Detailed Numerical Simulation of Multi-Dimensional Plasma Assisted Combustion

Lionel Cheng

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (Toulouse INP)

Discipline ou spécialité :

Dynamique des fluides

Présentée et soutenue par :

M. LIONEL CHENG

le mardi 15 février 2022

Titre :

Simulation numérique détaillée multidimensionnelle de la combustion assistée par plasma

---

Ecole doctorale :

Mécanique, Energétique, Génie civil, Procédés (MEGeP)

Unité de recherche :

Centre Européen de Recherche et Formation Avancées en Calcul Scientifique (CERFACS)

Directeur(s) de Thèse :

MME BÉNÉDICTE CUENOT

M. OLIVIER VERMOREL

Rapporteurs :

M. KHALED HASSOUNI, UNIVERSITE PARIS 13

M. YIGUANG JU, PRINCETON UNIVERSITY NEW JERSEY

Membre(s) du jury :

MME ANNE BOURDON, CNRS PARIS, Président

M. BRUNO DESPRES, UNIVERSITE SORBONNE, Membre

MME BÉNÉDICTE CUENOT, CERFACS, Membre

MME DEANNA LACOSTE, KING ABDULLAH UNIVERSITY OF SC&TECH, Membre

M. MICHAËL BAUERHEIM, ISAE-SUPAERO, Invité(e)

M. OLIVIER VERMOREL, CERFACS, Invité(e)

M. VINCENT MOUREAU, CORIA, Membre

Remerciements

Cette thèse a été réalisée dans le cadre du projet ANR GECCO (Green Electrical Control of COmbustion) au CERFACS au sein de l'équipe CFD.

Je souhaite tout d'abord remercier ma directrice de thèse, Bénédicte Cuenot, pour m'avoir fait découvrir un sujet aussi passionant que la combustion assistée par plasma. La grande liberté et confiance qu'elle m'a accordées m'ont permis de ne jamais écarter de pistes et d'approfondir de nombreux axes de travail.
Merci ensuite à mon co-directeur, Olivier Vermorel, pour ses conseils précieux et sa relecture toujours très pertinente qui m'ont permis d'avoir plus de clarté dans mon discours.
Je remercie également chaleureusement Anne Bourdon, chercheuse au LPP, qui m'a en pratique beaucoup encadré pendant ces trois ans. Son expertise des plasmas de décharge m'a toujours guidé dans la bonne direction.
Merci aussi à Deanna Lacoste, chercheuse à KAUST, pour m'avoir eclairé en apportant son avis d'experimentatrice si important lorsque l'on fait des simulations numériques.
Je remercie aussi Michaël Bauerheim, pour m'avoir permis de travailler sur les réseaux de neurones pour résoudre des équations aux dérivées partielles, chose qui n'était pas prévu au début, mais qui m'a permis de découvrir un monde passionant.

Je souhaite maintenant remercier les autres doctorants avec qui j'ai travaillés en étroite collaboration pendant ces trois ans dans la team plasma. Tout d'abord Nicolas Barleon, qui partageait un sujet commun avec le mien, et avec qui j'ai réalisé la majeure partie du travail en combustion assistée par plasma. Les nombreux débats, réflexions au tableau m'ont beaucoup appris et apportés: les compliments sur le travail réalisé par le jury sur cette partie t'étaient aussi destinés. Ensuite Guillaume Bogopolsky et Ekhi Ajuria pour la partie réseaux de neurones et équation de Poisson où nous avons passé de nombreuses heures à interpréter la manière dont le réseau comprenait cette equation. Merci en particulier à Guillaume de m'avoir appris à coder rigoureusement et d'avoir déclenché une passion pour le code.
Merci à la promotion de doctorants rentrée en même que moi et après pour les nombreux échanges scientifiques et les rigolades pendant les pauses déjeuners, je pense particulièrement à Benjamin, Antoine, Adrien et Jérôme.
Merci à l'équipe CSG (Computer Support Group) du CERFACS pour tout le travail de l'ombre qu'ils réalisent et qui permet à l'ensemble du laboratoire et des calculateurs de fonctionner parfaitement et sans encombre.

Ses trois ans de thèse n'auraient pas été pareils sans les nombreux amis qui m'ont soutenu et supporté et avec qui j'ai vécu des moments inoubliables: mes colocs au Perchoir Amaury et Adrien, la team fouenantaise Tristan, David, Joséphine, Laetitia et Marc ainsi que tous les potos d'amour, les Flying Donkeys Paul D., Quentin et Théo, les frenchies à LA Alexandre, Paul C. et Aurélien, mon amie d'enfance Elisa, les toulousains Marine, Marion, Ludivine, Maylis, les débiles de prépa Thomas et Corentin, les Atomes de Savoie Tristan, Clément, Brice et Bruno, les camarades de promo Matthieu, Marc, Thibault, Andrei, Benoît, Antoine, Nicolas, Victor, Louis ainsi qu'Hortense.

Enfin un grand merci à ma famille: mon père, ma mère, ma soeur, ma tante et mon

oncle, d'être venue de si loin assister à ma soutenance ainsi que pour leur soutien permanent lors de la réalisation de ce travail. Je souhaite remercier mon père de m'avoir inculqué cette capacité à travailler sans relache qui m'ont permis de vaincre bien des obstacles pendant ces trois ans. Je remercie ma mère pour son désir de toujours s'améliorer qu'elle m'a inculqué et qui me rend si perfectionniste et attentionné au détail.

Résumé: Les interactions entre flammes et plasmas sont le fil directeur de ce travail. Les décharges nanoseconde répétées (NRP en anglais) sont des plasmas hors-équilibre qui ont montré des caractéristiques intéressantes pour le contrôle de la combustion. Ces décharges peuvent interagir avec les flammes à travers non seulement un effet thermique, mais aussi un effet chimique à travers la production d'espèces réactives tel que les radicaux. Dans cette thèse, des simulations entièrement couplées de combustion assistée par plasma sont visées. Pour ce faire, la simulation de décharges plasmas a été implémentée dans le code de plasmas froids AVIP. Les méthodes numériques, ainsi que de nombreux cas de validations pour chaque ensemble d'équations, sont tout d'abord présentées. Ces équations comprennent les équations de dérive-diffusion couplées avec l'équation de Poisson pour les décharges plasmas. AVIP est couplé à AVBP qui résout les équations de Navier-Stokes réactives pour décrire la combustion. Dans une deuxième partie, nous commençons par construire une chimie détaillée pour les mélanges méthane-air à travers des calculs de réacteurs 0D. Cette chimie est ensuite réduite pour être applicable à des calculs multi-dimensionnels. La capacité d'AVIP à simuler des décharges plasmas, aussi appelés streamers, est ensuite montrée sur des chimies simples. Toutes les parties du code sont enfin rassemblées dans un calcul multi-dimensionel enitèrement couplé pour étudier l'allumage d'un mélange méthane-air en utilisant des décharges NRP. Au vu de l'expérience gagnée sur ces simulations entièrement couplées, des modèles phénoménologiques sont proposés. Dans une dernière partie, des réseaux de neurones sont utilisées pour résoudre l'équation de Poisson. Ceux-ci ont le potentiel pour accélérer les simulations plasmas par rapport aux solveurs linéaires classiques.
Mots-clés: Combustion Assistée par Plasma, Méthodes Numériques, Réseaux de Neurones, équation de Poisson

Abstract: Interaction between flames and plasmas are the guiding thread of this work. Nanosecond Repetitively Pulsed (NRP) discharges are non-thermal plasmas which have shown interesting features for combustion control. They can interact with flames not only through heat, but also chemically by producing active species. In this work, fully-coupled plasma assited combustion simulations are targeted. To achieve this goal, plasma discharge capabilities are built in the low temperature plasma code, AVIP. The corresponding numerical methods, as well as validation cases regarding each set of equations, are first presented. To simulate plasma discharges, the coupled drift-diffusion equations and the Poisson equation are considered. AVIP is coupled to the AVBP code which solves the reactive Navier-Stokes equations to describe combustion phenomena. In a second part, we start by constructing and validating a fully-detailed chemistry for methane-air mixtures in zero-dimensional reactors before reducing it for multi-dimensional simulations. The multi-dimensional streamer simulation capabilities of the code are then assessed using simple chemistries. All the validated parts of the code come together in a fully detailed simulation of ignition using NRP discharges. We finish by discussing phenomenological models built upon the knowledge that we gained from fully-detailed simulations. In a last part, finally, attempt to solve the Poisson and generalized Poisson equations using neural networks, which have a potential for speedup compared to classical linear solvers, is carried out.
Keywords: Plasma Assisted Combustion, Numerical Methods, Neural Networks, Poisson equation

Contents

Nomenclature

Below are lists of the main Acronyms, Roman characters and Greek characters used throughout this work. This is not an exhaustive list and clashes are sometimes encountered.

Symbol	Description
BC	Boundary condition
CFL	Courant-Friedrichs-Lewy
DL	Deep Learning
EEDF	Electron Energy Distribution Function
GRI	Generalized Riemann Invariants
HLL	Harten Lax van Leer
HLLC	Harten Lax van Leer Contact
ISG	Improved Scharfetter Gummel
KAUST	King Abdulla University of Science and Technology
LW	Lax-Wendroff
LLW	Limited Lax-Wendroff
MLPs	Multilayer perceptrons
NRP	Nanosecond Repetitively Pulsed (discharges)
NSCBC	Navier-Stokes Characteristic Boundary Conditions
NS	Navier-Stokes
PAC	Plasma Assisted Combustion
RH	Rankine Hugoniot
RK	Runge-Kutta
SGD	Stochastic Gradient Descent
SG	Scharfetter Gummel

Table 1: Acronyms

Symbol	Description
A	Matrix
a	Sound speed
$\mathbf{B}$	Magnetic field
c_p	Specific heat capacity at constant pressure
D	Diffusion coefficient
G	Green function or Gibbs free energy
e_s	Sensible energy
E	Total non-chemical energy
e_t	Total energy
$\mathbf{E}$	Electric field
$\mathbf{F}$	Flux for euler and drift-diffusion equations
F_0	Electron Energy Distribution Function
f_α	Distribution function of species α
$\mathbf{J}$	Current density
k_B	Boltzmann constant
Le_k	Lewis number of species k
n	Number density
p	Pressure
q_i	Charge of species i
$\mathbf{q}$	Heat flux
$\mathcal{Q}$	Partition function
$\dot{Q}_j$	Molar reaction j rate
R	Perfect gas constant
S_{ph}	Photoionization source term
s_L^0	Laminar flame speed
Sc_k	Schmidt number of species k
T	Gas temperature
T_e	Electron temperature
u	Scalar variable or velocity
$\mathbf{u}$	Mean velocity of the mixture
$\mathbf{U}$	Vector of conservative variables
$\mathbf{V}$	Velocity or voltage
$\mathbf{W}$	Drift-velocity or vector of primitive variables
W	Mean molecular weight of the mixture
W_k	Molecular weight of species k
Y_k	Mass fraction of species k

Table 2: Roman characters

Symbol	Description
β	Sweby limiter parameter
$\mathbf{\Gamma}$	Particle density flux
δ_L^0	Thermal layer thickness
ε_0	Vacuum permittivity
η	Viscosity coefficient
θ	Cylindrical coordinates angle
λ	Thermal conductivity
λ_D	Debye length
μ	Mobility coefficient
μ_k	Chemical potential of species k
ρ	Mass density
σ	Cross-section or activation function
τ	Cell of the computational domain
ϕ	Electromagnetic potential
ψ	Wave function
Ω	Domain of computation
$\dot{\Omega}$	Interior of computation domain
$\partial\Omega$	Border of computation domain
ω_p	Plasma oscillation pulsation frequency

Table 3: Greek characters

Chapter 1

Introduction: Flames and plasmas

Contents

1.1 Combustion challenges and needs

Combustion is and will be in the near future the primary source of energy in the world due to the high efficiency and flexibility of combustion engines. At the time of writing, it accounts for more than 80% of the world energy production. The impact on air pollution and climate change is however a major concern of the combustion of fossil fuels and needs to be adressed [Chu and Majumdar, 2012].

In ground transportation, the development of new combustion engines such as the HCCI (Homogeneous Charge Compression Ignition) engine [Lu et al., 2011] offers higher compression ratios with lower combustion temperature by relying on very lean premixed combustion. This leads to higher engine efficiency and lower pollutant emissions but these gains come with unpredictable ignition timings and less stable flames.

Concerning air transporation, the enhancement of combustion efficiency and emission is also possible through leaner combustion [Tacina et al., 2004]. However in-flight relight capabilities at high altitudes and combustion instabilities are of major concern in those regimes.

In ground power generation and industrial furnaces, flameless combustion regimes [Cavaliere and de Joannon, 2004] are investigated and seem to be a promising technique to adress the efficiency and emission challenges of combustion. This technology also relies on ultra-lean conditions with a low temperature reaction zone.

1.2 Plasma assisted combustion: a new way to enhance combustion

Plasmas, the fourth state of matter, have shown capabilities of enhancing radical production as well as decreasing the ignition delay. Therefore plasma assisted combustion might be the solution to these combustion challenges as shown in the reviews of Ju and Sun [2015], Starikovskiy and Aleksandrov [2013].

