on the testcase of thermochemical relaxation past a shockwave, with the main goal of testing the software implementation.

As seen, even if the aims are different, the *LARSEN* and *Shocking* codes share the same structure, extract values from the same thermodynamic library in the same way (Figure 5.2) and integrate their respective system of ODEs in a similar fashion. For this reason, the implementation of *LARSEN* will be here verified against *Shocking* itself.

Once a *Shocking* solution is computed, the velocity and density fields as a function of the position past the shock are assembled and provided to *LARSEN*, altogether with initial conditions of temperature and mass fractions of chemical species. *LARSEN* re-computes the temperature(s) and species mass fractions and those results are compared to the previously found Shocking result. Since the nonequilibrium model is the same for *Shocking* and *LARSEN*, the solution is (as one might expect) found to be in complete agreement with the reference flow.

In the following table are resumed the testcases shown in this section. By "air5" we refer to a mixture of non-ionized air species: N_2, O_2, NO, N, O, while "air11" also includes the ionized air species and free electrons, namely: N^+, O^+, NO^+, N_2^+, O_2^+, e.

Thermal equilibrium - T	Thermal nonequilibrium - T, Tv
Shocking air5 vs Larsen air5	Shocking air5 vs Larsen air5
Shocking air11 vs Larsen air11	-

Unless otherwise specified, the reaction mechanism for air is that of *Park, 2001* [11]. The free-stream conditions are the same for the three testcases and are shown in the following table, the difference among the testcases being the number of considered chemical species and the number of internal temperatures. The chosen free-stream conditions are arbitrary and were inspired from the atmospheric entry of the Fire-II capsule (see for example [12]).

P	5.2 Pa
T	210 K
U	11310 m/s

As a result of this verification step, we can state that *LARSEN* meets the expectations of reproducing *Shocking* (correct) results, up to numerical precision.

Shocking air5 vs Larsen air5 - 1T

The results shown here refer to a mixture of air5 in thermal equilibrium and chemical nonequilibrium. Figure 5.3 shows the temperature plotted against the distance from the shock (located at $x = 0$), and Figure 5.4 shows the mass fractions for the 5 simulated chemical species.

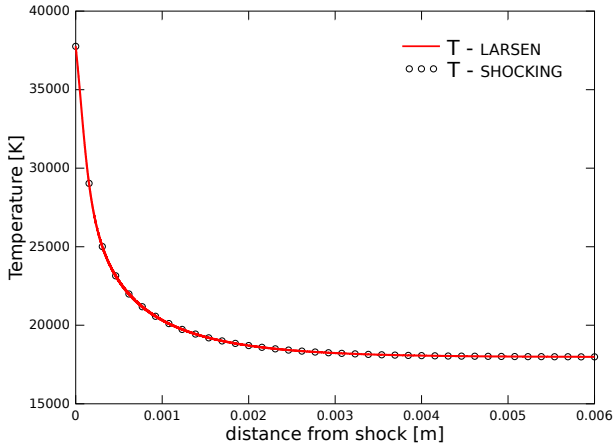

Figure 5.3: *Temperature relaxation past the shock.*

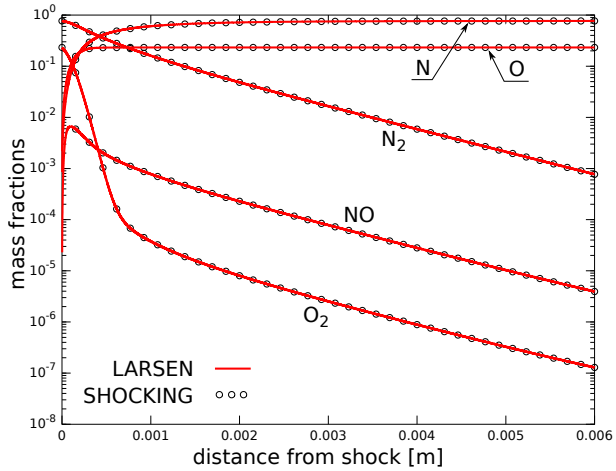

Figure 5.4: *Mass fractions past the shock.*

Shocking air11 vs Larsen air11 - 1T

This testcase introduces more chemical species with respect to the previous one, now allowing the ionization of atoms and molecules. Again, the computation is performed in the hypothesis of thermal equilibrium.

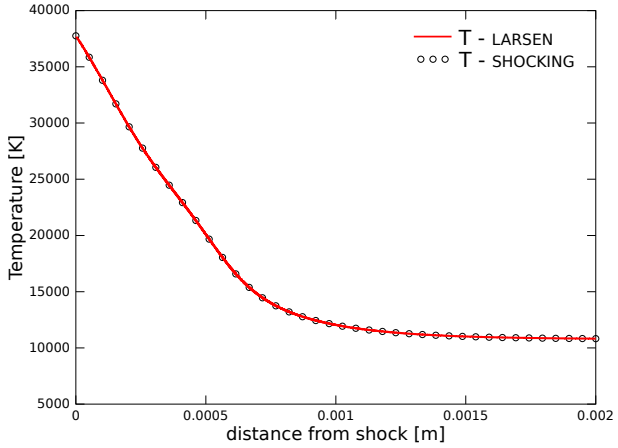

Figure 5.5: *Temperature relaxation past the shock.*

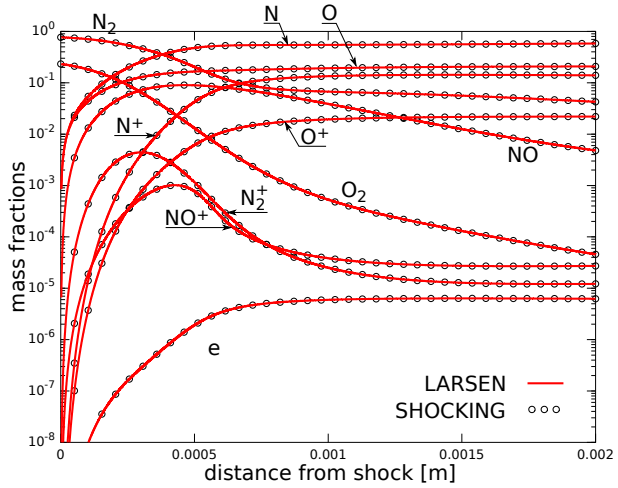

Figure 5.6: *Mass fractions past the shock.*

Shocking air5 vs Larsen air5 - T, Tv

This last testcase is performed on only 5 chemical species, but the hypothesis of thermal equilibrium is removed in favor of a two-temperatures model. The translational temperature jumps across the shock, while the vibrational one is supposed frozen in the pre-shock equilibrium state of 200 K.

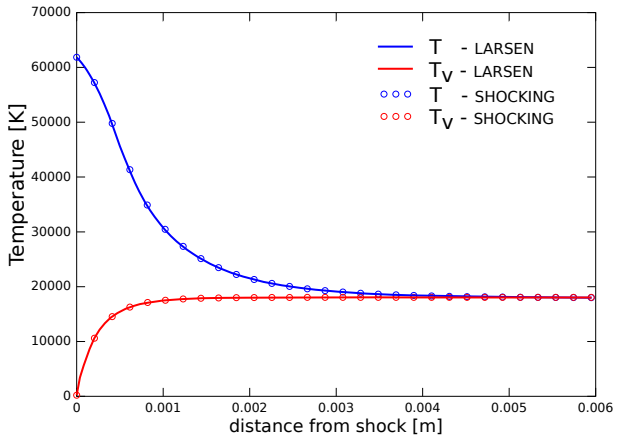

Figure 5.7: *Temperature relaxation past the shock.*

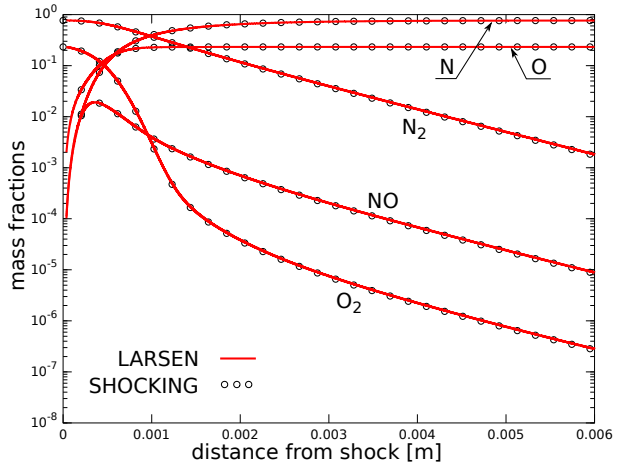

Figure 5.8: *Mass fractions past the shock.*

CHAPTER 6

Results

The implementation for the Lagrangian solver *LARSEN* was shown and the solver was verified on some testcases using the software *Shocking* as a reference solution. In this chapter, *LARSEN* is applied to some flows to improve the chemistry or recompute the temperatures by enabling thermal nonequilibrium. Section 6.1 deals with the thermochemical relaxation past a 1-dimensional shock, while in section 6.2 *LARSEN* is applied to a rarefied 2D axisymmetrical flowfield computed with the Direct Simulation Monte Carlo software *SPARTA*. Finally, in section 6.3 *LARSEN* is applied to the problem of computing the recombination of electrons in the trail of a meteoroid entering the atmosphere at high velocities and altitudes.

6.1 1D shockwave

If we were to compare two computations of strong shockwaves, obtained with different chemical models, we would find results that are very different in terms of velocity and density fields. In Figures 6.1 and 6.2 for example are compared the velocity and density fields obtained in case the chemistry model is composed of 5 species (*air5*) or 11 species (*air11*), for a shockwave moving at 11310 m/s in still air at rest pressure $P = 5.2$ Pa and $T = 210$ K. The basic hypothesis underlying this thesis work is that a good solution in terms of chemistry may be obtained even with roughly estimated (but consistent) velocity and density fields. This might seem unrealistic in the light of the shown pictures, however the velocity and density fields show a somehow complementary behavior and the hypothesis might be supported by the results. Verifying this hypothesis is the prerequisite for the Lagrangian solver developed in this work to be of some use, and is thus the aim of this paragraph.

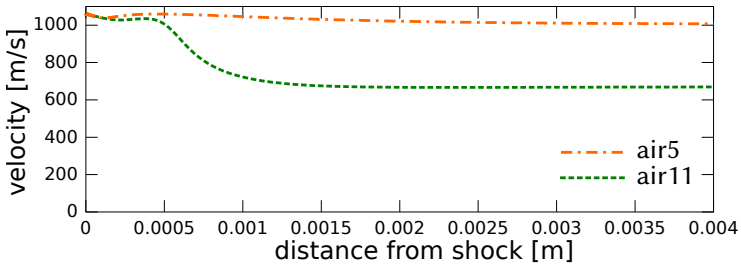

Figure 6.1: Velocity computed for a shockwave at free stream conditions: $u = 11310$ m/s, $P = 5.2$ Pa, $T = 210$ K

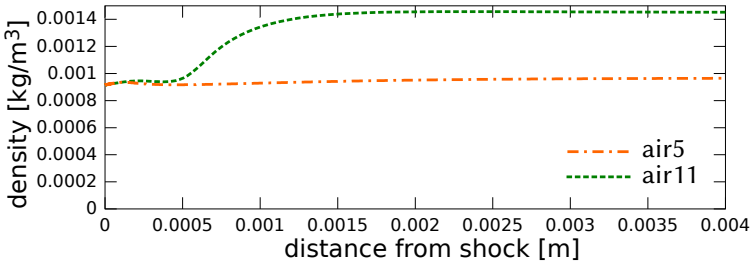

Figure 6.2: Density computed for a shockwave at free stream conditions: $u = 11310$ m/s, $P = 5.2$ Pa, $T = 210$ K

As in the previous section, the testcase is the relaxation past a shockwave and the reference software for both input files and reference solutions is *Shocking*. The route for running the testcases of this section is as follow:

1. Compute a simple flowfield with *Shocking* (few chemical species or one only temperature for example)

2. Refine the simple flowfield with *LARSEN* by passing it the density, velocity and initial conditions previously computed

3. Compute a complete solution with *Shocking* and compare it with the *LARSEN*-refined solution

The following testcases are here discussed:

- Improving chemistry:
  - from an N_2-O_2 mixture to air5, both in thermal equilibrium
  - from air5 to air11, both in thermal equilibrium
- Introducing thermal nonequilibrium:
  - from air5 in thermal equilibrium to air5 with two temperatures model

N_2-O_2 mixture to air5, 1T

In this testcase the post-shock conditions for a non-reacting mixture of N_2 and O_2 (also referred to as "air2") are obtained using *Shocking*. A file storing the reference solution is compiled and *LARSEN* is run on it, introducing the chemical species of air5. A new complete simulation of air5, 1T is also performed with *Shocking* and results are compared to the *LARSEN*-refined ones. All the simulations are performed in the hypothesis of thermal equilibrium. The following table resumes free stream conditions (pre-shock state):

P_∞	5.2 Pa
T_∞	210 K
U_∞	7000 m/s

In Figure 6.3 is shown the temperature profile refined by *LARSEN* (red) and the correct one computed by *Shocking* (black), while Figure 6.4 shows the mass fractions of chemical species. We can see that the introduction of the chemistry in the N_2-O_2 mixture have a strong effect on temperature, that decreases due to endothermic dissociation reactions, tending to a plateau which correspond to chemical equilibrium. Molecular oxygen quickly reaches a fully dissociated state, whereas molecular nitrogen does only partially. The formation of NO due to Zel'dovich reactions peaks at a distance of 0.8 mm from the shock.