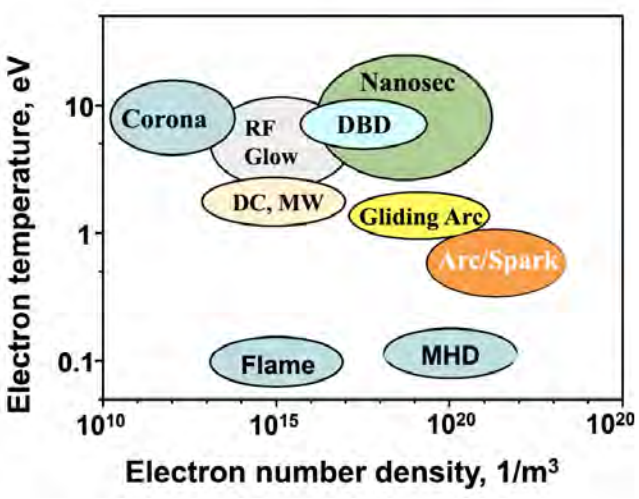


Figure 1.1: Plasma discharges overview taken from Ju and Sun [2015].

There is a wide variety of plasma discharges shown in Fig. 1.1 which can enhance combustion in three ways summarized in Fig. 1.2: thermally, kinetically and with a transport effect. Plasma discharges are controled by the reduced electric field E/N where E is the electric field and N the neutral gas density. This reduced electric field in turn controls the electron temperature and the electron density which are the two main parameters that discriminate different discharges.

An important notion to define is equilibrium for plasma discharges. To create a plasma discharge, a high voltage difference is imposed between two electrodes which accelerate electrons (more details in Chap. 2). At the onset of the discharge, electron temperature T_e deviates substantialy from the gas temperature T_g. As long as $T_e \gg T_g$ the plasma is a thermally non-equilibrium one with respect to the gas. If the plasma is maintained for a certain amount of time, electron temperature and gas temperature equilibrate ($T_e \simeq T_g$) through elastic collisions between electrons and neutrals and we obtain a thermal equilibrium plasma. During this thermalization process, electrons are basically heating the gas and causing a temperature increase (left part of Fig. 1.2). The arc and spark discharges of Fig. 1.1 are equilibrium plasmas that only use the thermal effect of plasmas to impact combustion.

During the non-equilibrium phase of a plasma, the highly energetic electrons are able to excite vibrational and electronic quantum states of neutrals (central pathway of Fig. 1.2). The amounts of vibrational and electronic population created are far from the Boltzmann distribution given by the canonical ensemble so that we have achieved *statistical non-equilibrium* on top of *thermal non-equilibrium*. These excited species are highly reactive and are able to create radicals which can also be formed by electron-impact reactions on neutrals. There is not one way to keep non-equilibrium plasmas and all the highest electron temperature plasmas of Fig. 1.1 from Corona to Nanosec use different approaches that focus more on the kinetic pathway enhancement of Fig. 1.2.

Finally charged species are subject to electric and magnetic forces and a transport effect can be achieved by plasmas (right pathway of Fig. 1.2). Accelerated charged species

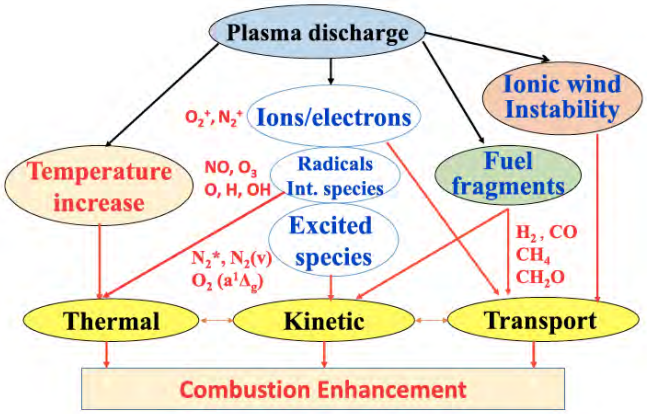


Figure 1.2: The schematic of major enhancement pathways of plasma assisted combustion taken from Ju and Sun [2015].

can breakdown the large fuel molecules to smaller fragments thus changing the combustion process. Alternatively, the ionic wind produced by charged species can also alter the flow velocity.

In this work, plasma discharges of small duration are used to obtain non-equilibrium plasma discharges: the Nanosecond Repetitively Pulsed (NRP) discharges. Since Kruger et al. [2002] showed that NRPs were able to maintain a non-equilibrium plasma state at atmospheric pressures, multiple experimental and numerical studies of NRP discharges have been performed.

Barbosa et al. [2015] showed the benefits of using such discharges on premixed swirl burners, representative of real industrial burners. In this experiment, NRP discharges extended the burner lean extinction limit and produced robust and stable flames in ultra-lean conditions. Another study in Xu et al. [2016] compared NRP discharges against conventional spark discharges and showed faster kernel expansion using the former.

Numerical studies of plasma discharges have been performed in Celestin [2008], Tholin [2012] using simplified chemistry in air and H_2-O_2 mixtures. A complete methodology to simulate NRP discharges has been laid out in these works to assess the influence of gap distance, radius of curvature of electrodes, applied voltage on the discharge propagation. To take into account NRPs in combustion simulations, a phenomonological model has been developed in Gracio Bilro Castela [2016] using the three major pathways of Fig. 1.2. A simpler model has also been used in Maestro [2018] only accounting for thermal effects.

1.3 Thesis objectives and outline

This thesis has focused on the detailed numerical simulation of plasma assited combustion. A previous code at CERFACS, AVIP, has been written for simulation of low-temperature plasmas in Hall-effect thrusters Joncquieres [2019]. A complete rewriting of AVIP to be able to simulate both Hall-effect thrusters plasmas and plasma discharges in the same code has been carried out.

The first part of this thesis is dedicated to plasma assisted combustion modelling. Basics of plasma physics in Chap. 2 are presented where important plasma phenomena are detailed. We then focus on turbulent combustion equations with the reactive Navier-Stokes equations in Chap. 3. Descriptions of premixed, non-premixed in laminar and turbulent conditions are given with the most important thermodynamic relations in gas mixtures. Finally Chap. 4 presents the plasma assisted combustion set of equations considered for the numerical simulation of NRP discharges in air and methane-air mixtures.

The numerical methods implemented in AVIP to simulate plasma assisted combustion are described in a second part. General definitions and properties of numerical schemes as well as the data structure of AVIP are given in Chap. 5. The Poisson equation is the subject of Chap. 6 where, besides presenting the AVIP numerical discretization, general properties and canonical solutions are given. Plasma species are described with drift-diffusion equations in AVIP which consist in advection-diffusion-reaction equations and their numerical integration is detailed in Chap. 7. Robust numerical schemes for gas mixture equations have been developed in this work for unstructured meshes and are presented in Chap. 8.

Numerical simulations of plasma assisted combustion are performed in the third part. Zero-dimensional computations are first presented with the creation of a new chemistry compiled from all across the litterature where validation against real cases is performed. The capability of AVIP to simulate plasma discharges is then carried out with a comparison to a streamers code benchmark as well as other works. The simulation of multi-dimensional plasma assisted combustion with reduced chemistry is finally performed on several cases in air and methane-air mixtures. These simulations fully couple reduced non-equilibrium plasma and combustion chemistries. Knowledge gained by these simulations allow to extend and precise the phenomological model of Gracio Bilro Castela [2016] and is presented in a last part.

Finally, work related to plasma simulations and neural networks is presented: the resolution of the Poisson equation with neural networks is attempted for plasma simulations. Since this topic is not directly related to the main body of this thesis, it can be read independenty and some redundances with the previous parts are present. The basics of deep learning are first recalled where the central backpropagation algorithm is detailed. Then a study of using convolutional neural networks to solve the Poisson and screened Poisson equations is performed.

Part I

Modeling Plasmas and Combustion

Chapter 2

Plasma physics

Contents

2.1 General properties of plasma

A general overview of plasma physics is given here and adapted from [Bittencourt, 2004, Chap. 1]. Simple definitions and examples of plasmas in nature are given with applications outside PAC.

2.1.1 The fourth state of matter

A plasma is a wide variety of macroscopically neutral substances containing many interacting free electrons and ionized atoms or molecules, which exhibit collective behavior due to the long range Coulomb forces.

It is the fourth state of matter after the solid, liquid and gaseous states. By heating a solid, liquid or gas, the atoms or melocules acquire more thermal kinetic energy until they are able to overcome the binding potential energy. At sufficiently elevated temperatures an increasing fraction of the atoms will possess enough energy to overcome, by collisions, the binding energy of the outermost orbital electrons, and an ionized gas or plasma results.

They are two main ways to produce plasma: either by photoionization *i.e.* by asborbing incident photons whose energy are equal to or greater than the ionization potential of the absorbing atom, or with a gas discharge, *i.e.* an electric field is applied across a gas which creates an avalanche of ionization.

The properties of a plasma depend on the type of particle interactions and its behavior display collective effects. In a plasma charge-charge and charge-neutral interactions are separated, they occur in respectively strongly ionized and weakly ionized plasmas.

Plasmas display a wide range of properties depending on the conditions in which they are created but they do have some common properties : they are good electrical/thermal conductors, for all of them the lower mass-electrons diffuse faster than ions generating a space charge and they all have an ability to sustain wave phenomena.

2.1.2 The occurence of plasmas in nature

With the progress made in astrophysics and in theoretical physics during the last century, it was realized that most of the matter in the known universe, with few exceptions such as the surface of cold planets (like the Earth), exists as plasma.

The *sun*, which is our nearest star and upon which the existence of life on Earth fundamentally depends, is a plasma phenomenon. Its energy output is derived from thermonuclear fusion reactions of protons forming helium ions deep in its interior and the consequent thermonuclear reactions keep the entire sun gaseous. Plasma erupted from the sun's surface can be seen in Fig. 2.1.

Figure 2.1: Plasma erupted from the sun, taken from https://www.livescience.com.

These eruptions produce a plasma called the *solar wind* which can travel great distances. Solar wind can disturb the Earth's magnetic field and produce the well-known polar lights shown in Fig. 2.2.

The Ionosphere, which is a part of the atmosphere which envelopes the Earth from an altitude of approximately 60 km to several thousands of kilometers, is another example of

Figure 2.2: Picture of Aurora, taken from https://en.wikipedia.org/wiki/Aurora.

plasma. The ionized particles in the ionosphere are produced during the daytime through absorption of solar extreme ultraviolet and X-ray radiation by the atmospheric species.

2.1.3 Application of plasma physics other than PAC

A wide variety of plasma experiments have been performed in laboratories to aid in the understanding of plasma, as well as to test and help expand plasma theory. The progress in plasma research has led to a wide range of plasma applications. Plasma assisted combustion is obviously one of them but this section highlights other ones.

The most important application of man-made plasmas is in the control of *thermonuclear fusion reactions*, which holds a vast potential for the generation of power. Nuclear fusions is the process whereby two light nuclei combine to form a heavier one, the total final mass being slightly less than the total initial mass. The mass difference (Δm) appears as energy (E) according to Einstein's famous law $E = (\Delta m)c^2$, where c denotes the speed of light. The nuclear fusion reaction is the source of energy in the stars, including the sun.

The *magnetohydrodynamic (MHD) energy generator* converts the kinetic energy of a dense plasma flowing across a magnetic field into electrical energy [Sherman, 1966]. MHD generators are different from traditional electric generators because they operate without moving parts to limit the upper temperature: the moving conductor is in this case a hot conductive ionized gas.

Plasma propulsion systems for satellites are based on a process that converts electrical energy into plasma kinetic energy, that is, the reverse of the MHD generator process. Among them, Hall-effect thrusters have received increased attention recently [Taccogna and Garrigues, 2019]: they generate thrust by accelerating charged species using the Hall effect. An example of Hall-effect thruster from Safran is shown in Fig. 2.3.

2.2 Governing equations

The plasma governing equations are detailed in this section. The Maxwell equations are first recalled to solve the electromagnetic fields. The particle orbit theory which is the basis of

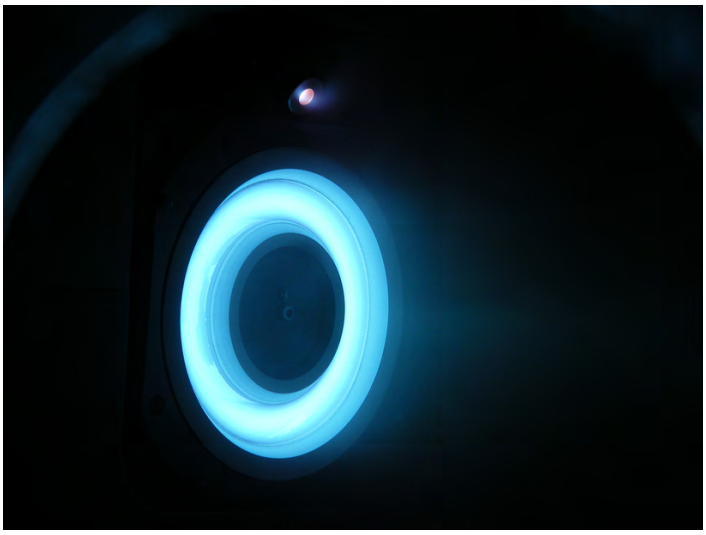


Figure 2.3: Picture of a PPS®5000, taken from https://www.safran-group.com.

PIC simulations is then briefly presented. The remaining part of this section is dedicated to the derivation of the plasma fluid equations from the Boltzmann equation and is adapted from [Raimbault, 2018, Chap. 2].

2.2.1 Charge phenomena

To model plasma we need the Maxwell equations to get the electric field $\mathbf{E}$ and the magnetic field $\mathbf{B}$. Those two fields appear in the momentum and energy equations of the fluid formulation of plasma physics. The Maxwell equations are as follows Jackson [1999]:

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{2.1}$$

$$\nabla \times \mathbf{B} = \mu_0(\mathbf{J} + \varepsilon_0 \frac{\partial \mathbf{E}}{\partial t}), \qquad \mathbf{J} = \frac{1}{\delta V} \sum_i q_i \mathbf{v}_i \tag{2.2}$$

$$\nabla \cdot \mathbf{E} = \frac{\rho_q}{\varepsilon_0}, \qquad \rho_q = \frac{1}{\delta V} \sum_i q_i \tag{2.3}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{2.4}$$

where q_i and $\mathbf{v}_i$ are respectively the charge and speed of each charged particle, δV is a suitably chosen small volument element, ρ_q is the global charge density, $\mathbf{J}$ is the global current density, ε_0 is the vacuum permitivity and μ_0 is the vacuum permeability.

2.2.2 Particle orbit theory

Each charged particle of plasma is treated individually and for each one Newton's second law is solved. In the case of uniform electrostatic, magnetostatic field and a uniform and

constant external force $\mathbf{F}_{ext}$ (*e.g.* the gravity):

$$m\frac{d\mathbf{v}}{dt} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B}) + \mathbf{F}_{ext} \tag{2.5}$$

where m is the particle mass,q is the particle charge, $\mathbf{v}$ is the speed. The particle velocity can be expressed as follows:

$$\mathbf{v}(t) = \mathbf{\Omega} \times \mathbf{r}_c + \frac{\mathbf{E}_\perp \times \mathbf{B}}{B^2} + \frac{q\mathbf{E}_\parallel}{m}t + \mathbf{v}_\parallel(0) + \frac{\mathbf{F}_{ext} \times \mathbf{B}}{qB^2} \tag{2.6}$$

where the parallel and perpendicular directions are defined in reference to the magnetic field $\mathbf{B}$, $\mathbf{\Omega} = -q\mathbf{B}/m$ is a vector whose norm is called the *cyclotron frequency*, the vector $\mathbf{r}_c$ is interpreted as the particle position vector with respect to the point G (the center of gyration) in the plane perpendicular to $\mathbf{B}$ which contains the particle.

The first term is associated to the cyclotron motion, the second term is called the plasma drift velocity, the third term is the uniform acceleration due to the electric field, the fourth one is the initial condition and the last one is the drift due to the external force.

2.2.3 Statistical approach

A plasma is a system containing a very large number of interacting charged particles that can be described using a statistical approach through the use of a distribution function f_α for each species α present in the plasma. The Boltzmann equation describing a system of N particles α in phase space ($\mathbf{r}$,$\mathbf{v}$,t) is :

$$\frac{\partial f_\alpha}{\partial t} + \mathbf{v} \cdot \nabla f_\alpha + \mathbf{a} \cdot \nabla_v f_\alpha = \left(\frac{\delta f_\alpha}{\delta t}\right)_c \tag{2.7}$$

where a is the total acceleration and $\nabla_v = \partial/\partial\mathbf{v}$ the gradient in velocity space.

The Vlasov equation is commonly used to model plasma. The collisions are neglected and it considers that the plasma particle motions are governed by the applied external fields plus the internal fields :

$$\frac{\partial f_\alpha}{\partial t} + \mathbf{v} \cdot \nabla f_\alpha + \frac{1}{m}\left[\mathbf{F}_{ext} + q(\mathbf{E}_i + \mathbf{v} \times \mathbf{B}_i)\right] \cdot \nabla_v f_\alpha = 0 \tag{2.8}$$

where $\mathbf{E}_i$ and $\mathbf{B}_i$ are *internal* smoothed electric and magnetic fields due to the presence and motion of all charged particles inside the plasma.

2.2.4 From Boltzmann to a fluid model

From the Boltzmann equation, the number of particles per unit volume, the number density, is defined as :

$$n_\alpha = \int_v f_\alpha(\mathbf{r}, \mathbf{v}, t)\mathrm{d}^3\mathbf{v} \tag{2.9}$$

This relation shows that f/n defines a normalized probability density function that will be noted F in the following. Let a($\mathbf{r}$,$\mathbf{v}$,t) be a molecular property defined in the phase space.

The average value of a is then:

$$\langle a \rangle(\mathbf{r}, t) = \frac{1}{n_\alpha(\mathbf{r}, t)} \int_v a(\mathbf{r}, \mathbf{v}, t) f_\alpha(\mathbf{r}, \mathbf{v}, t) \mathrm{d}^3 v \tag{2.10}$$

where n_α is the number density. This average value gives access to all macroscopic variables. Let us insist on the fact that $\langle a \rangle(\mathbf{r}, t)$ is a function of $\mathbf{r}$ and t, the integration over velocity space allows to define molecular properties in the physical space of position and time.

From now on the α subscript is ommited for more clarity. The characteristic function of a probability density function is its Fourier transform :

$$\hat{F}(\mathbf{k}) = \int e^{i\mathbf{k}\cdot\mathbf{v}} F(\mathbf{r}, \mathbf{v}, t) d\mathbf{v} = \langle e^{i\mathbf{k}\cdot\mathbf{v}} \rangle \tag{2.11}$$

By developing in series the exponential :

$$\langle e^{i\mathbf{k}\cdot\mathbf{v}} \rangle = 1 + i\langle \mathbf{k}\cdot\mathbf{v} \rangle + \frac{i^2}{2}\langle (\mathbf{k}\cdot\mathbf{v})^2 \rangle + \dots \tag{2.12}$$

Products such as $\langle v_x v_y v_z ... v_x \rangle$ appear in this development. These terms are associated to the concept of moments of a probability density function. More precisely, the kth moment of the distribution function F is :

$$M_k(\mathbf{r}, t) = \int \underbrace{\mathbf{v}\mathbf{v}\dots\mathbf{v}}_{\text{k times}} F d\mathbf{v} = \langle \mathbf{v}\mathbf{v}\dots\mathbf{v} \rangle \tag{2.13}$$

where the product used here is the tensor product. The information contained in the Fourier transform being tantamount to the information contained in the initial function it is clear that the knowledge of all the moments is equivalent to the knowledge of the initial function. We can also note that the higher the moment, the more information about the tail in velocity space of the probability density function we get.

To get the fluid equations the Boltzmann equation is then multiplied by $a(\mathbf{v})$ and integrated over velocity space. The general transport equation is finally derived:

$$\frac{\partial \langle n a(\mathbf{v}) \rangle}{\partial t} + \nabla_{\mathbf{r}} \cdot \langle n a(\mathbf{v}) \mathbf{v} \rangle = \langle \frac{\mathbf{F}}{m} \cdot \nabla_v n a(\mathbf{v}) \rangle + \int a(\mathbf{v}) \left(\frac{\delta f}{\delta t} \right)_c \mathrm{d}^3 \mathbf{v} \tag{2.14}$$

where $\nabla_{\mathbf{r}} = \partial / \partial \mathbf{r}$ is the gradient with respect to position and $\nabla_{\mathbf{v}}$ is the gradient with respect to velocity.

The following macroscopic properties are defined as follows :

- The mean velocity $\mathbf{V} = \langle \mathbf{v} \rangle$
- The peculiar velocity $\mathbf{w} = \mathbf{v} - \mathbf{V}$
- The kinetic pressure $\mathbf{P} = nm\langle \mathbf{w}\mathbf{w} \rangle$
- The temperature $T = \frac{k_b m}{3} \langle w^2 \rangle$
- The heat flux $\mathbf{q} = \frac{nm}{2} \langle w^2 \mathbf{w} \rangle$

where $v = |\mathbf{v}|$ and $w = |\mathbf{w}|$. Setting $a = 1, m\mathbf{v}, mv^2/2$, we retrieve the continuity, momentum and energy equations:

$$\frac{\partial n}{\partial t} + \nabla \cdot \mathbf{\Gamma} = \int \left(\frac{\delta f}{\delta t}\right)_c \mathrm{d}^3\mathbf{v} \tag{2.15}$$

$$\frac{\partial nm\mathbf{V}}{\partial t} + \nabla \cdot \mathbf{D} = nq(\mathbf{E} + \mathbf{V} \times \mathbf{B}) + \int m\mathbf{v} \left(\frac{\delta f}{\delta t}\right)_c \mathrm{d}^3\mathbf{v} \tag{2.16}$$

$$\frac{\partial n\varepsilon}{\partial t} + \nabla \cdot \mathbf{H} = nq\mathbf{E} \cdot \mathbf{V} + \int \frac{1}{2} mv^2 \left(\frac{\delta f}{\delta t}\right)_c \mathrm{d}^3\mathbf{v} \tag{2.17}$$

with the total energy $\varepsilon = \langle \frac{1}{2} mv^2 \rangle = \frac{1}{2} mV^2 + \frac{3}{2} k_B T$, the particle flux $\mathbf{\Gamma} = n\mathbf{V}$, the momentum flux $\mathbf{D} = \mathbf{P} + nm\mathbf{V}\mathbf{V}$ and the total energy flux $\mathbf{H} = n\varepsilon\mathbf{V} + \mathbf{q} + \mathbf{P} \cdot \mathbf{V}$.

2.2.5 The Boltzmann model for collisions

2.2.5.1 Collision parameters

Let us consider two species α and β with their respective distribution function f_α and f_β. $f_\alpha(\mathbf{r}, \mathbf{v}_\alpha, t)\mathrm{d}^3\mathbf{v}_\alpha$ is the number of particles per unit volume with speeds between $\mathbf{v}_\alpha$ and $\mathbf{v}_\alpha + \mathrm{d}^3\mathbf{v}_\alpha$ and $f_\beta(\mathbf{r}, \mathbf{v}_\beta, t)\mathrm{d}^3\mathbf{v}_\beta$ is the number of particles per unit volume with speeds between $\mathbf{v}_\beta$ and $\mathbf{v}_\beta + \mathrm{d}^3\mathbf{v}_\beta$.

The number of collisions, $\mathrm{d}Q_{\alpha\beta}$, between α and β particles per unit volume and per unit time, having speeds equal to $\mathbf{v}_\alpha$ (with $\mathrm{d}^3\mathbf{v}_\alpha$ uncertainty) and $\mathbf{v}_\beta$ (with $\mathrm{d}^3\mathbf{v}_\beta$ uncertainty), causing diffusion in the elementary solid angle $\mathrm{d}\Omega$ is:

$$\mathrm{d}Q_{\alpha\beta} = f_\alpha(\mathbf{v}_\alpha)\mathrm{d}^3\mathbf{v}_\alpha f_\beta(\mathbf{v}_\beta)\mathrm{d}^3\mathbf{v}_\beta v_{\alpha\beta}\sigma_{\alpha\beta}(v_{\alpha\beta}, \Omega)\mathrm{d}\Omega \tag{2.18}$$

This relation defines the differential cross section $\sigma_{\alpha\beta}$. A cross section must be associated with each type of collision (elastic collisions, inelastic collisions including ionization, excitation, attachment). In the case of inelastic collisions, a threshold effect is observed: cross section are equal to zero below a certain value. The calculation of these cross sections is in general very difficult and sometimes need to include quantum effects.

The plasma fluid equations are invariant in space and time and it can be thus shown that cross sections are equal for a reaction and its inverse

$$\sigma_{\alpha\beta\to\alpha'\beta'} = \sigma_{\alpha'\beta'\to\alpha\beta} \tag{2.19}$$

This relation is known as the *microreversibility* property. The *total cross section* $\Sigma_{\alpha\beta}(v)$ is also defined where information on angles is lost:

$$\Sigma_{\alpha\beta}(v_{\alpha\beta}) \equiv \int \sigma_{\alpha\beta}(v_{\alpha\beta}, \Omega)\mathrm{d}\Omega \tag{2.20}$$

Three important quantities depending on the total cross section are also *the collision rate* $K_{\alpha\beta}$, *the collision frequency* $\nu_{\alpha\beta}$ and *the mean free path* $\lambda_{\alpha\beta}$:

$$K_{\alpha\beta} \equiv \Sigma_{\alpha\beta}(v_{\alpha\beta})v_{\alpha\beta} \tag{2.21}$$

$$\lambda(v_{\alpha\beta}) \equiv \frac{1}{n_\beta \Sigma_{\alpha\beta}(v_{\alpha\beta})} \tag{2.22}$$

$$\nu_{\alpha\beta} \equiv n_\beta K_{\alpha\beta} = n_\beta \Sigma_{\alpha\beta} v_{\alpha\beta} = \frac{v_{\alpha\beta}}{\lambda_{\alpha\beta}(v_{\alpha\beta})} \tag{2.23}$$

The corresponding macroscopic physical quantities can be defined with an averaging using the distribution functions.