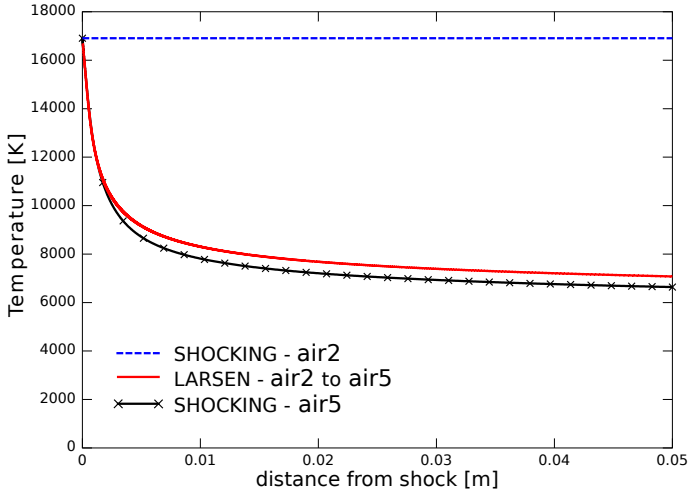

Figure 6.3: *Temperature profile computed by LARSEN and correct solution by Shocking.*

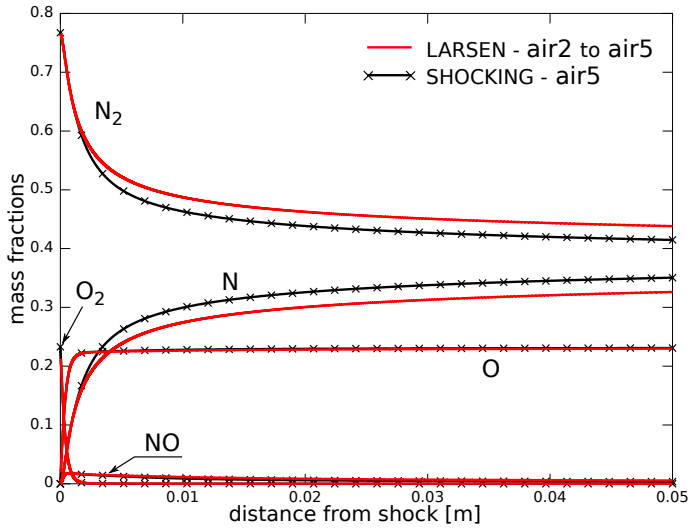

Figure 6.4: *Mass fraction for chemical species computed by LARSEN and correct solution by Shocking.*

In Figure 6.5 the mass fractions are shown in logarithmic scale in the vicinity of the shock.

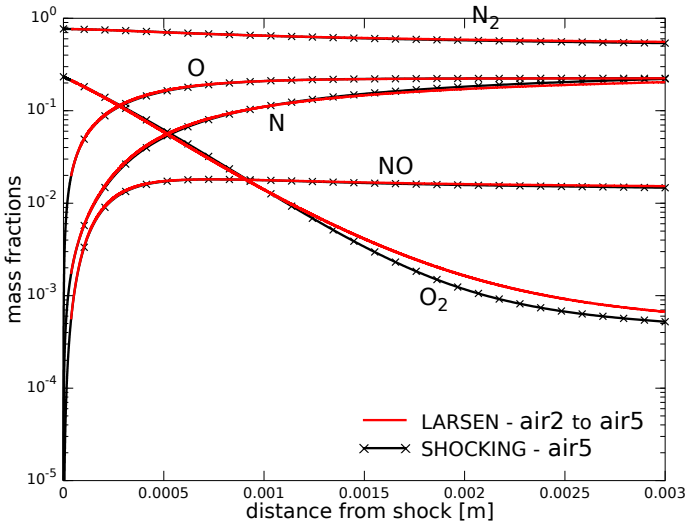

Figure 6.5: *Mass fraction for chemical species computed by LARSEN and correct solution by Shocking. Solution in logarithmic scale, close to the shock*

The refined solution shows big improvements in both the computed temperature and in the chemical species, the values being very similar to the correct results. The discrepancies in the mass fractions and temperature at equilibrium should be interpreted by recalling that both the velocity and densities used by *LARSEN* are those of non-reacting N_2 - O_2 mixture, possibly quite different than the natural results for air5.[1]

Air5 to air11, 1T

This testcase is very similar to the previous one. *LARSEN* refines an air5 solution by introducing ionization reactions. This time the shock is chosen to be stronger, the conditions being chosen to be similar to those encountered in the trajectory of the FireII probe during atmospheric entry around 80km of altitude, see for example [12]. The following table resumes free stream conditions (pre-shock state):

[1] Although the temperature is predicted to be a little higher, the dissociation is predicted by *LARSEN* to be lower. This might seem incorrect, however should be noted that dissociation also depends on densities.

P_∞	5.2 Pa
T_∞	210 K
U_∞	11310 m/s

In Figure 6.6 is shown the temperature profile refined by *LARSEN* (red) and the correct one computed by *Shocking* (black), while Figure 6.7 shows the mass fractions of chemical species.

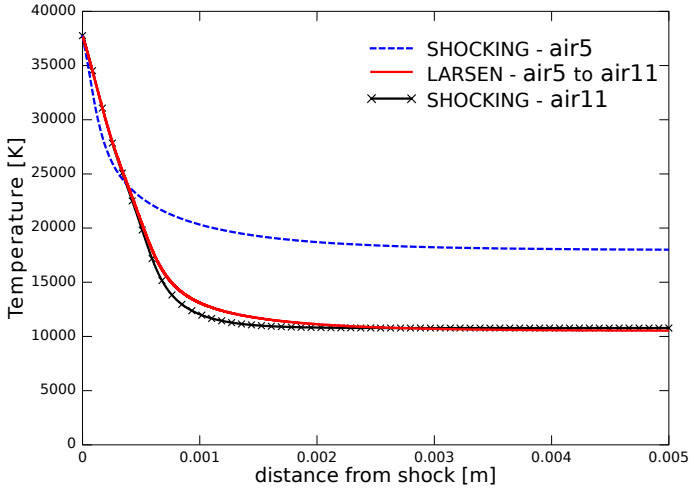

Figure 6.6: *Temperature profile computed by LARSEN and correct solution by Shocking.*

In this case, the ionized species predicted by *LARSEN* are in good agreement with the *Shocking* original result, while the prediction for neutral species shows big differences at about 3 mm after the shock. This difference eventually vanishes and at about 2 cm after the shock the results provided by *LARSEN* are very close to the correct ones computed by *Shocking*. In Figure 6.8, a plot of the species in logarithmic scale at a closer distance to the shock is shown.

From the mass fraction plots we can see that a considerable amount of N is ionized. The first electrons are provided by the associative ionization reactions:

$$\mathrm{N} + \mathrm{O} \rightleftharpoons \mathrm{NO}^+ + \mathrm{e}^-$$
$$\mathrm{N} + \mathrm{N} \rightleftharpoons \mathrm{N}_2^+ + \mathrm{e}^-$$

and as their concentration increases, ionization by electrons impact dominates, in particular for nitrogen atoms.

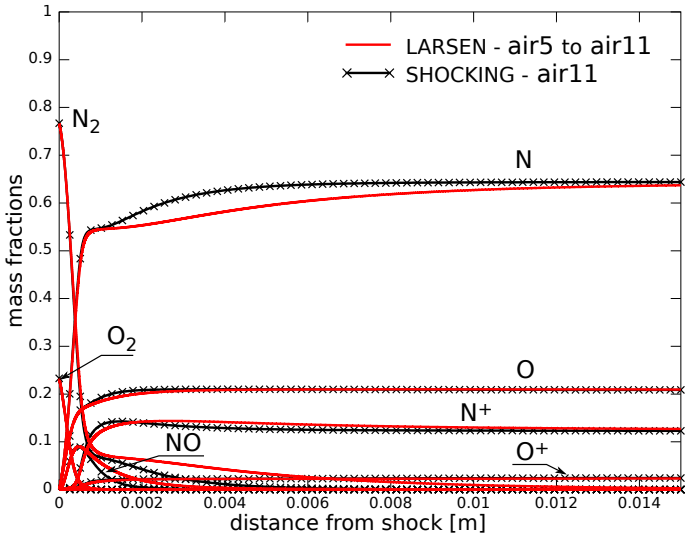

Figure 6.7: *Mass fraction for chemical species*

It should be noted that the ion N^+ is a "newcomer" in the problem, since it is not present in the starting air5 mixture. Since the ionization process requires much energy, it is reasonable to think that in the region where the ionization becomes appreciable, the density and velocity fields might experience possibly strong deviations from the previous non-ionizing air5 simulations. An unexpected deviation of the density and/or velocity field in this region could justify the error experienced by *LARSEN* starting from 2 mm up to 3 cm from the shock.[2]

It is important to recall that this testcase is at higher shock velocity (11310 m/s) than the previous one (7000 m/s).

[2] This hypothesis should be verified by plotting the density and velocity fields for the air5 and air11 simulations.

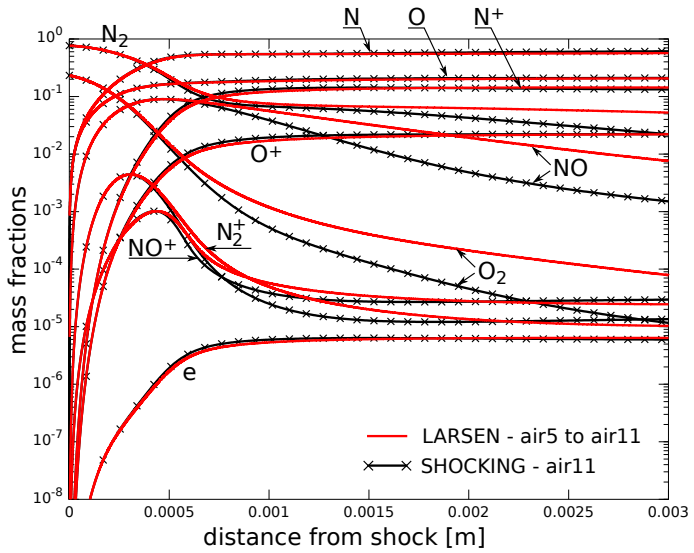

Figure 6.8: *Mass fraction for chemical species near the shock*

Air11 to air5, 1T

The reverse of the previous testcase is here performed for sake of curiosity: an input file computed for an air11 mixture is processed by *LARSEN*, that forces the mixture to be composed only by 5 chemical species. Of course, this case is not of practical interest, since one usually would like to *refine* a solution rather than making it more rough, however this testcase let some symmetry emerge in the behavior of the Lagrangian method. By comparing Figures 6.9, 6.10 and 6.11 with those in the previous testcase, we can see that the trend for the temperature is reversed: the temperature predicted by *LARSEN* now gets below the correct value and then slightly rises above it where equilibrium is reached. Also the error in the mass fractions prediction is inverted and has a similar extent to that from the "air5-to-11" testcase.