Some of the important interactions between plasma particles can be analyzed using a central force law:

$$F_{\alpha\beta} = \frac{C_{\alpha\beta}}{r_{\alpha\beta}^a} \tag{2.24}$$

where $r_{\alpha\beta}$ is the distance between the particles, and $C_{\alpha\beta}$ and a are constants which depend on the interaction. The important cases are $a = 2$, $a = 5$ and $a \to \infty$. In what follows, the functional dependence on the relative speed of each parameters described above is recalled.

1. $a = 2$: Coulomb interaction. This case describes interactions between charged particles: electron-electron, electron-ion, ion-ion.

$$F_{\alpha\beta}(r_{\alpha\beta}) \propto \frac{1}{r_{\alpha\beta}^2} \implies \sigma_{\alpha\beta}(v_{\alpha\beta}) \propto \frac{1}{v_{\alpha\beta}^4} \implies K_{\alpha\beta}, \nu_{\alpha\beta} \propto \frac{1}{v_{\alpha\beta}^3} \tag{2.25}$$

2. $a = 5$: charge-induced dipole interaction. This case corresponds to the interaction between a charged particle and a polarizable atom. The polarization occurs only if the relative speed between the two particles is not too big otherwise it is simply a hard spheres interaction. This interaction can also be used to describe ion-corresponding atom interaction.

$$F_{\alpha\beta}(r_{\alpha\beta}) \propto \frac{1}{r_{\alpha\beta}^5} \implies \sigma_{\alpha\beta}(v_{\alpha\beta}) \propto \frac{1}{v_{\alpha\beta}} \implies K_{\alpha\beta}, \nu_{\alpha\beta} = \text{Cte} \tag{2.26}$$

3. $a \to \infty$: hard spheres interaction. This interaction can be used for an interaction between a charged particle and a non-polarizable atom. It can also be used to describe short range atom-atom interaction (at longer range the van der Waals potential is needed).

$$\text{Hard spheres} \implies \sigma_{\alpha\beta}(v_{\alpha\beta}) = \text{Cte} \implies K_{\alpha\beta}, \nu_{\alpha\beta} \propto v_{\alpha\beta} \tag{2.27}$$

The last two interaction types where either the collision frequency or the mean free-path is independent of the relative speed of the particles are two limit models that allow simplifications in the case of weakly-ionized plasmas.

2.2.5.2 The Boltzmann collision integral

In this section, one derivation of the collision term $(\delta f_\alpha/\delta t)_c$ in the Boltzmann equation is presented. It was originally obtained by Boltzmann [1872] and applicable to low density neutral gases.

In the weakly ionized plasmas of interest, the dominant interactions are the ones between charged and neutral particles. Contrary to the long range Coulomb interactions, Boltzmann integral can be applied to these interactions.

The integral collision $(\delta f_\alpha/\delta t)_c$ represents the variation of the number of particles per phase space unit volume $\mathrm{d}^3\mathbf{r}\mathrm{d}^3\mathbf{v}$ and per unit time of α particles as a result of collisions with β particles (typically electron-neutral, ion-neutral or neutral-neutral).

The hypotheses made in this calculation are as follows:

- collisions are binary (diluted gas)
- collisions are local and instantaneous
- speeds of particles before collisions are uncorrelated

Variations of the number of particles α in the $\mathrm{d}^3\mathbf{r}\mathrm{d}^3\mathbf{v}$ phase space volume during dt due to collisions with β particles have two origins:

- a loss of particles having speed $\mathbf{v}_\alpha$ before collision and $\mathbf{v}'_\alpha$ after collision. From the definition of the differential cross section:

$$\mathrm{d}N_\alpha^- = \mathrm{d}^3\mathbf{r}dt \int_{\mathbf{v}_\beta}\int_\Omega v_{\alpha\beta}\sigma_{\alpha\beta}(v_{\alpha\beta}) f_\alpha(\mathbf{r},\mathbf{v}_\alpha,t) f_\beta(\mathbf{r},\mathbf{v}_\beta,t)\mathrm{d}\Omega\mathrm{d}^3\mathbf{v}_\alpha\mathrm{d}^3\mathbf{v}_\beta \tag{2.28}$$

- a gain of particles having speed $\mathbf{v}'_\alpha$ before collision and $\mathbf{v}_\alpha$ after collision. From the definition of the differential cross section:

$$\mathrm{d}N_\alpha^+ = \mathrm{d}^3\mathbf{r}dt \int_{\mathbf{v}_{\beta'}}\int_\Omega v'_{\alpha\beta}\sigma'_{\alpha\beta}(v'_{\alpha\beta}) f_\alpha(\mathbf{r},\mathbf{v}'_\alpha,t) f_\beta(\mathbf{r},\mathbf{v}'_\beta,t)\mathrm{d}\Omega\mathrm{d}^3\mathbf{v}'_\alpha\mathrm{d}^3\mathbf{v}'_\beta \tag{2.29}$$

  where $\sigma'_{\alpha\beta}(v'_{\alpha\beta}) \equiv \sigma_{\alpha'\beta'\to\alpha\beta}(v'_{\alpha\beta},\Omega)$.

Since:

$$\left(\frac{\delta f_\alpha}{\delta t}\right)_c^B \mathrm{d}^3\mathbf{v}_\alpha\mathrm{d}^3\mathbf{r}dt = \mathrm{d}N_\alpha^+ - \mathrm{d}N_\alpha^-, \tag{2.30}$$

the general expression for the integral collision is:

$$\begin{aligned}\left(\frac{\delta f_\alpha}{\delta t}\right)_c^B \mathrm{d}^3\mathbf{v}_\alpha = &\int_{\mathbf{v}_{\beta'}}\int_\Omega v'_{\alpha\beta}\sigma_{\alpha\beta}(v'_{\alpha\beta}) f_\alpha(\mathbf{r},\mathbf{v}'_\alpha,t) f_\beta(\mathbf{r},\mathbf{v}'_\beta,t)\mathrm{d}\Omega\mathrm{d}^3\mathbf{v}'_\alpha\mathrm{d}^3\mathbf{v}'_\beta \\ &- \int_{\mathbf{v}_\beta}\int_\Omega v_{\alpha\beta}\sigma_{\alpha\beta}(v_{\alpha\beta}) f_\alpha(\mathbf{r},\mathbf{v}_\alpha,t) f_\beta(\mathbf{r},\mathbf{v}_\beta,t)\mathrm{d}\Omega\mathrm{d}^3\mathbf{v}_\alpha\mathrm{d}^3\mathbf{v}_\beta\end{aligned} \tag{2.31}$$

where only the micro-reversibility property was used *i.e.* $\sigma_{\alpha\beta\to\alpha'\beta'} = \sigma_{\alpha'\beta'\to\alpha\beta}$.

2.2.5.3 Contributions of elastic collisions on plasma fluid equations

To obtain the fluid model from Boltzmann's equation an integration on velocity space is needed, therefore to evaluate the impact of elastic collisions on the fluid equations an evaluation of the quantity

$$S_{l\alpha}^{\mathrm{el}} = \int a(\mathbf{v}_\alpha) \left(\frac{\delta f_\alpha}{\delta t}\right)_c^{\mathrm{el}} \mathrm{d}^3\mathbf{v}_\alpha \tag{2.32}$$

where $a(\mathbf{v}_\alpha)$ represents the considered moment considered and l its corresponding number ($l = 0, 1, 2$ in the derivation of the fluid equations presented here). By using specific properties of elastic and binary collisions $v'_{\alpha\beta} = v_{\alpha\beta}$, $\mathrm{d}^3\mathbf{v}'_\beta \mathrm{d}^3\mathbf{v}'_\alpha = \mathrm{d}^3\mathbf{v}_\beta \mathrm{d}^3\mathbf{v}_\alpha$ and $\sigma_{\alpha\beta\to\alpha'\beta'} = \sigma_{\alpha'\beta'\to\alpha\beta}$, this term is reduced to:

$$S_{l\alpha}^{\mathrm{el}} = \iiint [a(\mathbf{v}'_\alpha) - a(\mathbf{v}_\alpha)] v_{\alpha\beta}\sigma_{\alpha\beta}(v_{\alpha\beta}) f_\alpha f_\beta \mathrm{d}\Omega \mathrm{d}^3\mathbf{v}_\alpha \mathrm{d}^3\mathbf{v}_\beta \tag{2.33}$$

One interesting thing to notice about this expression is that when $\beta = \alpha$ *i.e.* the collisions are between identical particles, it yields $S_\alpha^{\mathrm{el}} = 0$. Collisions between identical particles do not influence the plasma fluid equations. However in plasmas there are multiples species to ensure a global neutrality and the collisions contribute to the fluid equations.

1. *Continuity equation* From $a(\mathbf{v}_\alpha) \equiv 1$:

$$S_{0\alpha}^{\mathrm{el}} = 0 \tag{2.34}$$

2. *Momentum equation* $a(\mathbf{v}_\alpha) \equiv m_\alpha \mathbf{v}_\alpha$ is taken here. In the case of elastic collisions the variation of momemtum is:

$$m_\alpha \mathbf{v}'_\alpha - m_\alpha \mathbf{v}_\alpha = \mu_{\alpha\beta}(\mathbf{v}'_{\alpha\beta} - \mathbf{v}_{\alpha\beta}) = \mu_{\alpha\beta}[-(1-\cos\theta)\mathbf{v}_{\alpha\beta} + v_{\alpha\beta}\sin\theta\mathbf{e}_\perp] \tag{2.35}$$

   By introducing this new expression in the integral that defines $\mathbf{S}_{1\alpha}^{\mathrm{el}}$, it yields:

$$\mathbf{S}_{1\alpha}^{\mathrm{el}} = -\mu_{\alpha\beta} n_\alpha n_\beta \iint \mathbf{v}_{\alpha\beta} K_{\alpha\beta} \frac{f_\alpha}{n_\alpha}\frac{f_\beta}{n_\beta} \mathrm{d}^3\mathbf{v}_\alpha \mathrm{d}^3\mathbf{v}_\beta \tag{2.36}$$

   where the momentum transfer cross section $\sigma_{\alpha\beta}^t$, the collision frequency $\nu_{\alpha\beta}$, the reaction rate $K_{\alpha\beta}$ and the reduced mass $\mu_{\alpha\beta}$ are defined by the following relations:

$$\sigma_{\alpha\beta}^t \equiv \int (1-\cos\theta)\sigma_{\alpha\beta}\mathrm{d}\Omega \tag{2.37}$$

$$\nu_{\alpha\beta} \equiv n_\beta \sigma_{\alpha\beta}^t v_{\alpha\beta} \tag{2.38}$$

$$K_{\alpha\beta} \equiv \sigma_{\alpha\beta}^t v_{\alpha\beta} \tag{2.39}$$

$$\mu_{\alpha\beta} \equiv \frac{m_\alpha m_\beta}{m_\alpha + m_\beta} \tag{2.40}$$

3. *Energy equation* By setting $a(\mathbf{v}_\alpha) \equiv m_\alpha v_\alpha^2/2$ and using the expression of the energy transfer in a binary collision:

$$S_{2\alpha}^{\mathrm{el}} \equiv \int \frac{1}{2} m_\alpha v_\alpha^2 \left(\frac{\delta f_\alpha}{\delta t}\right)_c^B \mathrm{d}^3\mathbf{v}_\alpha \tag{2.41}$$

$$= -\kappa_{\alpha\beta} n_\alpha n_\beta \left\langle \left(\frac{1}{2} m_\alpha v_\alpha^2 - \frac{1}{2} m_\beta v_\beta^2 + \frac{m_\beta - m_\alpha}{2} \mathbf{v}_\alpha \cdot \mathbf{v}_\beta\right) K_{\alpha\beta} \right\rangle \tag{2.42}$$

where $\langle \dots \rangle$ indicates an integration with the two normalized distribution functions and $\kappa_{\alpha\beta}$ is the energy transfer coefficient defined by:

$$\kappa_{\alpha\beta} \equiv \frac{\mu_{\alpha\beta}}{m_\alpha m_\beta} \tag{2.43}$$

To go further into the computation of those terms, the dependence of the reaction rate $K_{\alpha\beta}$ on the relative speed $v_{\alpha\beta}$ is needed. This is equivalent to specify the collision cross section and needs to be done case by case.

2.2.6 Drift-diffusion model

An approximation of the plasma fluid equations is often made when simulating plasmas leading to the so-called drift-diffusion model. When the mean free path of the charged particles (electrons and ions) is small compared to the characteristic lengths of the plasma, the species undergo lots of collision before any significant acceleration. In these conditions, the inertia forces can be neglected compared to the other forces:

$$-\nabla p_\alpha + n_\alpha q_\alpha (\mathbf{E} + \mathbf{V}_\alpha \times \mathbf{B}) - m_\alpha n_\alpha \sum_{\beta \neq \alpha} \nu_{\alpha\beta} (\mathbf{V}_\alpha - \mathbf{V}_\beta) \simeq 0 \tag{2.44}$$

In the case of weakly ionized plasmas, dominant collisions are the ion-neutral and electron-neutral ones. Therefore the only species β in the equation above are the neutral species. Being insensitive to the electromagnetic forces, the neutral species velocity can be considered small compared to the charged species *i.e.* $\mathbf{V}_\beta \ll \mathbf{V}_\alpha$. The force resulting from collisions with neutral particles is given by:

$$-m_\alpha n_\alpha n_\beta \sum_{\alpha\beta} \nu_{\alpha\beta} (\mathbf{V}_\alpha - \mathbf{V}_\beta) \approx -m_\alpha n_\alpha \nu_\alpha \mathbf{V}_\alpha \tag{2.45}$$

where $\nu_{\alpha n} \equiv \nu_\alpha$ since the only collisions considered here are the one with neutral particles. Finally Eq. (2.44) yields:

$$\mathbf{V}_\alpha = -\frac{k_B T_\alpha}{m_\alpha \nu_\alpha} \frac{\nabla p_\alpha}{p_\alpha} + \frac{q_\alpha}{m_\alpha \nu_\alpha} (\mathbf{E} + \mathbf{V}_\alpha \times \mathbf{B}) \tag{2.46}$$

2.2.6.1 Non-magnetized collisional plasmas

In the case where there is no magnetic field $\mathbf{B} = 0$, and the classic drift-diffusion model is retrieved:

$$\mathbf{V}_\alpha = \mu_\alpha \mathbf{E} - D_\alpha \frac{\nabla p_\alpha}{p_\alpha} \tag{2.47}$$

where μ_α and D_α are transport coefficients, respectively called *mobility* and *diffusion coefficient*:

$$\mu_\alpha \equiv \frac{q_\alpha}{m_\alpha \nu_\alpha} \qquad D_\alpha \equiv \frac{k_B T_\alpha}{m_\alpha \nu_\alpha} \tag{2.48}$$

and these two coefficients are related through the *Einstein relation*:

$$\frac{D_\alpha}{\mu_\alpha} = \frac{k_B T_\alpha}{q_\alpha} \tag{2.49}$$

The drift-diffusion equation clearly shows that the origin of the fluid speed for electrons or ions is the existence of gradients in the plasma (such as potential, density or temperature). It can also be noted that the pressure gradient contains in the general case two contributions from the temperature gradient and the density gradient:

$$\frac{\nabla p_\alpha}{p_\alpha} = \frac{\nabla T_\alpha}{T_\alpha} + \frac{\nabla n_\alpha}{n_\alpha} \tag{2.50}$$

In most drift-diffusion models the temperature gradient is considered negligible and only the density gradient is kept.

2.2.6.2 Magnetized collisional plasmas

When the magnetic field is not equal to zero, the fluid speed is not anymore given explictly by Eq. (2.46) and writes:

$$\mathbf{V}_\alpha = \mu_\alpha(\mathbf{E} + \mathbf{V}_\alpha \times \mathbf{B}) - D_\alpha \frac{\nabla p_\alpha}{p_\alpha} \tag{2.51}$$

However this vector equation is linear for the speed and can be solved through some vector manipulation. The demonstration can be found in [Raimbault, 2018, p. 61-62] and the speed can be divided into three components:

$$\mathbf{V} \equiv \mathbf{V}_\parallel + \mathbf{V}_\perp + \mathbf{V}_\times \tag{2.52}$$

$\parallel$ referring to the direction of the magnetic field, $\perp$ to the direction of the electic field perpendicular to the magnetic field and $\times$ to the third direction with respect to the first two and that forms a direct base. The expressions of these speeds are as follows:

$$\mathbf{V}_\parallel = \mu_\parallel \mathbf{E}_\parallel - D_\parallel \frac{\nabla_\parallel p_\alpha}{p_\alpha} \tag{2.53}$$

$$\mathbf{V}_\perp = \mu_\perp \mathbf{E}_\perp - D_\perp \frac{\nabla_\perp p_\alpha}{p_\alpha} \tag{2.54}$$

$$\mathbf{V}_\times = \mu_\times \mathbf{E}_\perp \times \mathbf{b} - D_\times \frac{\nabla_\perp p_\alpha}{p_\alpha} \times \mathbf{b} \tag{2.55}$$

$$\tag{2.56}$$

where $\mathbf{b}$ is the normalized vector giving the direction of the magnetic field: $\mathbf{b} \equiv \mathbf{B}/||\mathbf{B}||$. The transport coefficients are given by:

$$\frac{\mu_\parallel}{\mu} = \frac{D_\parallel}{D} = 1, \quad \frac{\mu_\perp}{\mu} = \frac{D_\perp}{D} = \frac{1}{1+(\omega_c/\nu)^2}, \quad \frac{\mu_\times}{\mu} = \frac{D_\times}{D} = \frac{\omega_c/\nu}{1+(\omega_c/\nu)^2} \tag{2.57}$$

where $\omega_c \equiv qB/m$ is the cyclotron frequency. In the directions parallel and perpendicular, the solutions with the presence of the magnetic field are formally the same as without any magnetic field but with different transport coefficients.

2.3 Important plasma phenomena

Three important plasma phenomena covering different aspects of plasma physics are presented in this section taken from [Bittencourt, 2004, Chap. 11]. The canonical plasma oscillation is first developed: it is the simplest interaction of plasma species and electromagnetic fields. The Debye shielding and effective electric field felt by an individual particle is then discussed. Finally basics of plasma sheath are recalled.

2.3.1 Electron-plasma oscillation

One of the fundamental properties of plasma is its tendency to maintain electric charge neutrality on a macroscopic scale under equilibrium conditions. When this macroscopic charge neutrality is disturbed, large Coulomb forces come into play and tend to restore the macroscopic charge neutrality. Since these Coulomb forces cannot be naturally sustained in the plasma, it breaks into high-frequency electron plasma oscillations *i.e.* the large Coulomb forces disappear but they trigger the plasma oscillations. These oscillations enable the plasma to maintain on average its electrical neutrality.

The simplest closed system of macroscopic transport equations that can be formed is known as the *cold plasma model*. This simple model encompasses only the equations of conservation of mass and of momentum. The highest moment of the distribution function, appearing in the momentum equation, is the kinetic pressure dyad, which, in this model, is taken equal to zero. It yields the following equations :

$$\frac{\partial n}{\partial t} + \nabla \cdot \mathbf{\Gamma} = S \tag{2.58}$$

$$\rho_m \frac{D}{Dt}\mathbf{u} = nq(\mathbf{E} + \mathbf{u} \times \mathbf{B}) + \mathbf{A} - \mathbf{u}S \tag{2.59}$$

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \tag{2.60}$$

where ρ_m is the mass density, ρ is the charge density, $\mathbf{u}$ the averaged velocity over phase space, q the charge, n the number density, E and B the electric and magnetic fields. $\mathbf{A}$ is the collision term in the momentum conservation equation (integral term in (2.16)) and S is the source term due to collision in the mass conservation (integral term in (2.15)).

To study the characteristics of the electron plasma oscillations we can use the cold plasma model, in which the particle thermal motion and the pressure gradient force are not taken into account as well as the source term. That way only the electric field influence is taken

into account. We shall neglect ion motion (since its mass is way larger than that of the electron) and assume a very small electron density perturbation such that :

$$n_e(\mathbf{r},t) = n_0 + n'_e(\mathbf{r},t) \tag{2.61}$$

where n_0 is a constant number density and $|n'_e| \ll n_0$. Similarly we assume that the electric field $\mathbf{E}(\mathbf{r},t)$ and the velocity $\mathbf{u}_e$ produced are first-order perturbations depending only on the position. Note that the electric field doesn't need to be one dimensional anymore and that the result is therefore more general in this case. The linearized continuity and momentum equations become :

$$\frac{\partial n'_e(\mathbf{r},t)}{\partial t} + n_0 \nabla \cdot \mathbf{u}_e(\mathbf{r},t) = 0 \tag{2.62}$$

$$\frac{\partial \mathbf{u}_e(\mathbf{r},t)}{\partial t} = -\frac{e}{m_e}\mathbf{E}(\mathbf{r},t) \tag{2.63}$$

where e is the elementary charge.

In the momentum equation we have assumed that the rate of momentum loss from the electron gas due to collisions $\mathbf{A}_\alpha$ is negligible. Considering singly charged ions, the charge density is given by

$$\rho(\mathbf{r},t) = -e[n_0 + n'_e(\mathbf{r},t)] + en_0 = -en'_e(\mathbf{r},t) \tag{2.64}$$

where the ion density is considered to be constant and uniform, and equal to n_0. Therefore the Poisson equation yields :

$$\nabla \cdot \mathbf{E} = -\frac{e}{\varepsilon_0} n'_e(\mathbf{r},t) \tag{2.65}$$

Eqs. (2.62), (2.63) and (2.65) constitute a complete set of equations to be solved for the variables $n'_e, \mathbf{u}_e$ and $\mathbf{E}$. Taking the divergence of Eq. (2.63), using Eq. (2.62) to substitute for $\nabla \cdot \mathbf{u}_e$ and Eq. (2.65) to eliminate $\nabla \cdot \mathbf{E}$ we obtain :

$$\frac{\partial^2 n'_e}{\partial t^2} + \omega_p^2 n'_e = 0 \tag{2.66}$$

$$\boxed{\omega_p = \sqrt{\frac{n_e e^2}{m_e \varepsilon_0}}} \tag{2.67}$$

Therefore the density varies harmonically in time at the electron plasma frequency ω_p. In fact, all first-order perturbations have a harmonic time variation at the plasma frequency ω_p. To justify this statement it is convenient to start with the assumption that all first-order quantities vary harmonically at the pulsation ω. Thus Eqs. (2.62) and (2.63) become :

$$n'_e = -\frac{i}{\omega} n_0 \nabla \cdot \mathbf{u}_e \tag{2.68}$$

$$\mathbf{u}_e = -\frac{ie}{\omega m_e} \mathbf{E} \tag{2.69}$$

which yields :

$$n'_e = -\frac{n_0 e}{\omega^2 m_e} \nabla \cdot \mathbf{E} \tag{2.70}$$

Substituting this expression for n'_e into Eq. (2.65) yields :

$$(1 - \omega_p^2/\omega^2) \nabla \cdot \mathbf{E} = 0 \tag{2.71}$$

which shows that a nontrivial solution requires $\omega = \omega_p$. Therefore all perturbations vary harmonically in time at the electron plasma frequency. These oscillations are stationary and longitudinal.

2.3.2 The Debye Shielding

The Debye length is a measure of the distance over which the influence of the electric field of an individual particle is felt by the other charged particles : there is a shielding of electrostatic fields in a plasma. This Debye length is :

$$\boxed{\lambda_D = \sqrt{\frac{\varepsilon_0 k_B T}{n_e e^2}}} \tag{2.72}$$

We can note than :

$$v_{th} = \sqrt{\frac{k_B T}{m_e}} = \lambda_D \times \omega_p \tag{2.73}$$

To illustrate this phenomenon, let's consider a plasma whose equilibrium state is perturbed by an electric field due to an external charged particle. We assume the test particle to have a positive charge +Q, and choose a spherical coordinate system whose origin coincides with the position of the test particle. We are interested in determining the electrostatic potential $\phi(\mathbf{r})$ that is established near the test charge Q due to the combined effects of the test charge and the distribution of charged particles. We assume one population of electrons and ions with opposite charges and in the far field $n_e(\infty) = n_i(\infty) = n_0$.

This is a steady state problem under a conservative electric field $\mathbf{E} = -\nabla\phi(\mathbf{r})$. It can be shown [Bittencourt, 2004, Chap. 7] that when perturbed by a conservative force, a Maxwellian distribution is perturbed in such a way that :

$$\begin{cases} n_e(\mathbf{r}) &= n_0 \exp\left(+\frac{e\phi}{k_B T}\right) \\ n_i(\mathbf{r}) &= n_0 \exp\left(-\frac{e\phi}{k_B T}\right) \end{cases}$$

From Maxwell Gauss' equation (2.3) one gets :

$$\nabla^2\phi(\mathbf{r}) - \frac{2}{\lambda_D^2}\phi(\mathbf{r}) = -\frac{Q}{\varepsilon_0}\delta(\mathbf{r}) \text{ with } e\phi(\mathbf{r}) \ll k_B T \tag{2.74}$$

The resolution of this equation yields :

$$\begin{cases} \phi(r) = \phi_c(r)\exp\left(-\frac{\sqrt{2}r}{\lambda_D}\right) \\ \phi_c(r) = \frac{Q}{4\pi\varepsilon_0} \end{cases}$$

ϕ_c is the potential created by the particle of charge Q in the absence of plasma. ϕ becomes much less than this ordinary Coulomb potential once r exceeds the distance λ_D called the Debye length.

2.3.3 The plasma sheath

When a material body is immersed in a plasma, the body acquires a net negative charge and therefore a negative potential with respect to the plasma potential. In the region near the wall of the body there is a boundary layer, known as the plasma sheath, in which the electron and the ion number densities are different, this boundary layer is of the order of the Debye length.

Let's consider a wall immersed in a plasma as depicted in Fig. 2.4. The random particle flux Γ_α going in the direction to the wall yields for a species α [Bittencourt, 2004, Chap. 7] :

$$\Gamma_\alpha = n_\alpha\sqrt{\frac{k_B T_\alpha}{2\pi m_\alpha}} \tag{2.75}$$

where n_α, T_α and m_α are the particle density, temperature and mass of species α.