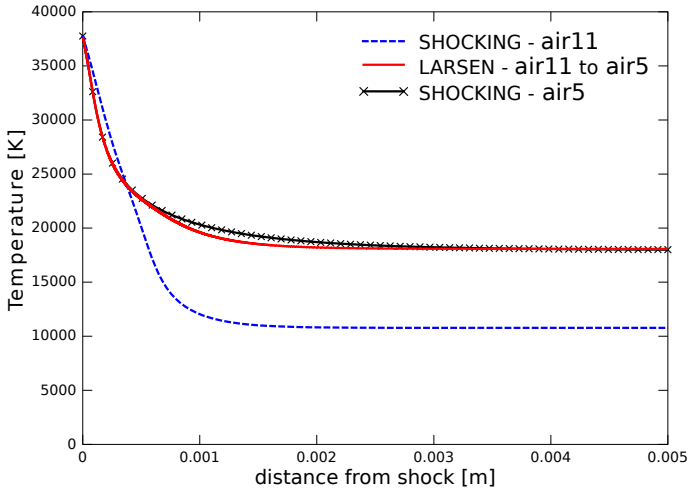

Figure 6.9: *Temperature profile computed by LARSEN and correct solution by Shocking.*

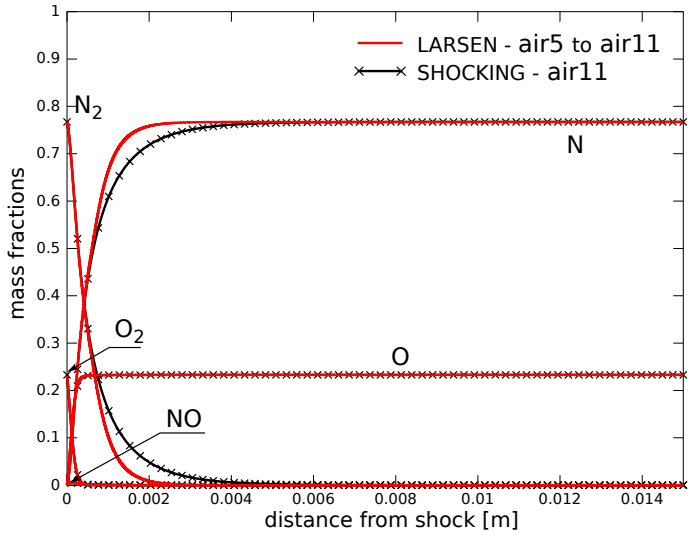

Figure 6.10: *Mass fraction for chemical species very close to the shock, as computed by LARSEN, compared with correct solution by Shocking*

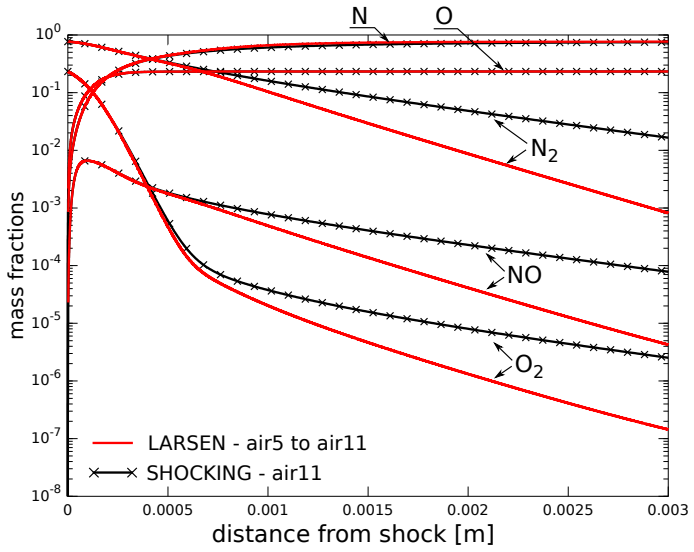

Figure 6.11: *Mass fraction for chemical species, as computed by LARSEN, compared with correct solution by Shocking. Region near the shock - Logarithmic scaling*

Air5 1T to air5 2T

In this testcase, the *LARSEN* solver is used to introduce thermal nonequilibrium into a mixture, preserving the chemical model. The simulation to be refined is an air5 mixture in thermal equilibrium, chemically relaxing after a shockwave whose free-stream conditions are:

P_∞	5.2 Pa
T_∞	210 K
U_∞	7000 m/s

As usual, the velocity and density fields are taken from the starting solution and the initial value for the species mass fraction is that of the free-stream state. The initial condition for the vibrational temperature is the equilibrium free-stream value of 210 K, while the initial translational temperature is found using Rankine-Hugoniot conditions for a gas of constant specific heats.[3]

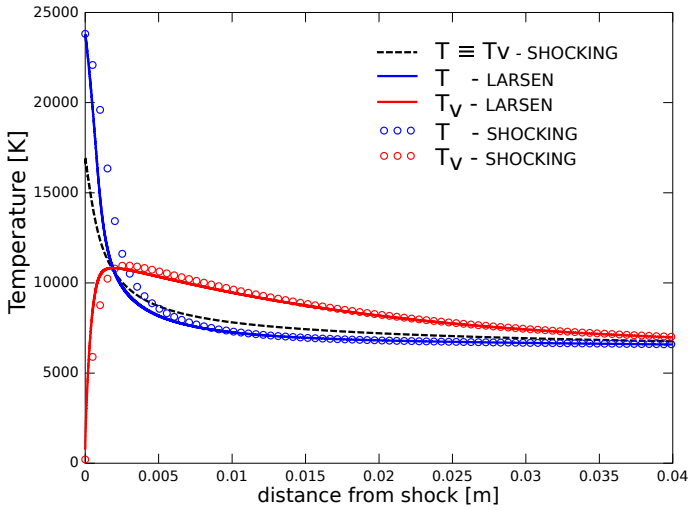

Figure 6.12: *Translational and vibrational temperatures as refined by LARSEN, vs starting and correct solutions*

[3] The physical meaning of this condition is the assumption that translational and rotational degrees of freedom are very quick to be excited, so that they almost jump across the shockwave. Vibrational and electronic degrees of freedom on the other hand are supposed slower and their excitation is negligible across the (thin) shock layer.

We can see that introducing thermal nonequilibrium brings appreciable improvements in the mass fractions and that the predicted temperatures are qualitatively correct. In both the *LARSEN*-refined solution and the correct solution by *Shocking* we can see that the maximum of vibrational temperature is located where the vibrational and translational temperatures are equal, as expected from the adopted model for chemistry-vibrational energy coupling.

It should be noted that *LARSEN* is able to correctly predict the peak value for NO, that is underestimated by 3 times in the thermal equilibrium simulation.[4]

In this testcase, the final values of velocity and density are the correct ones since the chemical model is the same for all the simulations and when the mixture reaches thermal equilibrium all the thermodynamic parameters will be the same. The improving capability of *LARSEN* for this testcase is thus to be investigated just in the non-equilibrium region.

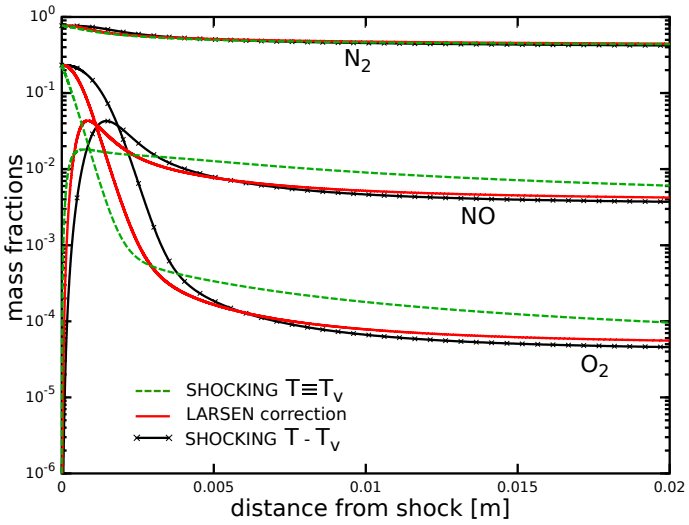

Figure 6.13: *Mass fraction for the species N_2, O_2 and NO.*

[4] This might suggest that *LARSEN* may be of some use to obtain a first rough correction to CFD computations performed in the equilibrium hypothesis. In fact, the cost of CFD simulations grows quickly if many temperatures are to be taken into account, since introducing one equation would heavily increase the dimension of the system to be solved.

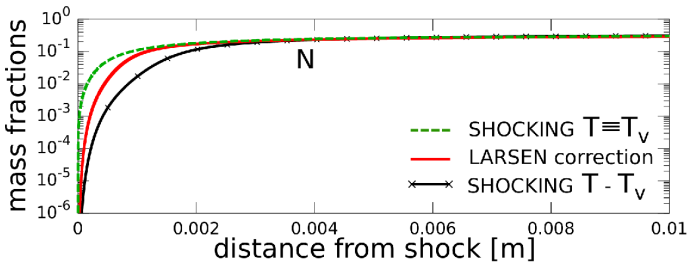

Figure 6.14: *Mass fraction of atomic nitrogen.*

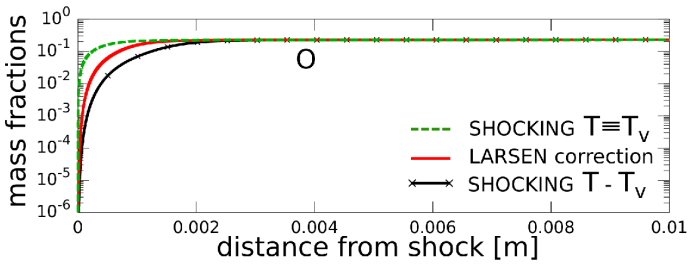

Figure 6.15: *Mass fraction of atomic oxygen.*

Final remarks

The fact that temperature and mass fractions approach quite well the correct results gives credit to the assumption that velocity and density can be decoupled and taken as given from a previous simulation.

In order to accept the difference in the equilibrium values between the *LARSEN* refined result and the correct ones from *Shocking*, one should recall that the velocity and density fields given to *LARSEN* are *different* from the exact values since are based on the simplified initial model.

From the testcases it is evident that except for the equilibrium values - that are predicted fairly accurately - the decoupling of density-and-velocity and the chemistry yields some substantial error in the nonequilibrium region. However, the Lagrangian solver is able for each testcase to significantly refine the initial guess and provide a result which is closer to the correct values.

6.2 2D argon flow from DSMC

This section is aimed at assessing *LARSEN*'s capabilities of elaborating a multi-dimensional DSMC flowfield. An argon flow over a cylinder at three different Knudsen numbers is computed with *SPARTA* and fed into the *LARSEN* solver. First of all *LARSEN* is applied to the stagnation line for all the three Knudsen numbers and the temperature profiles are computed, then for the case of $Kn = 0.05$ the computation is repeated on a whole streamline. A pure argon flow was chosen in order to have the simplest possible DSMC result, where no thermal nonequilibrium is present.[5]

The *LARSEN* computations are performed both in the hypothesis of "adiabatic fluid particle" and by including the energy fluxes as explained in section 5.1. As could be expected, the adiabatic hypothesis shows to be totally inadequate for the testcase considered, not only in the shock region and near the wall, but also in the wake.

DSMC simulations

Three DSMC simulations were performed with *SPARTA* on the forehead region of the cylinder at Knudsen numbers of 0.01, 0.05 and 0.25, based on the cylinder diameter. Those three simulations will be used for running *LARSEN* on the stagnation line. The following tables resume input values for the simulations.

M_∞	10
U_∞	2624 m/s
T_∞	200 K
T_{wall}	500 K
d_{cyl}	0.152 m

	Kn	$n_\infty[part/m^3]$	$\rho_\infty[kg/m^3]$
testcase 1	0.01	4.247×10^{20}	2.817×10^{-5}
testcase 2	0.05	8.494×10^{19}	5.635×10^{-6}
testcase 3	0.25	1.699×10^{19}	1.127×10^{-6}

The initial grids used for the simulations are provided in the following table, where N_x and N_y represent the number of cells in the x and y directions. The software was set to locally refine the grid in order to ensure that nowhere in the domain the local Knudsen number[6] gets below unity. However, it is customary

[5] The only internal degree of freedom for Argon, namely electronic excitation, was neglected in this simulations.

[6] The local Knudsen number is defined as the ratio between the mean free path and the cell dimension: $Kn_{local} = \lambda/\Delta x$

to tolerate a small deviation from this general rule very close to the stagnation point, in case the condition would require a too heavy refinement of the grid.

A schematic view of the domain is provided in Figure 6.16.

	N_x	N_y	$L_x[m]$	$L_y[m]$
testcase 1 - Kn 0.01	544	544	0.381	0.381
testcase 2 - Kn 0.05	222	222	0.457	0.457
testcase 3 - Kn 0.25	666	400	1.0	0.7

Table 6.1: *Grids used for the three testcases*

Note that for the testcase number 3, at Kn = 0.25, the domain is quite large with respect to the other testcases. This dimension was found to be the smallest to properly simulate the (very) smeared shock.

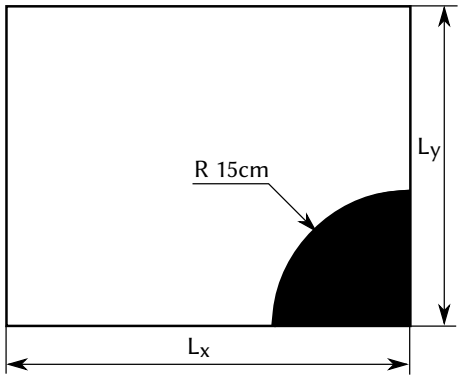

Figure 6.16: *Computational domain for the testcases.*

The parameters for the VHS model are shown in Table 6.2:

ω_{Ar}	T_{ref} $[K]$	d_{ref} $[m]$	α
0.734	1000	3.595E10	1

Table 6.2: *VHS parameters for Argon*

Results were verified to be in agreement with literature values provided by Lofthouse [18]. In Figures 6.17, 6.18 and 6.19 are shown the translational temperature fields for the three testcases. As expected, the shock becomes increasingly spread-out as the Knudsen number increases.