Initially electron and ion densities are equal in the bulk of the plasma and the electron temperature is way higher than the ion temperature such that $\Gamma_e \gg \Gamma_i$. Therefore the wall in contact with the plasma accumulates negative charge which creates a negative potential at the wall. This negative potential repels the electrons and attracts the ions, so that the electron flux decreases and the ion flux increases. Eventually the negative potential becomes large enough in magnitude to equalize the rate at which electrons and ions hit the surface. At this floating negative potential the wall and the plasma reach a dynamical equilibrium such that the net current at the wall is zero.

2.4 Streamer Physics

Before reviewing gas discharge theory, it is necessary to define two important concepts: *gas discharge* and *electrical breakdown*. The term *discharge* is applied to any flow of electric current through ionized gas, and to any process of ionization of the gas by the applied electric field. In the most general sense the electrical breakdown is the process of transformation of a nonconducting material into a conductor as a result of applying to it a sufficiently strong field. If the applied field is strong and lasts long enough, a self-sustaining discharge, no matter of what kind, is ignited by breakdown. The mechanism of streamers described below is part of the conventional breakdown theory and is adapted from Liu [2006], Yuri P. Raizer [1991] and [Fridman, 2008, Chap. 4].

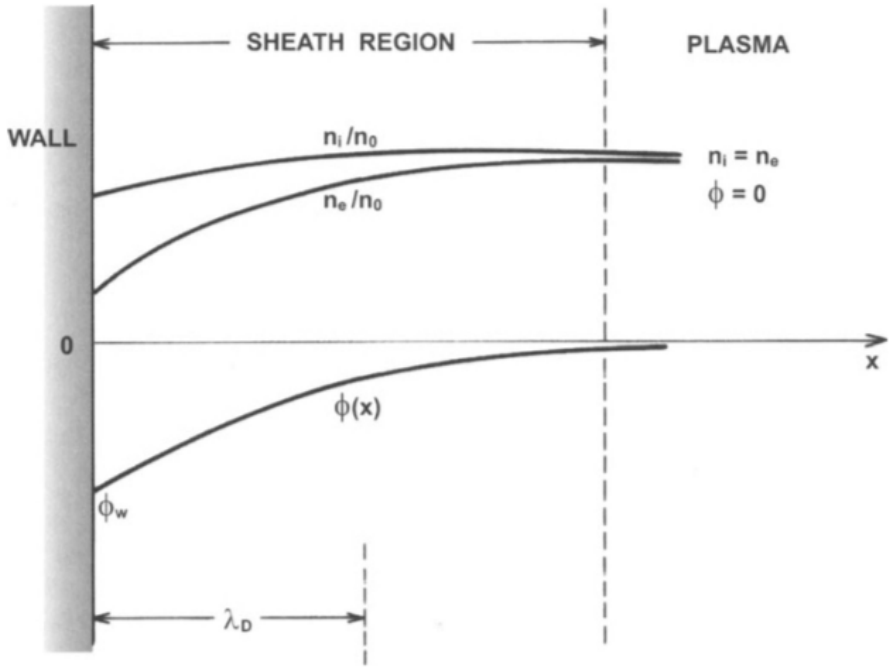


Figure 2.4: Diagram showing the variation of the eletrostatic potential $\phi(x)$ and the number densities $n_e(x)$ and $n_i(x)$ inside the plasma sheath near an infinite plane wall.

2.4.1 Electron avalanche

If an electron appears in the gap between two planar electrodes, which establish a uniform field, it will accelerate due to the electric field force and collide with the neutral molecules. If the electric field is strong enough, the energy gained by the electron between collisions will enable it to ionize the neutral molecules. As a result, there will be two electrons: the incident one (primary electron) and the new one (secondary electron). Those electrons can repeat the same collision ionization process, and the total number of electrons will exponentially increase. The increasing of the number of electrons acts as an avalanche, called an electron avalanche which is illustrated in Fig. 2.5.

The avalanche develops both in time and in space, and mathematically it can be described by the following equation:

$$\frac{\mathrm{d}n_e}{\mathrm{d}x} = \alpha n_e \tag{2.76}$$

or equivalently

$$n_e(x) = n_{e0} \exp(\alpha x) \tag{2.77}$$

where n_{e0} and n_e are the electron number density at the starting point of the avalanche and at point x, respectively; α is the Townsend ionization coefficient that gives the electron production per unit length.

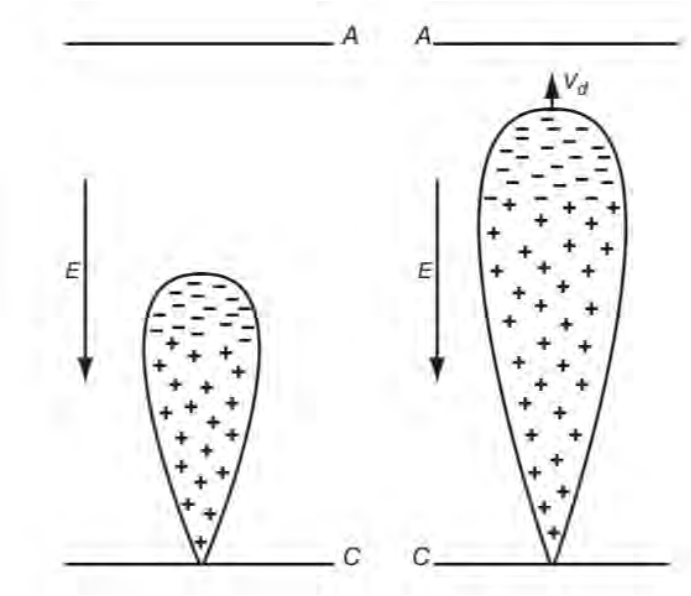


Figure 2.5: Evolution of an avalanche in a gap between cathode (C) and anode (A).

2.4.2 Townsend Breakdown Mechanism

The Townsend breakdown mechanism originates from the work on discharges in low pressure gases and is the classical theory of gas discharge. In Fig. 2.5, the electron avalanche proceeds in the opposite direction to the electric field, and electrons reach and disappear at the anode. Ions produced in the collision ionization are slowly drifting toward the cathode. Having reached the cathode, an ion knocks out an electron with a certain probability. This process is called ion bombardment. The electron then undergoes a new avalanche, whose ions again bombard the cathode to knock out new electrons. The cycle can be repeated again and again, as long as the uniform field is maintained [Raizer, 1998, p.10]. To sustain the breakdown process, Townsend theory relies on the bombardment of ions on the cathode to release seed electrons to initiate electron avalanches. Due to their large mass, ions move much slower than electrons, which results in a large time lag for the development of breakdown according to Townsend theory [Raizer, 1998, p.11].

If the gap size d and the probability for an electron to be knocked out from the cathode by ion bombardment is γ (called the secondary emission coefficient for the cathode), a self-sustaining discharge will occur if the following condition is satisfied, i.e, positive ions generated by the electron avalanche must produce at least one electron to start a new avalanche:

$$\gamma[\exp(\alpha d) - 1] = 1 \tag{2.78}$$

The left side of this equation represents the number of secondary electrons generated by the ions colliding at the surface. It must be at least equal to one for the breakdown to sustain itself. The discharge is self-sustained in a sense that it can maintain itself in a steady state of primitive reproduction of electrons in the discharge volume, when the total outflow of electrons and ions through the boundaries is fully compensated by the inflow due to the emission of secondary electrons from the cathode and production of both types of species

in the discharge volume due to ionization. Therefore, the Townsend breakdown mechanism is a mechanism of ignition of a self-sustaining discharge in a gap, controlled by secondary electron emission from the cathode. The ignition potential V_t (or equivalent critical electric field E_t) for a Townsend breakdown depends on the gas, the material of the cathode, the gas pressure p, and the inter-electrode gap size d. The actual dependence is commonly expressed as:

$$V_t = \frac{B(pd)}{C + \ln(pd)} \quad \text{or} \quad \frac{E_t}{p} = \frac{B}{C + \ln(pd)} \tag{2.79}$$

where B and C are constants and specified by the properties of the gas and the material of the electrodes.

2.4.3 Streamer Discharge Theory

The above-discussed Townsend mechanism of breakdown, which is relatively homogeneous and includes the development of independent avalanches, takes place usually at $pd < 200$ Torr cm. At larger pd values, the avalanches essentially disturb the electric field and are no longer independent, which leads to the streamer mechanism of discharges. Streamer discharge theory was put forward in the 1930s to explain spark discharges. The theory is based on the concept of a streamer. Streamers are narrow filamentary plasmas, which are driven by highly nonlinear space charge waves. The dynamics of a streamer is mostly controlled by a highly enhanced field region, known as a streamer head, and the streamer polarity is defined by the sign of the charge in its head. A schematic illustration of a positive streamer propagating in an ambient electric field is given in the left panel of Fig. 2.6. The right panel of the figure gives the distribution of electron density, electric field and space charge along the central axis of the streamer that is usually considered having cylindrical symmetry. The propagation of the positive streamer is the same as the direction of the ambient electric field. A large amount of space charges exists in the streamer head (the shaded region in the figure), which strongly enhances the electric field in the region just ahead of the streamer, while screening the ambient field out of the streamer channel (the region behind of the streamer head). The peak space charge field can reach a value about 4-7 times the breakdown field E_k. The large space charge field results in very intense impact ionization occurring in the streamer head. This ionization rapidly raises the electron density from its ambient value to the level in the streamer.

The channel of the streamer extends to the head region. In this sense, streamers are called space charge waves. The ionization in the streamer head proceeds as electron avalanches (Fig. 2.7). Multiple electron avalanches move towards the head of the streamer in the figure, increase the electron density rapidly to the channel density, and neutralize the net space charges in the head, while leaving positive ions in the trail of the avalanches. It has been known that the photoionization process is needed to generate seed electrons ahead of the streamer in order to initiate the electron avalanches. The large electric field in the head leads to existence of many high energy electrons in this region, and the collisions of those electrons with neutral molecules not only ionize neutral molecules but also excite those molecules. The excited states of molecules can radiate photons (wavy arrows in Fig. 2.7), and those photons can ionize neutral molecules beyond the head region which provide seed electrons ahead of the streamer for an electron avalanche.

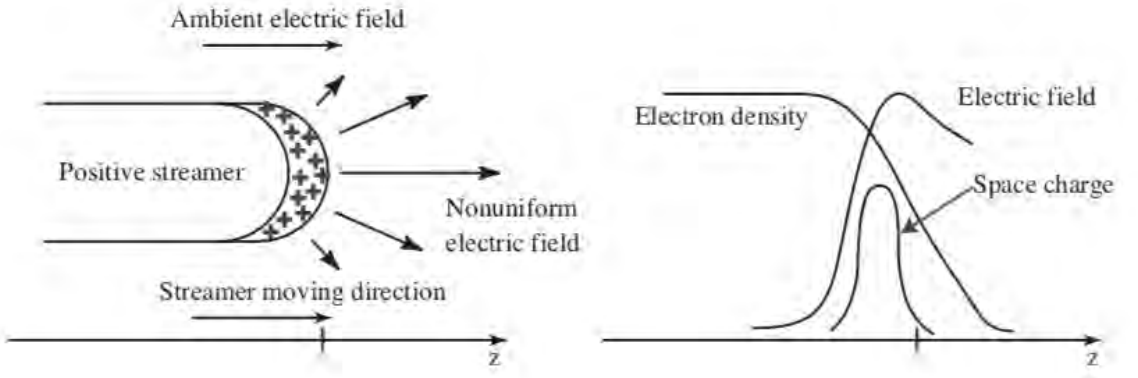


Figure 2.6: Schematic of a positive streamer propagating in an ambient field

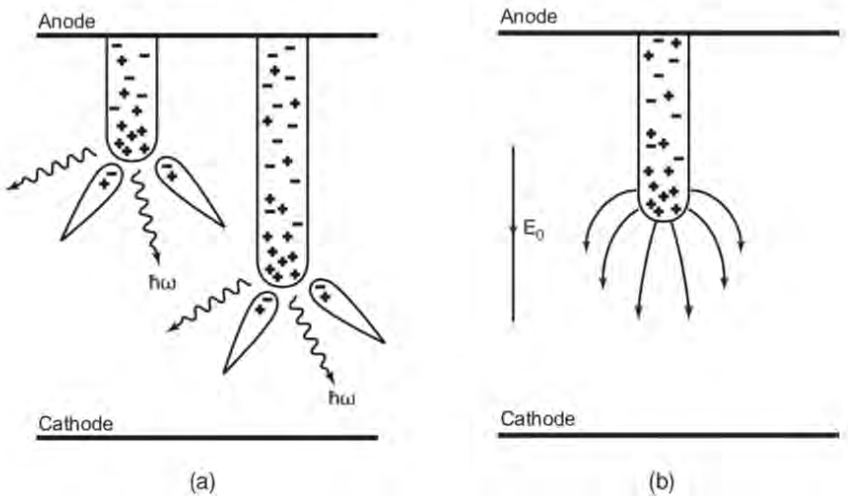


Figure 2.7: Illustration of a cathode-directed (positive) streamer: (a) propagation of the positive streamer;(b) electric field near the streamer head.