From the simulations we can see two regions where (as expected) the temperature gradients are big: the shock region and the region near the (colder) surface. In those regions the *LARSEN* solver will predict very bad results if the assumption of adiabatic particle is made.

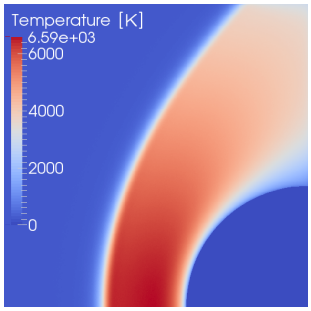

Figure 6.17: *Translational temperature for Kn 0.01*

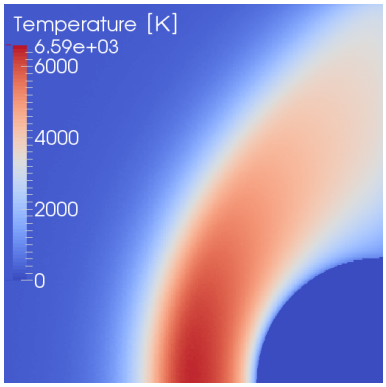

Figure 6.18: *Translational temperature for Kn 0.05*

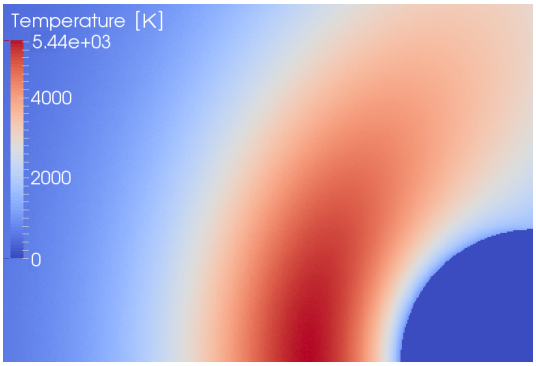

Figure 6.19: *Translational temperature for Kn 0.25*

Table 6.3 resumes the final number of cells after the automatic Knudsen-based refinements, as well as the number of simulated particles in the domain and the chosen timestep for the simulations.

		$N_{\text{simulated}}$	N_{cells}	$\Delta t[s]$
Kn = 0.01	→	2.63 M	330000	2.0E-7
Kn = 0.05	→	1.67 M	51000	1.0E-6
Kn = 0.25	→	225 K	7000	5.2E-6

Table 6.3: *Number of simulated particles, number of cells and timestep for the forehead simulations*

Also, a simulation was performed on an extended domain for the testcase at Kn = 0.05. This simulation will be used for running *LARSEN* on a whole streamline. Flow parameters for the simulation are the same as for the previous smaller argon simulation, while the initial grid is shown in Table 6.4 and the domain is shown in Figure 6.20. Just as in the previous simulations, the initial grid was automatically adapted in order to keep the local Knudsen number above unity everywhere in the domain.

N_x	N_y	$L_x[m]$	$L_y[m]$
1200	530	2.743	1.219

Table 6.4: *Initial grid for the extended argon case*

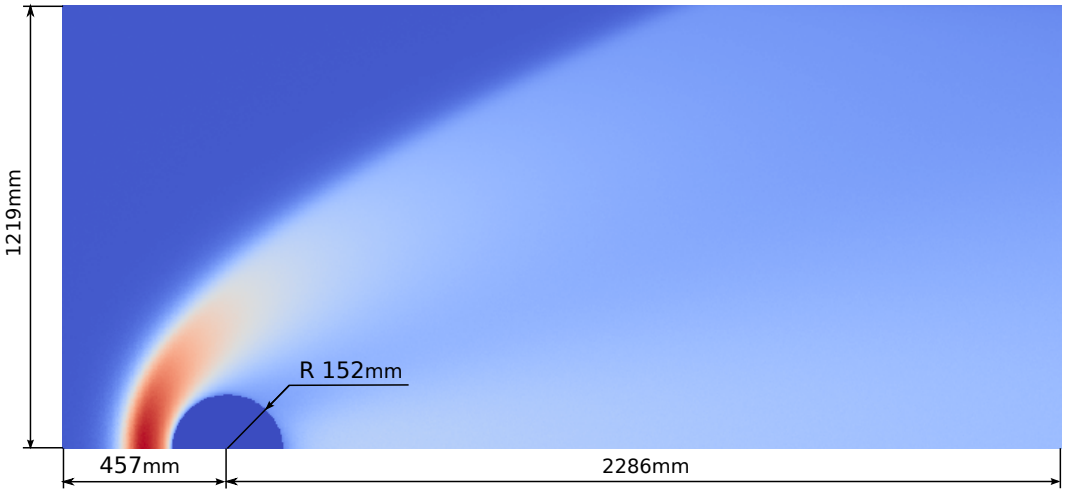

Figure 6.20: *Domain and temperature profile for the extended testcase. $Kn = 0.05$*

Stagnation line

From the three shown simulations at three Knudsen numbers, values along the stagnation line are extracted and passed as a "starting solution" to *LARSEN*. Since the mixture is composed only by argon and electronic excitation is neglected, no internal degrees of freedom are present and the only operation actually done by *LARSEN* is recomputing the temperature profile, starting the integration from the free-stream conditions.

The computation is first of all performed in the hypothesis of "adiabatic fluid particle", where the total enthalpy is conserved. Results for the three Knudsen numbers are shown in Figure 6.21, where the position $x = -0.15$ m is coincident to the stagnation point. This adiabatic hypothesis shows to be inadequate especially near the surface, that has a lower temperature with respect to the post-shock region (namely, $T_{wall} = 500$ K).

It should be noted that for all the Knudsen numbers, the adiabatic solution by *LARSEN* reaches the values given by the Rankine-Hugoniot jump conditions as the particle approaches the wall and thus the velocity reaches zero.

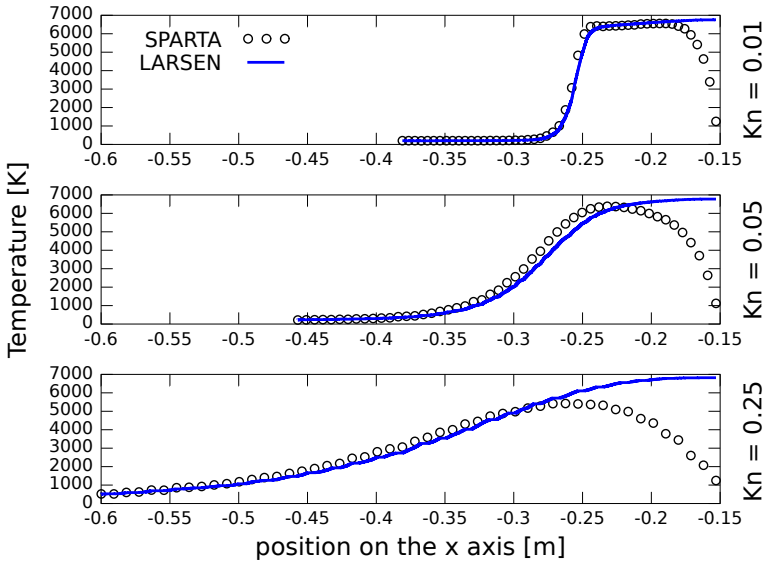

Figure 6.21: *Temperature profile for adiabatic fluid particle along the stagnation line.*

The *LARSEN* computation was then repeated by introducing diffusion ef-

fects, allowing the total enthalpy of the fluid particle to change along the streamline. The effect of the heat flux and all the dissipative effects is estimated from DSMC directly, by computing the variation of the total enthalpy from one point on the streamline to the next one, as explained in section 5.1. Results for this computation are plotted in Figure 6.22 and show to be extremely close to the values computed by DSMC.

Although not formally exact,[7] the hypothesis of importing dissipative effects in *LARSEN* directly from the DSMC solution proves to be extremely valuable for this testcase.

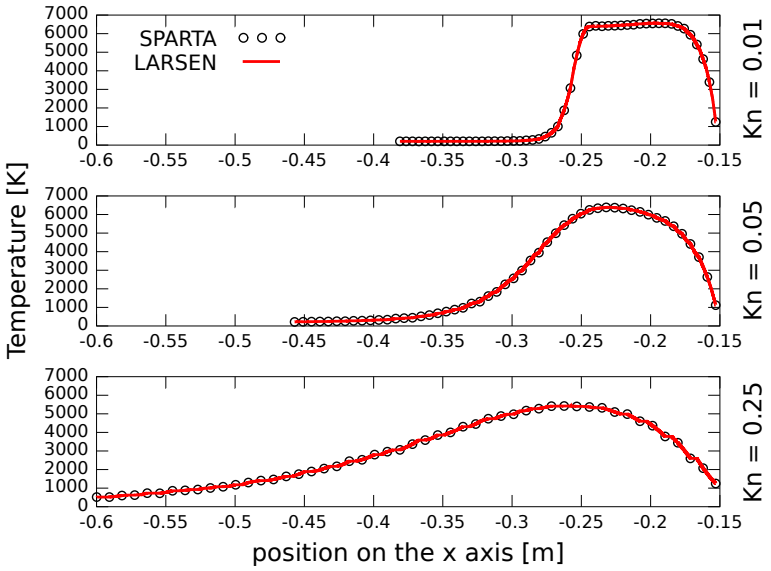

Figure 6.22: *Temperature profile along the stagnation line with diffusion effects.*

[7] From a physical point of view, diffusion effects should be modeled through the gradients of the velocity field and of the temperature, which is an unknown of the problem itself. The inclusion of dissipative effects is here achieved from the enthalpy based on the temperature field provided by the DSMC simulation. This means that the included diffusion effects are completely decoupled from the *LARSEN* solution.

Streamline

After testing *LARSEN* on the stagnation line, a generic streamline is here analyzed. The computation is performed in the hypothesis of conserved enthalpy and then repeated introducing diffusion effects.

In Figure 6.23 are shown some streamlines extracted from the flowfield using the Open Source software *ParaView*, integrating the velocity field with a Runge-Kutta 4-5 method. The streamlines here analyzed are the two in solid line style.

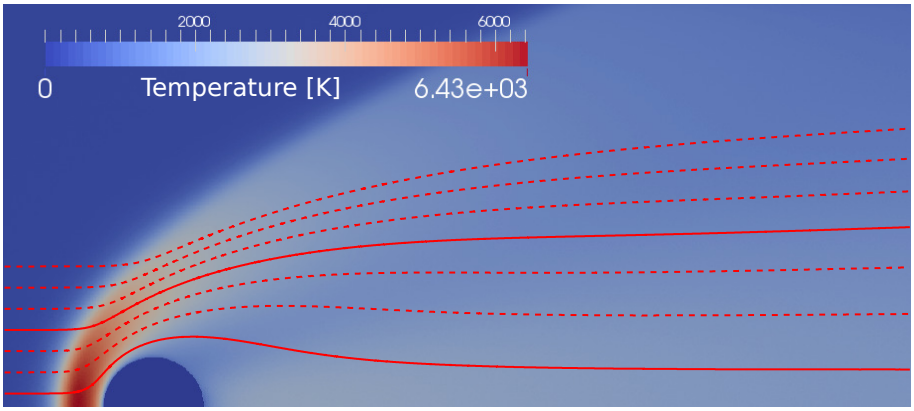

Figure 6.23: *Streamlines for the argon flow at* Kn = 0.05*, superimposed to the temperature field. Continuous lines are processed by LARSEN.*

From the results in Figure 6.24, it can be stated that the hypothesis of adiabatic fluid particle heavily fails for streamlines near the body, giving reasonable results only for farther streamlines, where gradients are much less steep. On the other hand, taking into account the variation of total enthalpy due to diffusive processes gives the expected result.

The success of *LARSEN* with rarefied and multidimensional DSMC simulations, altogether with the capabilities of improving the chemistry or introduce nonequilibrium in a given flow (section 6.1), give hope that the solver may improve the results in the mixed case of multidimensional reacting flows, possibly out of equilibrium.

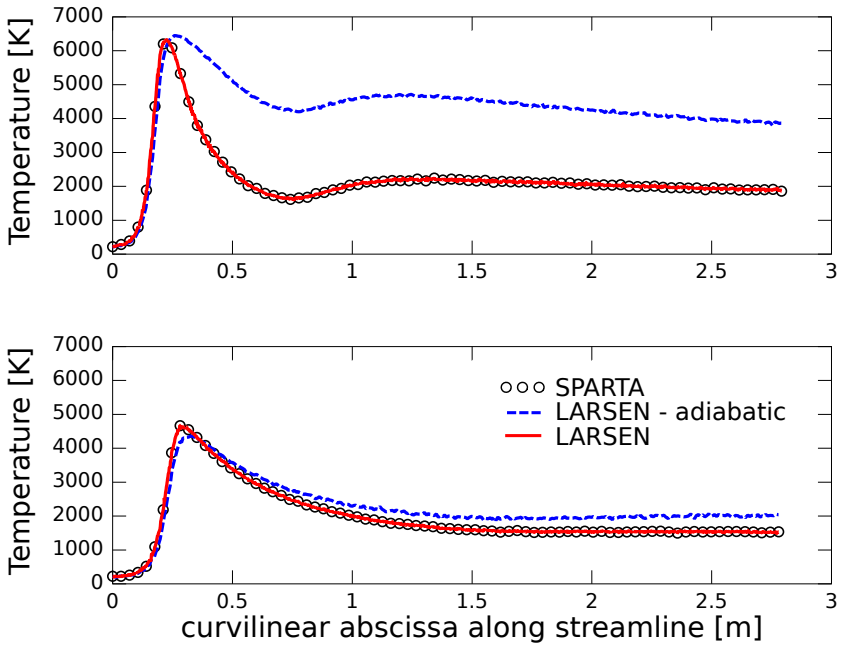

Figure 6.24: *Results computed by LARSEN with and without diffusion, vs SPARTA. Top: streamline closer to the body; Bottom: streamline farther*

6.3 Electrons recombination in a meteoroid trail

In this section, *LARSEN* is applied to the analysis of the ionized flowfield around a meteoroid entering the Earth's atmosphere at high altitude and velocity.

As seen in the introductory chapter (section 1.3), the detection of incoming meteoroids via radar signals requires an estimation for the shape of the ionized meteoroid trail. Since the current implementation of *SPARTA* (and many DSMC codes as well) does not support electrons recombination with their ions through three-body reactions,[8] *LARSEN* is asked to compute a more accurate chemistry past the meteoroid, the long-term goal being determining the length of the ionized region for radar detection purposes.

The current work focuses on a *non-ablating meteoroid*, where the only source for electrons are ionizing collisions among air species. It's important to

[8] At least not in the framework of the ambipolar assumption

remark that the ablation process might have two important effects: increasing the density of electrons[9] and filling the trail.

The fluid is a mixture of 11 air species, namely N_2, O_2, NO, N, O, N^+, O^+, N_2^+, O_2^+, NO^+ and electrons, entering the domain with a molar fraction of $0.79N_2 - 0.21O_2$,[10] the set of reactions being reported in appendix B, while VSS parameters for air constituents are given in section 2.3. Once again, it should be remarked that electrons are treated with the ambipolar assumption and do not recombine.

The considered meteoroid is modeled as a sphere of 1 cm diameter, traveling at 72 km/s at an altitude of approximately 80 km, the Knudsen number based on the meteoroid diameter being approximately $Kn = 0.1$. The free-stream conditions are shown in the following table:

T_∞	221 K
U_∞	72 km/s
n_∞	1.1889×10^{21}[part/m^3]
T_{wall}	2000 K

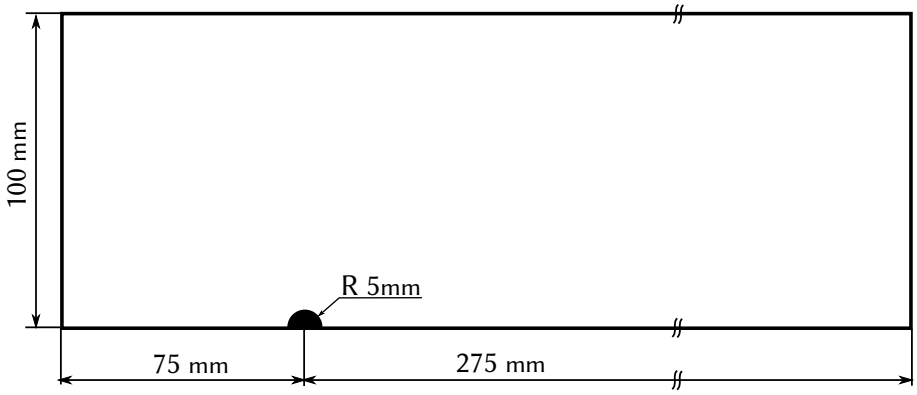

Figure 6.25: *Computational domain for the nonablating meteoroid simulation*

[9] Metals have a ionization energy that is almost half that of air species and are very reactive. It's reasonable to expect that ablated metallic species will drastically increase the density of electrons with respect to non-ablating results.

[10] Should be noted that at high altitudes, the atmosphere composition is altered by solar radiation and the concentration of dissociated species in the free stream might be non-negligible.

Collisions with the surface are treated as fully diffusive. Figure 6.25 shows the computational domain used in the DSMC simulation. A starting cartesian grid of 1540×436 cells was chosen, then refined near the surface to ensure that the local Knudsen number does not get below unity. As a final result, the domain is composed by 527 000 cells, the number of particles after the initial transitory is about 30 millions[11] and the chosen timestep is equal to 10^{-10} s. The simulation was run until the number of simulated particles had reached an almost stationary value for each chemical species, that is after 90 000 timesteps. Only then, the transitory of the simulation is supposed over and cumulative sampling is started to get averages. The simulation is 2D axisimmetric.

Figure 6.26 shows the number density of electrons as computed with the DSMC code. The peak value is found to be approximately 1.76×10^{22} particles per cubic meter and is located at the stagnation point.

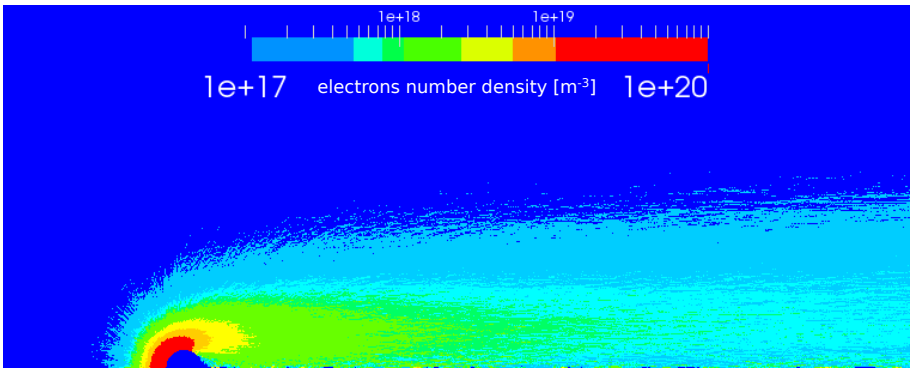

Figure 6.26: *Number density of electrons for nonablating meteoroid*

The electrons number density can be used to compute the plasma frequency and is fundamental to assess the conductivity and reflectivity properties of the plasma. As reviewed by *Pellinen-Wannberg et al.* in [16], electrons number densities of 10^{14} m^{-3} are already detectable with radars working in the range of VHF, while number densities of 10^{16} m^{-3} can be tracked by UHF. From the DSMC simulation we can thus see that air alone is able to produce enough

[11] The number of particles is chosen so that the number of particles in the free-stream cells is around 20 for each cell. Some regions in the trail very close to the body end up having few particles, around 8 - 10: this is a well-known difficulty in rarefied hypersonic flows, where the region past the body is almost empty. The simulation of ablating flows solves this issue and the trail results composed mainly by ablated species.

plasma for the meteor trail to be detected by currently employed radars. In Figure 6.27 can be seen a contour plot that shows more clearly the number densities in the range $10^{16} - 10^{18}$ part/m^3 as computed by *SPARTA*.

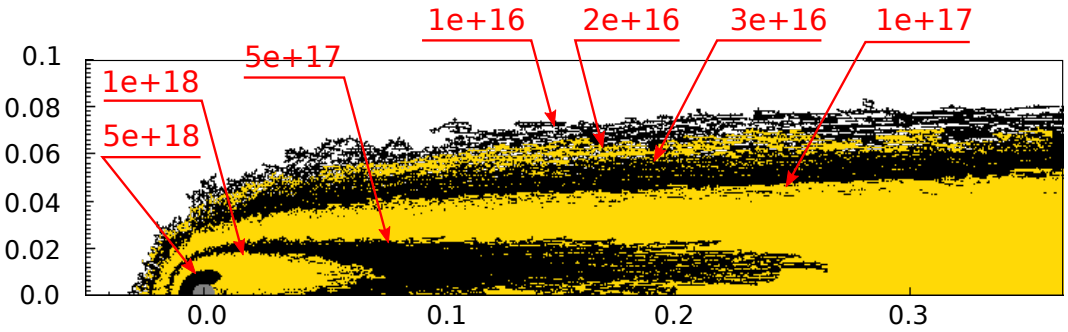

Figure 6.27: *Electrons number densities from the DSMC simulation. Distances in meters.*

It should be noted that since electrons do not recombine in the current DSMC simulation, the only reason for the number density diminishment is mass diffusion. The idea is now to find the streamline carrying the (more or less) maximum in electrons density and applying the *LARSEN* solver to that line to compute recombination. The DSMC simulation could then be scaled accordingly along the x axis, to also include free electrons recombination. In order to find the line of maxima in the electrons number densities, some vertical sections were considered and the maximum was found to be located at about $y = 10$ mm. The corresponding streamline, shown in Figure 6.28, was then picked and passed to *LARSEN*, starting at 20 mm past the meteor center.

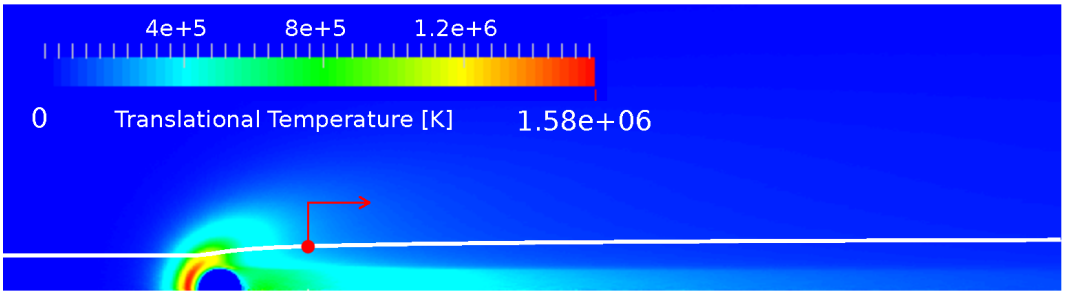

Figure 6.28: *Streamline holding the maximum number density for electrons, superimposed on the translational temperature field for heavy species.*

Figure 6.29 shows the number density of electrons along this streamline. As can be seen, data is quite scattered: this is due to the fact that electrons are a minor species in the problem, differing from the most present species by some orders of magnitude, a non-scattered solution would thus require an even bigger number of simulated particles or much longer averaging times.

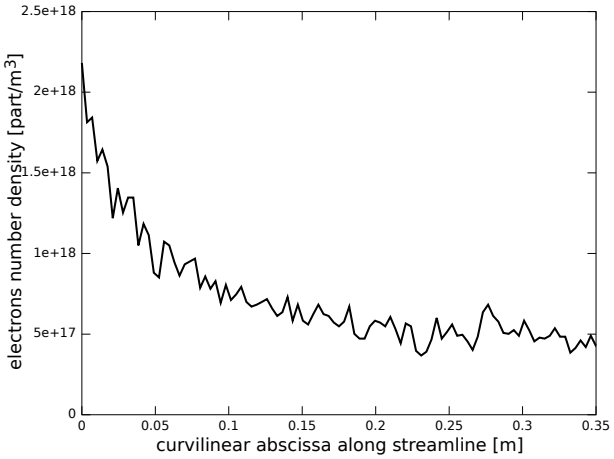

Figure 6.29: *Number density of free electrons along the streamline.*

LARSEN T - Tv on the streamline, adiabatic assumption

The extracted streamline is now used as an input for *LARSEN*, run with the two-temperatures nonequilibrium model: translational and vibrational temperatures at the *beginning* of the streamline are taken as initial conditions, as well as the mass fraction of chemical species. The density and velocity fields along the streamline are taken as external datum.

Since the DSMC simulation returns a translational temperature T_t and a rotational T_r (see Figure 6.30), the first step is merging them into one only temperature T, accordingly to the formulation of the two-temperatures model. This can be done by averaging the temperatures with the mass fractions of molecules (showing rotational energy) as follows:

$$T = \frac{T_t \sum_{i \in \mathcal{S}} y_i + T_r \sum_{i \in \mathcal{M}} y_i}{\sum_{i \in \mathcal{S}} y_i + \sum_{i \in \mathcal{M}} y_i} \tag{6.1}$$

where the subscript $\mathcal{S}$ refers to all species, while $\mathcal{M}$ refers to molecular species and y_i are the mass fractions.