The above discussion is about positive streamers. For negative streamers, the same large electric field also appears in the streamer head and the streamer develops in a similar manner as the positive one (i.e, as ionization wave). However, there exists the opposite polarity of space charges in the streamer head, and the propagating direction of a negative streamer is opposite to the ambient electric field but the same as the drifting direction of electrons as can be seen in Fig. 2.8. Therefore it is possible that the drifting electrons initiate electron avalanches ahead of the streamer tip, which enables the streamer to advance forward without the participation of the photoionization process. However, since the speed of a streamer can reach as high as one tenth of the speed of light that is much larger than the drift velocity of electrons, it is still expected that the photoionization process plays an essential role in the development of a negative streamer.

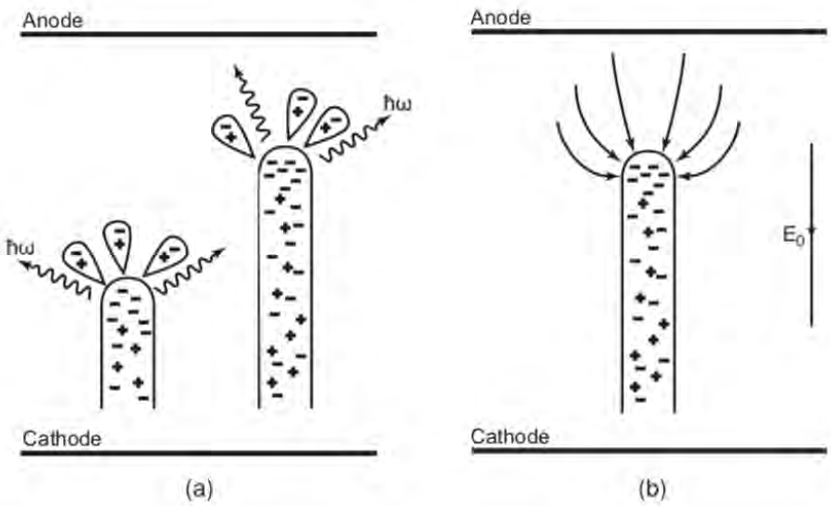


Figure 2.8: Illustration of an anode-directed (negative) streamer: (a) propagation of the positive streamer;(b) electric field near the streamer head.

2.4.4 Photoionization Model

In the streamer process the photoionization of O_2 is caused by the radiation in the region of the spectrum $980 < \lambda < 1025\,\text{Å}$. The radiations in this interval are produced due to the radiative transitions from three singlets of N_2 to the ground state. The model from Zheleznyak et al. [1982] is presented here as well as the two and three-exponential Helmholtz models for photoionization in air explained in Celestin [2008].

2.4.4.1 Integral model

In Zheleznyak et al. [1982], the number of ionization events $\mathrm{d}Q_{ph}$ in volume dV_2 per unit time owing to the absorption of photons emitted per unit time from the superposition of elementary volumes dV_1 of the source at a distance $r = |\mathbf{r}_1 - \mathbf{r}_2|$ (Fig. 2.9) can be expressed in the form $\mathrm{d}Q_{ph} = S_{ph}dV_2$ where

$$S_{ph} = \frac{\mathrm{d}Q_{ph}}{\mathrm{d}V_2} = \int_{V_1} \frac{\phi(r)}{4\pi r^2}\mathrm{d}V_1 \tag{2.80}$$

$$\phi(r) = \int \epsilon_f \xi_f k_f e^{-k_f r}\mathrm{d}f \tag{2.81}$$

and where ϵ_f is the emissivity of the gas ($\epsilon_f \mathrm{d}f$ is the number of photons emitted per unit volume per unit time in the frequency interval $[f, f + \mathrm{d}f]$), k_f is the absorption coefficient, and ξ_f is the probability of ionization through absorption of a photon with frequency f. The integral (2.81) can be approximated as a sum over frequency regions Δf characterized by different values of the absorption coefficient:

$$\phi(r) = \sum_i \frac{\xi_i q_i}{\Delta f_i} \int_{\Delta f_i} k_f e^{-k_f r}\mathrm{d}f \tag{2.82}$$

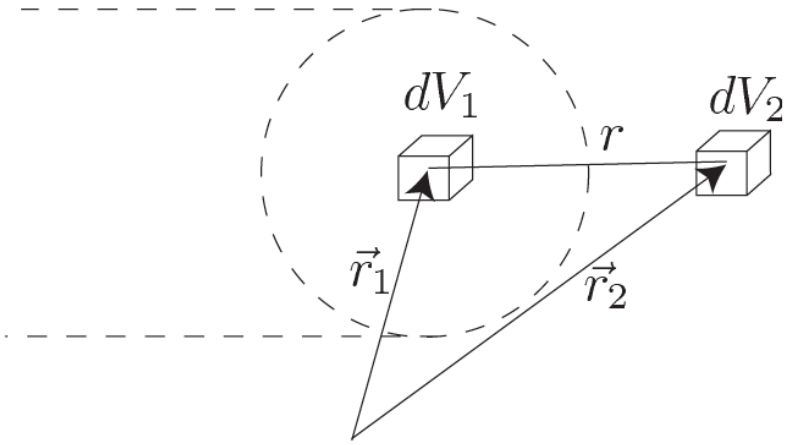


Figure 2.9: Geometry of the photoionization model

where ξ_i is the average value of ξ_f in the interval Δf_i and $q_i = \int_{\Delta f_i} \epsilon_f \mathrm{d}f$. In the $980 < \lambda < 1025$ Å range, the absorption coefficient of O_2 is a sharp function of frequency:

$$k_f = k_1 \left(\frac{k_2}{k_1}\right)^{(f-f_1)/(f_2-f_1)} \tag{2.83}$$

where $k_1 = \chi_{\min} p_{O_2}$, $k_2 = \chi_{\max} p_{O_2}$, and $\chi_{\min} = 0.035 \mathrm{Torr}^{-1}\mathrm{cm}^{-1}$ and $\chi \mathrm{max} = 2 \mathrm{Torr}^{-1}\mathrm{cm}^{-1}$ are, respectively, the minimum and maximum absorption cross section of O_2 in the interval 980-1025 Å. By considering the interval 980-1025 Å as a single frequency range:

$$\phi(r) = \frac{\xi q}{r \ln(k_2/k_1)} (e^{-k_1 r} - e^{-k_2 r}) \tag{2.84}$$

q is the average number of photons emitted per unit time per unit volume in the frequency range 980-1025 Å and can be expressed as:

$$q = \frac{p_q}{p + p_q} q_0 \tag{2.85}$$

where $\frac{p_q}{p+p_q}$ is a quenching multiplier, p is the pressure and p_q is the quenching pressure of the singlet states of N_2. q_0 is the emissivity in the absence of quenching and can be expressed as:

$$q_0 = \frac{\nu_*}{\nu_i} \nu_i n_e \tag{2.86}$$

where ν_* is the effective ionization frequency, ν_i the ionization frequency and n_e the

electron density. In the end Zheleznyak model yields:

$$\phi(r) = \frac{p_q}{p + p_q} \xi \frac{\nu_*}{\nu_i} \nu_i n_e g(r) \tag{2.87}$$

$$g(r) = \frac{\exp(-\chi_{\min} p_{O_2}) - \exp(-\chi_{\max} p_{O_2})}{r \ln(\chi_{\max}/\chi_{\min})} \tag{2.88}$$

2.4.4.2 Two and three-exponential Helmholtz models for photoionization in air

To compute the photoionization source term a volume integral needs to be calculated at every point of the mesh which is extremely costly. To minimize the cost, another coarser grid could be defined and the source term could be reinterpolated on the initial grid. Another way of solving this photoionization source term is presented here using an exponential fit of the g function in Eq. (2.88). This function is interpolated as follows:

$$\frac{g(r)}{p_{O_2}} = p_{O_2} r \sum_j A_j e^{-\lambda_j p_{O_2} r} \tag{2.89}$$

Here j is equal to 2 or 3 depending on the degree of precision wanted for the interpolation. The coefficients for the two and three-exponential Helmholtz models are given in 2.1 and 2.2. The graphs of the real g(r) function and the exponential fits are given in Fig. 2.10.

j	A_j [$\mathrm{cm^{-2}Torr^{-2}}$]	λ_j [$\mathrm{cm^{-1}Torr^{-1}}$]
1	0.0021	0.0974
2	0.1775	0.5877

Table 2.1: Parameters of the two exponential model

j	A_j [$\mathrm{cm^{-2}Torr^{-2}}$]	λ_j [$\mathrm{cm^{-1}Torr^{-1}}$]
1	1.986×10^{-4}	0.0553
2	0.0051	0.1460
3	0.4886	0.89

Table 2.2: Parameters of the three exponential model

By doing so the photoionization source term becomes a sum of three source terms:

$$S_{ph}(\mathbf{r}) = \sum_j S^j_{ph}(\mathbf{r}) \tag{2.90}$$

$$S^j_{ph}(\mathbf{r}) = \int_{V'} \frac{I(\mathbf{r}') A_j p_{O_2} e^{-\lambda_j p_{O_2} R}}{4\pi R} dV' \tag{2.91}$$

where $R = |\mathbf{r} - \mathbf{r}'|$. Each $S_p h^j$ satisfies the following equation:

$$\nabla^2 S^j_{ph}(\mathbf{r}) - (\lambda_j p_{O_2})^2 S^j_{ph}(\mathbf{r}) = -A_j p^2_{O_2} I(\mathbf{r}) \tag{2.92}$$

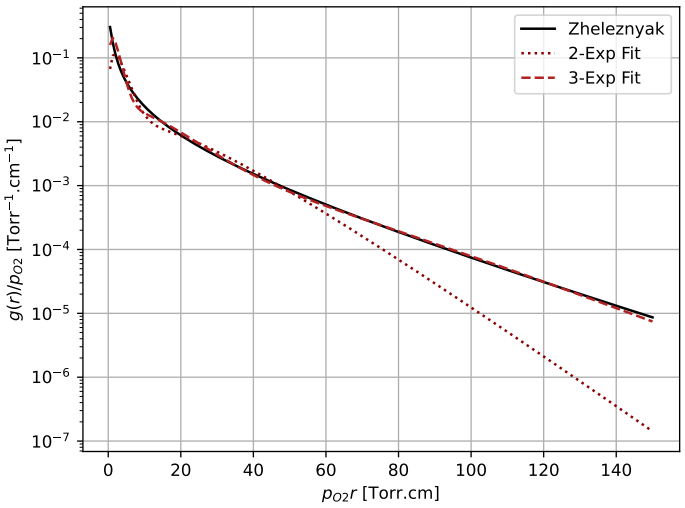


Figure 2.10: g(r) and its exponential fits.

Chapter 3

Turbulent Combustion

Contents

This chapter is dedicated to turbulent combustion. Thermodynamic relations of gases relevant to the subsequent sections and chapters are first recalled. Different formulations of the Navier-Stokes equations are then presented with an emphasis on cylindrical frames as axisymmetric simulations are carried out in this thesis. Next the complete equations for reacting flows are laid out with specific relations concerning multispecies mixture of gases.

3.1 Thermodynamic relations

Classical thermodynamics relations are given in this section. Any non-reactive system in thermodynamic equilibrium is completely described by the basic thermodynamic variables p (pressure) and $v = V/m = 1/\rho$ (specific volume). Temperature can be introduced through a thermal equation of state:

$$T = T(p, v) \tag{3.1}$$

which is here the perfect gas law for ideal gases.

To close the Euler equations we need a caloric equation of state relating internal energy e to pressure and density. In fact for the Euler equations only this equation of state is needed:

$$e = e(p, \rho) \tag{3.2}$$

The first principle for the specific internal energy states that for a closed system (no transfert of matter) $\mathrm{d}e = \delta q + \delta w$ *i.e.* the change in internal energy is due to the work done on the system and the heat transmitted to the system. If the work done on the system is only due to pressure forces then $\delta w = -p_{\text{ext}}\mathrm{d}v$. If the system is going through a reversible transformation then $p_{\text{ext}} = p$.