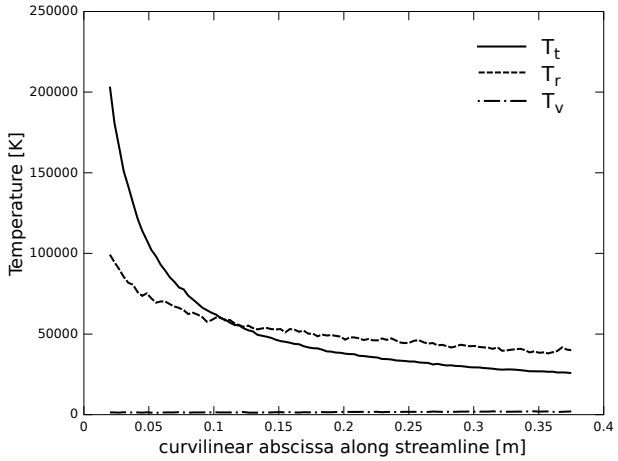

Figure 6.30: *Translational, rotational and vibrational temperatures along the streamline.*

Just as done in the previous section, two computations can be performed: one by conserving the total enthalpy of the fluid particle (adiabatic hypothesis) and the other one by including diffusive effects. This time, results for the adiabatic simulation are even more interesting, since they lead to the physically non acceptable solution shown in Figure 6.31, with the temperature reaching 0 K.

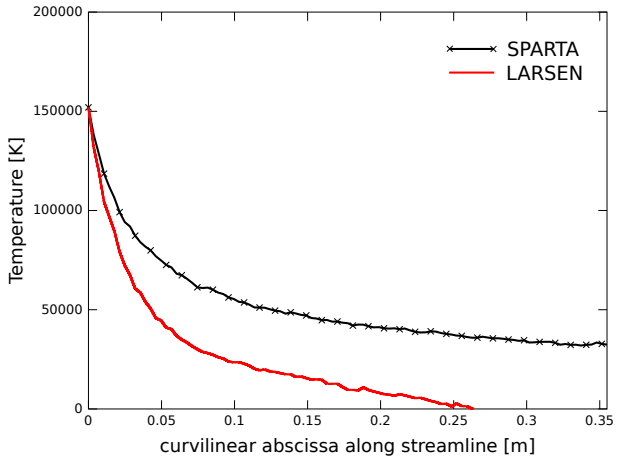

Figure 6.31: *Translational temperature T along the streamline. LARSEN conserves the particle enthalpy.*

In order to understand this result, we should refer to Figure 6.32, showing the enthalpy of the fluid particle along the streamline (from DSMC solution).

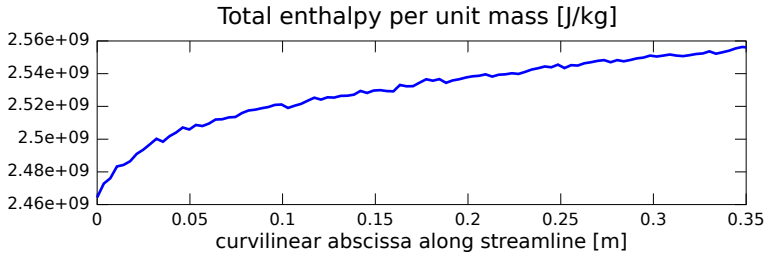

Figure 6.32: *Total enthalpy along streamline from DSMC simulation*

This plot shows that the total enthalpy is not conserved but increases, meaning that energy is being transferred to the particle due to diffusion effects. It's reasonable to think that this energy will partially produce an acceleration of the flow and thus become kinetic energy. In the adiabatic hypothesis, however, all the energy flowing to the kinetic energy term $u^2/2$ necessarily comes from the internal enthalpy, since no external flux is contemplated:

$$H = h + \frac{u^2}{2} = const \rightarrow h_1 = h_0 + \frac{u_0^2}{2} - \frac{u_1^2}{2} \tag{6.2}$$

Since the enthalpy at the beginning of the streamline h_0 is known, as well as the velocity at the beginning and at the end of the streamline u_0 and u_1, the final enthalpy can be computed and leads a (physically unacceptable) negative value.

h_0	2.167×10^8 J/kg
u_0	67292 m/s
u_1	70598 m/s
h_1	-1.117×10^7 J/kg

Of course, *LARSEN* crashes when a negative temperature is reached, experiencing some troubles in computing chemical rates and transfer terms for example.

To sum up, the behavior of the *LARSEN* solver in this case is attributable to the fact that we are feeding it with a velocity field that is *too much* inconsistent with the conserved enthalpy hypothesis.

LARSEN T - Tv on the streamline with diffusion effects

The computation on the streamline is here repeated, this time including effects of diffusion.

Figure 6.33 shows the translational and vibrational temperatures computed by *LARSEN*, plotted against the previous results from *SPARTA*. A small deviation is seen in the translational temperature and can be attributed to an increased translational-vibrational coupling computed by *LARSEN*. In fact, the vibrational temperature is predicted by *LARSEN* to relax much faster than what happens in the DSMC simulation, based on the Larsen-Borgnakke model. Further efforts should be dedicated to make sure that the chosen parameters make the two descriptions consistent or at least acceptable. Also, by plotting the mass fractions (Figure 6.34) we can see that *LARSEN* predicts higher dissociation and ionization rates, we might then expect a lower temperature.

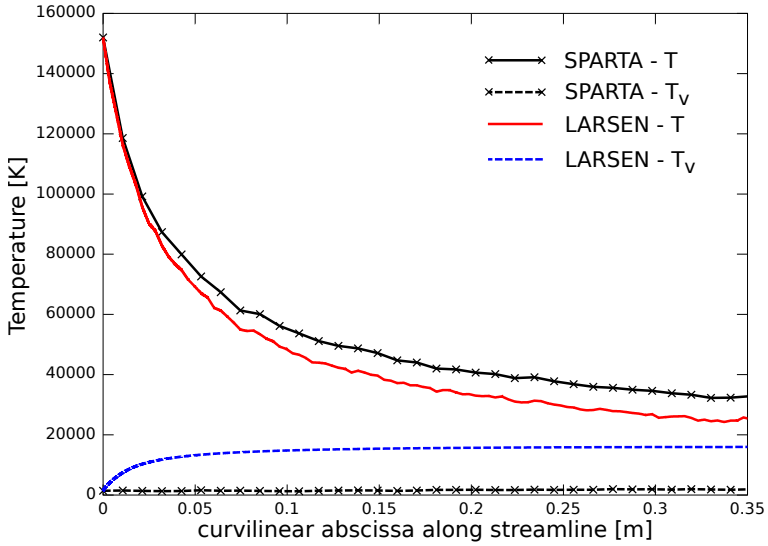

Figure 6.33: *Temperatures recomputed by LARSEN including diffusion and DSMC results by SPARTA*

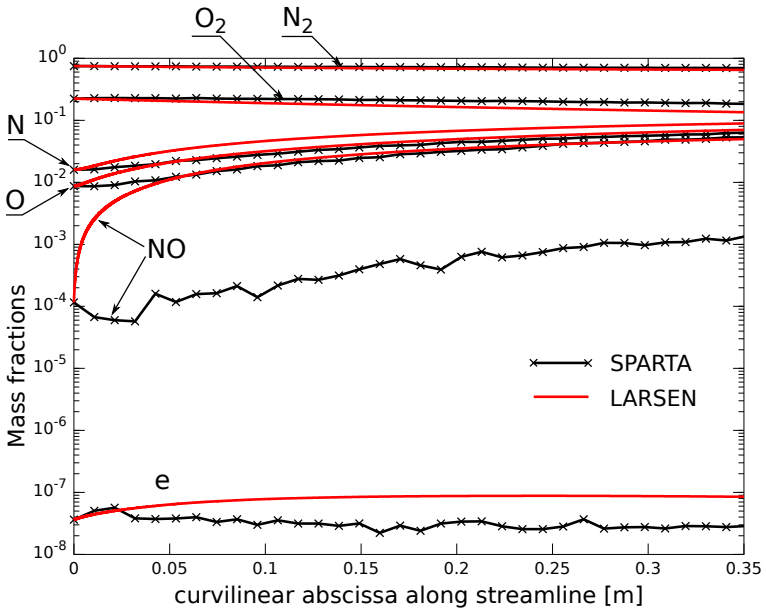

Figure 6.34: *Mass fractions of neutral species and of electrons as computed by LARSEN and SPARTA*

The *molar* fraction of electrons along the streamline as computed by *LARSEN* is shown in Figure 6.35. It should be recalled that in the *LARSEN* solver, mass diffusion is neglected and thus also the total electrical charge inside a fluid particle is necessarily conserved. This implies that the only mean of diminishing the number of electrons in the *LARSEN* computation is through recombination, since diffusion is forbidden.

Once computed by *LARSEN*, the effect of recombination could be heuristically applied to the DSMC simulation *a posteriori*, by properly scaling the DSMC-computed electrons number density with the fraction of recombined electrons. However, an analysis of the *molar fraction* of electrons (Figure 6.35) shows that after a first region where *LARSEN* predicts quite different rates with respect to *SPARTA*, recombination starts (slightly) diminishing the number of electrons only at around 0.2 m from the beginning of the streamline. This result indicates that the extension of the ionized trail is for this testcase (much) bigger than the analyzed domain, since the quantity of electrons is still of the same order of magnitude as the starting point.

The best (and more obvious) solution to this problem would be extending the computational domain for the DSMC computation, the only problem being

that for long trails this way becomes too much demanding. Another possibility would be performing an approximated analytical modeling of the wake in terms of velocity and density, that will eventually reach free-stream values as the distance from the meteoroid increases. Once a model for the flowfield is obtained, *LARSEN* can be run again, providing an estimation for the trail extension.

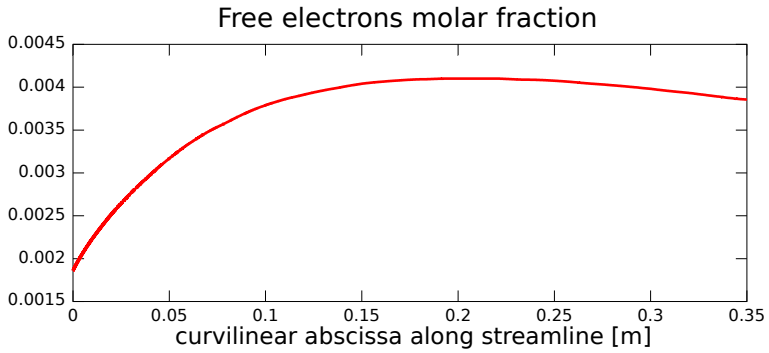

Figure 6.35: *Molar fraction of free electrons, computed by LARSEN*

CHAPTER 7

Conclusions

This chapter resumes the results obtained in this work and points out some improvements to be apported to the developed Lagrangian solver, as well as some future work directions.

7.1 Summary of results

In this work, a Lagrangian solver for nonequilibrium flows was developed with the goal of providing a lightweight tool that could be used to refine a previous numerical simulation by introducing more elaborated chemichal models and/or thermal nonequilibrium.

The solver works in the hypothesis that the velocity field is not-so-tightly-coupled with chemistry and takes it "as given" from a previous simulation, performed with whatever numerical or analytical method. This hypothesis was tested on 1D shockwaves, recomputing the thermochemical relaxation by introducing a more realistic chemical model and/or by allowing thermal nonequilibrium with the two-temperatures model. In the case of "chemistry refinement" the solver have brought drastic improvements to the initial rough solution, predicting equilibrium concentrations very close to the correct values, with some deviations in the nonequilibrium region. For the introduction of internal nonequilibrium, the results are a little less accurate, however the application of the Lagrangian solver halves the error with respect to the correct values.

The solver was then applied to 2D simulations of argon flows with three different degrees of rarefaction, the basic simulations being performed with the DSMC method. Both on the stagnation line and on streamlines close to the body, the solver is able to reproduce the starting temperature profile with

high accuracy.

Finally, the solver is applied to a rarefied air flow over a non-ablating meteoroid, entering the atmosphere at velocity of 72 km/s. The long-term goal of this computation is estimating the length of the ionized trail past the body for radar observation purposes. The goal is pursued by introducing recombination of free electrons, neglected in the baseline DSMC simulation. For this testcase the solver predicts that recombination is not enough to extinguish the electrons before the end of the computational domain, indicating that the ionized trail extends over several diameters (at least over 35 diameters) from the meteoroid. However, some non negligible discrepancies are encountered in the chemistry prediction and in the behavior of the vibrational temperature and some more investigation should be done. The problem of estimating the length of the ionized trail for this testcase is thus still open and would require bigger DSMC simulations or some approximated modeling of the wake.

7.2 Future work and possible applications

Much work could be done to improve the results of this thesis and to extend the current capabilities of the Lagrangian solver. This section suggests some future developments to which, according to the author, priority should be attributed. Also, some possible applications of the Lagrangian solver are reviewed.

Future work: Diffusion mass fluxes

Governing equations take into account diffusion fluxes of mass[1] through the term $\boldsymbol{\nabla} \cdot \boldsymbol{\rho_i U_i}$. This term can be estimated by postprocessing the baseline simulation, computing gradients in the x and y directions.

Implementing diffusion fluxes might be particularly useful for flows with high gradients of chemical species concentration. Also, this would allow the Lagrangian solver to take into account the process of *elemental demixing*. Diffusion mass fluxes will be necessary for example to address the study of an ablating meteor using the Lagrangian solver: with diffusive mass fluxes, the integration could start from free-stream conditions and as the region near the surface -full of ablation products- is reached, the mass fluxes automatically introduce the new species in the computation.

[1] Should be noted that the diffusive flux of *energy* is already taken into account via the term $\mathcal{Q}$ in the governing equations.

Future work: N-temperatures model

Equations for the N-temperatures model are shown in this work in the Lagrangian framework. Extending the two-temperatures model might be very useful: for example, including rotational nonequilibrium in the Lagrangian solver could be necessary for properly treating many rarefied flows. Generalizing the solver for an N-temperatures model is mainly a matter of obtaining the energy transfer terms from the thermodynamic library, implementing them from scratch where needed.

Care should be put in how the thermochemical library computes the chemical rates, where geometrical averages of the temperatures might be needed and may lead to discrepancies with respect to literature results.

Future work: Radiation

Some simple models for radiation could be implemented in the solver, providing a mechanism for energy loss and thus cooling-down of the particle.

Future work: Ablating meteoroid

The simulation performed to the meteoroid could be improved by adding simple evaporation models for the surface. A first approach could be treating a purely iron meteoroid, computing the ablation with simple evaporation models. Given the surface temperature, it's possible to obtain the vapor pressure of evaporating iron, and thus the vapor density. This value can then be implemented into a DSMC software as a surface blowing.

Including ablation might provide appreciable changes in the concentration of electrons, since the ionization potential of metals is typically way lower than that of air species.

Application: Chemistry of pseudo-species

For some mixtures, very detailed and accurate models have been obtained for the energy exchange among internal energy modes. The approach, referred to as "state-to-state", treats rotational and vibrational levels as chemical species ("pseudo-species") and the probability that an interaction would result in a change of state is recast in the form of chemical rates, see for example Bruno [14]. The number of pseudo-species is *very* high for some models, so much that a direct approach to the problem with CFD or DSMC methods is almost prohibited. Due to its inherently lightweigth structure, the Lagrangian solver

might be a suitable tool for applying the detailed state-to-state description to non-trivial geometries.

Implementing a state-to-state approach basically reduces to creating a mixture and storing all the rates for reactions among pseudo-species.

Application: Plasma frequency and blackout problems

Due to the possibility of computing ionization and recombination, the Lagrangian solver might be applied to the estimation of the peak plasma frequency for atmospheric entry problems. The plasma frequency, tied to the square root of electrons number density, can be used to estimate the points of communication blackout along the entry trajectory of space capsules. Recombination of free electrons can be introduced by the Lagrangian solver as done for the meteoroid ionized trail. Moreover, the Lagrangian solver gives the opportunity of having a solution not based on the TCE model, usual standard for DSMC simulations.

Application: Recombination in the plume of plasma thrusters

The plume of plasma thrusters is studied for various purposes, such as the interaction with parts of the spacecraft structure. The plume usually works in extremely rarefied environment, the DSMC being a useful tool, to which electrons recombination could be added with the Lagrangian solver.

Appendices

APPENDIX A

Equations for the N-temperatures model

In this appendix the derivation of the equations implemented in the Lagrangian solver is shown with a little more detail with enough generality to provide the basis for an N-temperatures implementation.

First of all the mass conservation equation for the *i-th* chemical species can be recast in Lagrangian form by introducing the species mass fraction $y_i = \rho_i/\rho$:

$$\dot{\omega}_i - \boldsymbol{\nabla}\cdot\boldsymbol{J}_i \;=\; \frac{\partial \rho_i}{\partial t} + \boldsymbol{\nabla}\cdot(\rho_i \boldsymbol{u}) \;=\; \frac{\partial}{\partial t}(\rho y_i) + \boldsymbol{\nabla}\cdot(\rho y_i \boldsymbol{u}) \;= \\ \rho\left[\frac{\partial y_i}{\partial t} + \boldsymbol{u}\cdot\boldsymbol{\nabla} y_i\right] + y_i\cancel{\left[\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla}\cdot(\rho\boldsymbol{u})\right]} \quad \text{(A.1)}$$

where the second term at the right hand side is the mass conservation for the mixture and is thus identically zero. The material derivative can be introduced:

$$\frac{\partial \bullet}{\partial t} \;+\; \boldsymbol{u}\cdot\boldsymbol{\nabla}\bullet \;=\; \frac{\mathrm{D}\bullet}{\mathrm{D}t} \quad \text{(A.2)}$$

and the mass equation for the *i-th* species becomes:

$$\frac{\mathrm{D}y_i}{\mathrm{D}t} = \frac{\dot{\omega}_i - \boldsymbol{\nabla}\cdot\boldsymbol{J}_i}{\rho} \quad \text{(A.3)}$$

In the further hypothesis of zero diffusion fluxes, we have:

$$\frac{\mathrm{D}y_i}{\mathrm{D}t} = \frac{\dot{\omega}_i}{\rho} \quad \text{(A.4)}$$

Now, the first step in developing the N-temperatures model is choosing which degrees of freedom are in equilibrium among each others ($T = T_r$ and $T_v = T_{el} = T_e$ in the case of two temperatures model). All the steps are then performed with the idea in mind of merging together the internal enthalpies relative to DOFs that are in equilibrium.

By labeling with $\mathcal{P}$ a pool of degrees of freedom supposed in equilibrium among each other (such as vibrations, electronic excitations and free electrons translation in the case of 2T model), one can write an equation for the temperature $T_{\mathcal{P}}$. The first step in doing this is taking a chemical species and writing a conservation equation for the fraction of its enthalpy belonging to the pool, here called $h_i^{\mathcal{P}}$:

$$\frac{\partial}{\partial t}\left(\rho_i h_i^{\mathcal{P}}\right) + \boldsymbol{\nabla}\cdot\left(\rho_i \boldsymbol{u} h_i^{\mathcal{P}}\right) = -\boldsymbol{\nabla}\cdot\left(\boldsymbol{q}_i^{\mathcal{P}} + h_i^{\mathcal{P}}\boldsymbol{J}_i\right) + \Omega_i^{\mathcal{P}} \tag{A.5}$$

In the case of T-T_v model for example, we'd write:

$h_i^P = h_i^v + h_i^{el}$ $\quad \leftarrow$ for molecules

$h_i^P = h_i^{el}$ $\quad \leftarrow$ for atoms

$h_i^P = h_e$ $\quad \leftarrow$ for free electrons

The l.h.s. of equation A.5 can be elaborated by introducing the species mass fraction y_i:

$$\frac{\partial}{\partial t}\left(\rho_i h_i^{\mathcal{P}}\right) + \boldsymbol{\nabla}\cdot\left(\rho_i \boldsymbol{u} h_i^{\mathcal{P}}\right) = \rho\left[\frac{\partial}{\partial t}\left(y_i h_i^{\mathcal{P}}\right) + \boldsymbol{u}\cdot\boldsymbol{\nabla}\left(y_i h_i^{\mathcal{P}}\right)\right] + y_i h_i^{\mathcal{P}}\cancel{\left[\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla}\cdot(\rho\boldsymbol{u})\right]} \tag{A.6}$$

where the mass conservation for the mixture elides the second term at the r.h.s; by also introducing the mass conservation for the *i-th* species in Lagrangian form, an equation for the enthalpy $h_i^{\mathcal{P}}$ along the streamline is obtained:

$$\frac{\mathrm{D}h_i^{\mathcal{P}}}{\mathrm{D}t} = \frac{-\boldsymbol{\nabla}\cdot\left(\boldsymbol{q}_i^{\mathcal{P}} + h_i^{\mathcal{P}}\boldsymbol{J}_i\right) + \Omega_i^{\mathcal{P}} - h_i^{\mathcal{P}}\dot{\omega}_i}{\rho y_i} \tag{A.7}$$

where the term $\Omega_i^{\mathcal{P}}$ accounts for the energy flowing from the *i-th* chemical species mode belonging to the pool $\mathcal{P}$ to other modes. This equation will be used in a few lines.

The next step is writing the material derivative of the total enthalpy of the pool $h^{\mathcal{P}}$:

$$\frac{\mathrm{D}h^{\mathcal{P}}}{\mathrm{D}t} = \frac{\mathrm{D}}{\mathrm{D}t}\sum_{i\in\mathcal{S}} y_i h_i^{\mathcal{P}} =$$

$$\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{D}h_i^{\mathcal{P}}}{\mathrm{D}t} + \sum_{i\in\mathcal{S}} h_i^{\mathcal{P}} \frac{\mathrm{D}y_i}{\mathrm{D}t} = \left(\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{d}h_i^{\mathcal{P}}(T_{\mathcal{P}})}{\mathrm{d}T_{\mathcal{P}}}\right) \frac{\mathrm{D}T_{\mathcal{P}}}{\mathrm{D}t} + \sum_{i\in\mathcal{S}} h_i^{\mathcal{P}} \frac{\mathrm{D}y_i}{\mathrm{D}t} \quad \text{(A.8)}$$

Writing the previous equation is not really useful, except for the fact that it's a nice way to see the next step, namely recalling that:

$$\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{D}h_i^{\mathcal{P}}}{\mathrm{D}t} = \left(\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{d}h_i^{\mathcal{P}}(T_{\mathcal{P}})}{\mathrm{d}T_{\mathcal{P}}}\right) \frac{\mathrm{D}T_{\mathcal{P}}}{\mathrm{D}t} \quad \text{(A.9)}$$

and thus:

$$\frac{\mathrm{D}T_{\mathcal{P}}}{\mathrm{D}t} = \left(\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{D}h_i^{\mathcal{P}}}{\mathrm{D}t}\right) \Big/ \left(\sum_{i\in\mathcal{S}} y_i c_{p,i}^{\mathcal{P}}(T_{\mathcal{P}})\right) \quad \text{(A.10)}$$

By now exploiting the equation for $\mathrm{D}h_i^{\mathcal{P}}/\mathrm{D}t$, under the hypothesis of no heat and diffusion fluxes for the pool energy:

$$\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{D}h_i^{\mathcal{P}}}{\mathrm{D}t} = \sum_{i\in\mathcal{S}} \frac{\Omega_i^{\mathcal{P}} - \dot{\omega}_i h_i^{\mathrm{P}}}{\rho} = \frac{\Omega^{\mathcal{P}} - \sum_{i\in\mathcal{S}} \dot{\omega}_i h_i^{\mathcal{P}}}{\rho} \quad \text{(A.11)}$$

where $\Omega^{\mathcal{P}}$ accounts for the energy transferred to the energy pool from other pools. The equation for the pool temperature $T_{\mathcal{P}}$ then reads:

$$\frac{\mathrm{D}T_{\mathcal{P}}}{\mathrm{D}t} = \left(\frac{\Omega^{\mathcal{P}} - \sum_{i\in\mathcal{S}} \dot{\omega}_i h_i^{\mathcal{P}}}{\rho}\right) \Big/ \left(\sum_{i\in\mathcal{S}} y_i c_{p,i}^{\mathcal{P}}\right) \quad \text{(A.12)}$$

After deriving an equation for each internal energy pool, the last step is deriving one for the energies in equilibrium with translation, whose temperature is simply referred to as T. Starting from the conservation of total enthalpy $\mathrm{D}H/\mathrm{D}t = \mathcal{Q}$, the enthalpy H is split into its contributions, namely the kinetic energy $u^2/2$ plus all the energy (enthalpy) pools:

$$\frac{\mathrm{D}H}{\mathrm{D}t} = \frac{\mathrm{D}h^t}{\mathrm{D}t} + \sum_p \frac{\mathrm{D}h^p}{\mathrm{D}t} + \frac{\mathrm{D}u^2/2}{\mathrm{D}t} = \mathcal{Q} \quad \text{(A.13)}$$

the pool that includes the translational enthalpy was kept separated and labeled h^t. In the case of two-temperatures model, the sum over the pool reduces to the vibration - electronic - free-electrons pool.

Now, since h^t is function of the translational temperature T (and of the mixture composition), it is splitted into the contributions from each chemical species and then the chain rule is applied to let emerge the material derivative of T:

$$\frac{\mathrm{D}h^t}{\mathrm{D}t} = \frac{\mathrm{D}}{\mathrm{D}t}\left(\sum_i y_i h_i^t\right) = \sum_i \frac{h_i^t \dot{\omega}_i}{\rho} + \left(\sum_i y_i c_{p,i}^t\right)\frac{\mathrm{D}T}{\mathrm{D}t} \tag{A.14}$$

so, $\mathrm{D}T/\mathrm{D}t$ can be inserted into the total enthalpy equation, leading to:

$$\frac{\mathrm{D}T}{\mathrm{D}t} = \left(-\frac{\mathrm{D}u^2/2}{\mathrm{D}t} - \sum_p \frac{\mathrm{D}h^p}{\mathrm{D}t} - \sum_{i\in\mathcal{S}} \frac{h_i^t \dot{\omega}_i}{\rho} + \mathcal{Q}\right) \Big/ \left(\sum_i y_i c_{p,i}^t\right) \tag{A.15}$$

The material derivatives for the enthalpies of the pools can now be explicited:

$$\sum_p \frac{\mathrm{D}h^p}{\mathrm{D}t} = \sum_p \left(\sum_{i\in\mathcal{S}} y_i \frac{\mathrm{D}h_i^p}{\mathrm{D}t} + \sum_{i\in\mathcal{S}} \frac{h_i^p \dot{\omega}_i}{\rho}\right) = \sum_{i\in\mathcal{S}} y_i \frac{\mathrm{D}\left(\sum_p h_i^p\right)}{\mathrm{D}t} + \sum_{i\in\mathcal{S}} \frac{\left(\sum_p h_i^p\right)\dot{\omega}_i}{\rho} \tag{A.16}$$

by reinserting this term into the master equation, the enthalpies multiplying chemical rates join the translational enthalpy, giving the total internal enthalpy h of the species, for the basic identity: $h_i = h_i^t + \sum_p h_i^p$. By also exploiting the equation for $\mathrm{D}h^p/\mathrm{D}t$ in the hypothesis of null diffusion and heat flux:

$$\frac{\mathrm{D}T}{\mathrm{D}t} = \left[\mathcal{Q} - \frac{\mathrm{D}u^2/2}{\mathrm{D}t} - \sum_{i\in\mathcal{S}} \frac{h_i^{tot}\dot{\omega}_i}{\rho} - \left(\Omega^t - \sum_{i\in\mathcal{S}} \frac{h^{non\text{-}t}\dot{\omega}_i}{\rho}\right)\right] \Big/ \left(\sum_{i\in\mathcal{S}} y_i c_{p,i}^t\right) \tag{A.17}$$

where the term Ω^t accounts for the energy flowing from the pool that includes the translational enthalpy to the others and $h^{non\text{-}t}$ refers to all the pools except the one including the translational DOF.

An equation for the translational temperature for the case of thermal equilibrium can be easily obtained starting from equation A.13 and considering that the "translational pool" is the only one.

APPENDIX B

Chemical reactions for DSMC simulations

Table B.1 shows the set of chemical reactions used in this work to simulate a mixture of 11 air species in DSMC simulations. Reactions follow the Arrhenius form:

$$k = AT^b e^{-E_a/k_B T} \tag{B.1}$$

where k_B is the Boltzmann constant, E_a the activation energy and A and b are respectively the Arrhenius prefactor and exponend.

In the table are shown the parameters A, b and E_a, plus the number of internal degrees of freedom N_i (needed by the TCE model) and the overall reaction energy E^{react}(positive for exothermic). The TCE model is explained by Bird [4]. Another useful reference is Goldsworthy and Macrossan [20].

Reaction	N_i	E_a	A	b	E^{react}
$O2 + N \rightarrow O + O + N$	1.0	$8.197e-19$	$1.660e-8$	-1.5	$-8.197e-19$
$O2 + NO \rightarrow O + O + NO$	1.0	$8.197e-19$	$3.321e-9$	-1.5	$-8.197e-19$
$O2 + N2 \rightarrow O + O + N2$	1.0	$8.197e-19$	$3.321e-9$	-1.5	$-8.197e-19$
$O2 + O2 \rightarrow O + O + O2$	1.0	$8.197e-19$	$3.321e-9$	-1.5	$-8.197e-19$
$O2 + O \rightarrow O + O + O$	1.0	$8.197e-19$	$1.660e-8$	-1.5	$-8.197e-19$
$N2 + O \rightarrow N + N + O$	1.0	$1.561e-18$	$4.980e-8$	-1.6	$-1.561e-18$
$N2 + O2 \rightarrow N + N + O2$	1.0	$1.561e-18$	$1.162e-8$	-1.6	$-1.561e-18$
$N2 + NO \rightarrow N + N + NO$	1.0	$1.561e-18$	$1.162e-8$	-1.6	$-1.561e-18$
$N2 + N2 \rightarrow N + N + N2$	1.0	$1.561e-18$	$1.162e-8$	-1.6	$-1.561e-18$
$N2 + N \rightarrow N + N + N$	1.0	$1.561e-18$	$4.980e-8$	-1.6	$-1.561e-18$
$NO + N2 \rightarrow N + O + N2$	1.0	$1.043e-18$	$8.302e-15$	0.0	$-1.043e-18$
$NO + O2 \rightarrow N + O + O2$	1.0	$1.043e-18$	$8.302e-15$	0.0	$-1.043e-18$
$NO + NO \rightarrow N + O + NO$	1.0	$1.043e-18$	$8.302e-15$	0.0	$-1.043e-18$
$NO + O \rightarrow N + O + O$	1.0	$1.043e-18$	$1.862e-13$	0.0	$-1.043e-18$
$NO + N \rightarrow N + O + N$	1.0	$1.043e-18$	$1.862e-13$	0.0	$-1.043e-18$
$NO + O \rightarrow O2 + N$	0.0	$2.684e-19$	$1.389e-17$	0.0	$-2.684e-19$
$N2 + O \rightarrow NO + N$	0.0	$5.175e-19$	$1.069e-12$	-1.0	$-5.175e-19$
$O2 + N \rightarrow NO + O$	0.0	0.0	$4.601e-15$	-0.546	$2.684e-19$
$NO + N \rightarrow N2 + O$	0.0	0.0	$4.059e-12$	-1.359	$5.175e-19$
$O + N \rightarrow NO^+ + e$	0.0	$4.404e-19$	$8.766e-18$	0.0	$-4.404e-19$
$N + N \rightarrow N2^+ + e$	0.0	$9.319e-19$	$3.387e-17$	0.0	$-9.319e-19$
$O + O \rightarrow O2^+ + e$	0.0	$1.1128e-18$	$1.8580e-17$	0.0	$-1.1128e-18$
$NO^+ + N \rightarrow O + N2^+$	0.0	$4.832e-19$	$1.1956e-16$	0.0	$-4.832e-19$
$N2^+ + O \rightarrow N + NO^+$	0.0	0.0	$1.744e-18$	0.302	$4.832e-19$
$N2 + N^+ \rightarrow N + N2^+$	0.0	$1.684e-19$	$1.6605e-18$	0.5	$-1.684e-19$
$N2^+ + N \rightarrow N2 + N^+$	0.0	0.0	$1.295e-18$	0.5	$1.684e-19$
$NO^+ + N \rightarrow N2 + O^+$	0.0	$1.767e-19$	$5.6458e-17$	1.08	$-1.767e-19$
$N2 + O^+ \rightarrow N + NO^+$	0.0	0.0	$3.9708e-18$	-0.710	$1.767e-19$
$NO^+ + O \rightarrow O2 + N^+$	0.0	$1.767e-19$	$1.6605e-18$	0.5	$-1.767e-19$
$O2 + N^+ \rightarrow O + NO^+$	0.0	0.0	$3.040e-18$	-0.29	$1.767e-19$
$NO^+ + O \rightarrow N + O2^+$	0.0	$6.710e-19$	$1.1956e-17$	0.29	$-6.710e-19$
$O2^+ + N \rightarrow O + NO^+$	0.0	0.0	$8.918e-13$	-0.969	$6.710e-19$
$NO^+ + O2 \rightarrow NO + O2^+$	0.0	$4.501e-19$	$3.9853e-17$	0.41	$-4.501e-19$
$O2^+ + NO \rightarrow O2 + NO^+$	0.0	0.0	$3.990e-17$	0.41	$4.501e-19$
$O2^+ + N \rightarrow O2 + N^+$	0.0	$3.949e-19$	$1.4447e-16$	0.14	$-3.949e-19$
$O2^+ + O \rightarrow O2 + O^+$	0.0	$2.485e-19$	$6.6422e-18$	-0.09	$-2.485e-19$
$O^+ + O2 \rightarrow O + O2^+$	0.0	0.0	$4.993e-18$	-0.004	$2.485e-19$
$O2^+ + N2 \rightarrow O2 + N2^+$	0.0	$5.619e-19$	$1.6439e-17$	0.0	$-5.619e-19$
$N2^+ + O2 \rightarrow N2 + O2^+$	0.0	0.0	$4.5899e-18$	-0.037	$5.619e-19$
$O^+ + N2 \rightarrow O + N2^+$	0.0	$3.148e-19$	$1.5111e-18$	0.0	$-1.148e-19$
$N2^+ + O \rightarrow N2 + O^+$	0.0	0.0	$4.118e-11$	-2.2	$1.148e-19$
$O^+ + NO \rightarrow O2 + N^+$	0.0	$3.673e-19$	$2.3248e-25$	1.90	$-3.673e-19$
$N^+ + O2 \rightarrow NO + O^+$	0.0	0.0	$2.443e-26$	2.102	$3.673e-19$
$O + e \rightarrow O^+ + e + e$	0.0	$2.188e-18$	$6.4761E3$	-3.78	$-2.188e-18$
$N + e \rightarrow N^+ + e + e$	0.0	$2.322e-18$	$4.1513E4$	-3.82	$-2.322e-18$

Table B.1: *Set of reactions for air mixture, used in DSMC simulations.*

Bibliography

[1] J. D. Anderson, *Hypersonic and High Temperature Gas Dynamics*, 1989, McGraw-Hill Book Company.

[2] W. G. Vincenti and C. J. Kruger, *Introduction to Physical Gas Dynamics*, 1965, Krieger Publishing Company.

[3] C. Park, *Nonequilibrium Hypersonic Aerothermodynamics*, Wiley.

[4] G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Clarendon Press, Oxford, 1994.

[5] P. M. Bellan, *Fundamentals of Plasma Physics*, 2006, Cambridge University Press.

[6] Ya. B. Zel'dovich and Yu. P. Raizer, *Physics of Shock Waves and High-Temperature Hydrodynamic Phenomena*, 2002, Dover Edition.

[7] L. Quartapelle and F. Auteri, *Fluidodinamica incomprimibile*, 2013, Casa Editrice Ambrosiana.

[8] L. Quartapelle and F. Auteri, *Fluidodinamica comprimibile*, 2013, Casa Editrice Ambrosiana.

[9] L. Landau and E. Teller, Phys. Z. Sow., 1936, vol. **10**, p. 34.

[10] E. E. Nikitin and J. Troe, Phys. Chem. Chem. Phys., 2008, vol. **10**, p. 1483.

[11] C. Park et al, *Chemical-Kinetic Parameters of Hyperbolic Earth Entry*, Journal of Thermophysics and Heat Transfer, 2001, vol. **15**, No. 1.

[12] E. D. Farbar, *Kinetic Simulation of Rarefied and Weakly Ionized Hypersonic Flow Fields*, PhD dissertation, The University of Michigan, 2010.

[13] W. Wagner, *A Convergence Proof for Bird's Direct Simulation Monte Carlo Method for the Boltzmann Equation*, Journal of Statistical Physics, 1992, vol. **10**.

[14] D. Bruno et al, *Direct simulation of non-equilibrium kinetics under shock conditions in nitrogen*, Chemical Physics Letters, 2002, vol. **360**, p. 31-37.

[15] N. V. Ardelyan et al, *Prebreakdown air ionization in the atmosphere*, Russian Journal of Physical Chemistry B, 2015, vol. **9**, p. 807818.

[16] A. Pellinen-Wannberg et al, *The Hyperthermal Ionization and High Absolute Meteor Velocities Observed with HPLA Radars*, in book: Modern Meteor Science An Interdisciplinary View, 2006, p. 627-632.

[17] F. Thivet, *Modelisation et calculs d'ecoulements hypersoniques en desequilibre chimique et thermodynamique*, PhD dissertation, Ecole Cent. Paris, 1992.

[18] A. J. Lofthouse, *Nonequilibrium Hypersonic Aerothermodynamics Using The Direct Simulation Monte Carlo and Navier-Stokes Models*, PhD dissertation, The Universty of Michigan, 2008.

[19] T. J. Scanlon et al, *Open Source DSMC Chemistry Modelling for Hypersonic Flows*.

[20] M. Goldsworthy, M. Macrossan, *Vibrational degrees of freedom in the Total Collision Energy DSMC chemistry model*, Department of Mechanical Engineering, Report No. 2009/01, University of Queensland, 2009.

[21] G. A. Bird, *The Q-K model for gas-phase chemical reaction rates*, Physics of Fluids, 2011, vol. **23**.