The second law introduces a new state variable, the specific entropy, such that $\mathrm{d}s = \delta q/T$ for a reversible process and $\mathrm{d}s > \delta q/T$ for an irreversible one. Thus we get

$$\mathrm{d}e = T\mathrm{d}s - p\mathrm{d}v \tag{3.3}$$

from which the canonical equation of state $e = e(s, v)$ is obtained. Other state functions can be defined such as the enthalpy $h = e + pv = h(s, p)$, the Helmholtz free energy $f = e - Ts = f(T, v)$ and the Gibbs free energy $g = h - Ts = g(T, p)$. If a thermal equation of state $v = v(p, T)$ is given then one can define the volume expansivity α and isothermal compressibility β:

$$\alpha = \frac{1}{v}\left(\frac{\partial v}{\partial T}\right)_p \tag{3.4}$$

$$\beta = -\frac{1}{v}\left(\frac{\partial v}{\partial p}\right)_T \tag{3.5}$$

Using the cycle rule and the chain rule we can obtain

$$\left(\frac{\partial s}{\partial v}\right)_T = \left(\frac{\partial p}{\partial T}\right)_v = \frac{\alpha}{\beta} \tag{3.6}$$

$$\left(\frac{\partial e}{\partial v}\right)_T = T\frac{\alpha}{\beta} - p \tag{3.7}$$

In general when an addition of heat δq leads to an increase in temperature δT the ratio $c = \delta q/\delta T$ is called the heat capacity of the system. The heat capacity at constant volume and constant pressure are then defined as:

$$c_v = \left(\frac{\partial e}{\partial T}\right)_v = T\left(\frac{\partial s}{\partial T}\right)_v \tag{3.8}$$

$$c_p = \left(\frac{\partial h}{\partial T}\right)_p = T\left(\frac{\partial s}{\partial T}\right)_p \tag{3.9}$$

The speed of sound is another variable of fundamental interest. Given $p = p(\rho, s)$ it is defined as

$$a = \sqrt{\left(\frac{\partial p}{\partial \rho}\right)_s}. \tag{3.10}$$

3.1.1 Ideal gases

For ideal gases the fundamental equation of state $pv = RT$ holds. This implies that

$$\alpha = \frac{1}{T} \quad \text{and} \quad \beta = \frac{1}{p} \implies \left(\frac{\partial e}{\partial v}\right)_T = 0 \tag{3.11}$$

The internal energy is thus only a function of temperature $e = e(T)$. It is possible to relate c_p and c_v via the general relation

$$c_p - c_v = \frac{\alpha^2 T v}{\beta} \tag{3.12}$$

which reduces to the classical $c_p - c_v = R$ for ideal gases. Defining the ratio of heat capacities $\gamma = c_p/c_v$ one obtains:

$$c_v = \frac{R}{\gamma - 1} \tag{3.13}$$

$$c_p = \frac{\gamma R}{\gamma - 1} \tag{3.14}$$

From kinetic theory and statistical physics it is known that each translational and rotational degree of freedom accounts for an energy of $RT/2$, so that the heat capacity for such molecules is

$$c_v = \frac{M}{2} R \tag{3.15}$$

where M is the number of degrees of freedom. From this heat capacity the caloric equation for ideal gases, an exact equation for entropy s and the expression for the sound speed a are retrieved:

$$e = \frac{p}{\rho(\gamma - 1)} \tag{3.16}$$

$$s = c_v \ln\left(\frac{p}{\rho^\gamma}\right) + s_0 \tag{3.17}$$

$$a = \sqrt{\frac{\gamma p}{\rho}} \tag{3.18}$$

In reality, the activation of degrees of freedom depend on temperature so that the heat capacity c_v is not constant but a function of temperature.

3.1.2 Covolume and van der Waals gases

Taking into account the volume of molecules leads to the covolume equation of state

$$p(v-b) = RT \tag{3.19}$$

where b is the covolume and for which the internal energy and sound speed are given by

$$e = \frac{p(1-b\rho)}{\rho(\gamma-1)} \tag{3.20}$$

$$a = \sqrt{\frac{\gamma p}{(1-b\rho)\rho}} \tag{3.21}$$

Finally the attractive forces between molecules are taken into account in the van der Waals equation of state

$$\left(p + \frac{c}{v^2}\right)(v-b) = RT \tag{3.22}$$

3.2 Navier-Stokes equations

The Navier-Stokes equations in cartesian and cylindrical frames are discussed in this section as both types of frames are used in the simulations of this thesis. In tensor form the Navier-Stokes (NS) equations are given by the following system of equations valid in all types of coordinates frames:

$$\frac{\partial \mathbf{U}}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{F} = 0 \tag{3.23}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho\mathbf{u} \\ \rho E \end{bmatrix} \quad \text{and} \quad \mathbf{F} = \mathbf{F}^C + \mathbf{F}^D = \begin{bmatrix} \rho\mathbf{u} \\ \rho\mathbf{u}\otimes\mathbf{u} + p\mathbf{I} \\ (\rho E + p)\mathbf{u} \end{bmatrix} + \begin{bmatrix} 0 \\ -\boldsymbol{\tau} \\ -\boldsymbol{\tau}\cdot\mathbf{u} + \mathbf{q} \end{bmatrix} \tag{3.24}$$

where ρ is the mass density, $\mathbf{u}$ is the speed, E is the total energy per mass unit, p is the pressure, $\boldsymbol{\tau}$ is the viscous flux and $\mathbf{q}$ the energy flux vector. The total flux is splitted into a convective part $\mathbf{F}^C$ and a diffusive part $\mathbf{F}^D$. The Navier Stokes equations reduce to the Euler equations if $\mathbf{F}^D = 0$ *i.e.* no diffusion is taken into account.

3.2.1 Cartesian coordinates

In 3D cartesian coordinates, the Navier-Stokes equations are explicitely given by

$$\mathbf{U}_t + \nabla \cdot (\mathbf{F}^C + \mathbf{F}^D) = 0 \tag{3.25}$$

with

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ \rho E \end{bmatrix} \quad \mathbf{F}^C = \begin{bmatrix} \rho u_x & \rho u_y & \rho u_z \\ \rho u_x^2 + p & \rho u_x u_y & \rho u_x u_z \\ \rho u_y u_x & \rho u_y^2 + p & \rho u_y u_z \\ \rho u_z u_x & \rho u_z u_y & \rho u_z^2 + p \\ (\rho E + p) u_x & (\rho E + p) u_y & (\rho E + p) u_z \end{bmatrix}$$

and

$$\mathbf{F}^D = - \begin{bmatrix} 0 & 0 & 0 \\ \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \tau_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \tau_{zz} \\ \tau_{xi} u_i - q_x & \tau_{yi} u_i - q_y & \tau_{zi} u_i - q_z \end{bmatrix}$$

3.2.2 Axisymmetric conditions

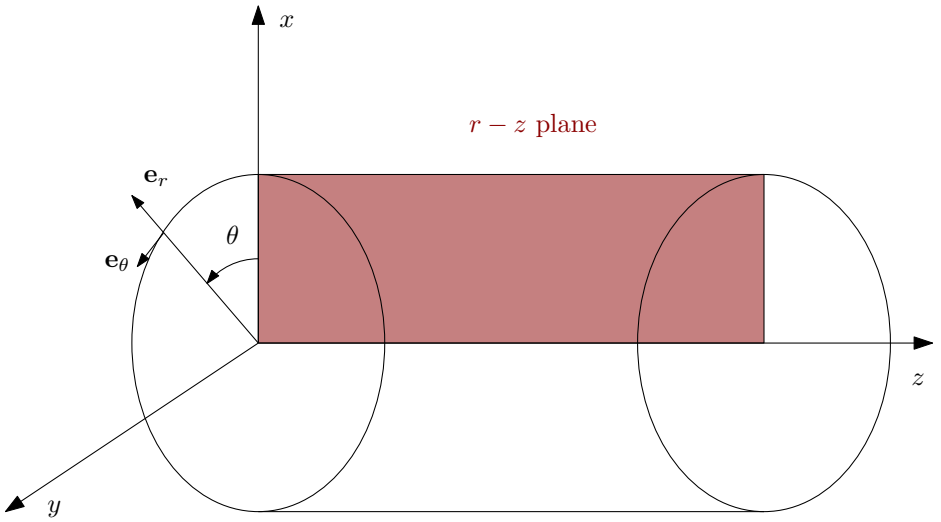


Figure 3.1: Cylindrical frame in $r - z$ plane

In axisymmetric configurations we have $\dfrac{\partial}{\partial \theta} = 0$. Expansions of tensor operators in cylindrical coordinates are recalled in Appendix. C as well as their proofs. Multiplying by r Eq. (C.13) and setting $\mathbf{t} = \rho \mathbf{u} \otimes \mathbf{u} + p\mathbf{I} - \boldsymbol{\tau}$ the Navier-Stokes equations in axisymmetric conditions are given by

$$\frac{\partial r\mathbf{U}}{\partial t} + \boldsymbol{\nabla}_{2\mathrm{D}} \cdot (r\mathbf{F}^C_{rz} + r\mathbf{F}^D_{rz}) = \mathbf{S}^C_{rz} + \mathbf{S}^D_{rz} \tag{3.26}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u_r \\ \rho u_\theta \\ \rho u_z \\ \rho E \end{bmatrix} \qquad \mathbf{F}^C_{rz} = \begin{bmatrix} \rho u_r & \rho u_z \\ \rho u_r u_r + p & \rho u_r u_z \\ \rho u_\theta u_r & \rho u_\theta u_z \\ \rho u_z u_r & \rho u_z u_z + p \\ (\rho E + p) u_r & (\rho E + p) u_z \end{bmatrix} \qquad \mathbf{S}^C_{rz} = \begin{bmatrix} 0 \\ \rho u_\theta u_\theta + p \\ -\rho u_r u_\theta \\ 0 \\ 0 \end{bmatrix} \tag{3.27}$$

and

$$\mathbf{F}^D_{rz} = - \begin{bmatrix} 0 & 0 \\ \tau_{rr} & \tau_{rz} \\ \tau_{\theta r} & \tau_{\theta z} \\ \tau_{zr} & \tau_{zz} \\ \tau_{xi} u_i - q_x & \tau_{zi} u_i - q_z \end{bmatrix} \qquad \mathbf{S}^D_{rz} = \begin{bmatrix} 0 \\ -2\eta \frac{u_r}{r} - \lambda \left(\frac{u_r}{r} + \frac{\partial u_r}{\partial r} + \frac{\partial u_z}{\partial z} \right) \\ \eta \left(\frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r} \right) \\ 0 \\ 0 \end{bmatrix} \tag{3.28}$$

This allows us to solve the Euler equations only in a two-dimensional mesh while the flow is actually three-dimensional. As depicted in Fig. 3.1, the mesh is only in the r-z plane but we can solve for the azimuthal velocity. Two difficulties arise: the need to include the radiuses in the discretization and geometrical source terms. The latter make the equations non-conservative and special care must be taken when applying them. We can further simplify the above expressions as in streamer simulations for example, setting all the θ variables to zero:

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u_r \\ \rho u_z \\ \rho E \end{bmatrix} \qquad \mathbf{S}^C_{rz} = \begin{bmatrix} 0 \\ p \\ 0 \\ 0 \end{bmatrix} \qquad \mathbf{S}^D_{rz} = \begin{bmatrix} 0 \\ -2\eta \frac{u_r}{r} - \lambda \left(\frac{u_r}{r} + \frac{\partial u_r}{\partial r} + \frac{\partial u_z}{\partial z} \right) \\ 0 \\ 0 \end{bmatrix} \tag{3.29}$$

Another derivation of these equations with somewhat different notations can be found in [Nishikawa, 2017, 3.7, 4.11].

3.3 Governing equations for reacting flows

3.3.1 Conservation equations for reacting flows

A reactive mixture of N species is considered. Compared to classical Navier-Stokes equations, each species k in the mixture must be tracked individually. The species react which each other and the rate at which the species react requires specific modeling [Poinsot and Veynante, 2012]. The primitive variables are thus:

- the density $\rho = m/V$
- the velocity field of the mixture $\mathbf{u}$
- one variable for energy or temperature

- the mass fractions of each species $Y_k = m_k/m$

We also introduce the mole fractions of each species $X_k = n_k/n$. In the following, all state functions (such as the mass enthalpy h) with small letters are mass quantities, when superscript m is applied these are molar quantities (molar enthalpy h^m) and finally when a capital letter is used it is the total quantity ($H = mh = nh^m$). The perfect gas law generalizes to a mixture by defining partial densities $\rho_k = \rho Y_k$ and partial pressures p_k such that:

$$\rho = \sum_{k=1}^{N} \rho_k \tag{3.30}$$

$$p_k = \rho_k \frac{R}{W_k} T \tag{3.31}$$

$$p = \sum_{k=1}^{N} p_k = \frac{\rho R T}{W} \tag{3.32}$$

where R is the perfect gas constant, W_k the molecular weight of species k, p the pressure of the mixture and W the mixture molecular weight given by

$$\frac{1}{W} = \sum_{k=1}^{N} \frac{Y_k}{W_k} \tag{3.33}$$

$$W = \sum_{k=1}^{N} X_k W_k \tag{3.34}$$

Various energies can be defined and depending on the situation one formulation can be more appropriate than the other. The sensible energy e_{sk} represents the part of the energy of a species that varies with the temperature while the energy e_k of species k includes the constant formation enthalpy. These energies are defined by:

$$e_{sk} = \int_{T_0}^{T} c_{vk} dT - \frac{R T_0}{W_k} \tag{3.35}$$

$$e_k = e_{sk} + \Delta h^0_{f,k} \tag{3.36}$$

where $\Delta h^0_{f,k}$ is the mass enthalpy formation of species k at temperature T_0. When working at constant pressure, the enthalpies defined as $h_{sk} = e_{sk} + p_k/\rho_k$ $h_k = e_k + p_k/\rho_k$ are more relevant. Finally from these definitions the sensible e_s and total non chemical E energies are

$$e_s = \sum_k Y_k e_{sk} \tag{3.37}$$

$$E = e_s + \frac{1}{2}\mathbf{u}^2 \tag{3.38}$$

In total non-chemical energy ρE formulation, the reactive Navier-Stokes equations are [Poinsot and Veynante, 2012, Chap. 1]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \tag{3.39}$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u} + p\mathbf{I} - \boldsymbol{\tau}) = \rho \sum_{k=1}^{N} Y_k \mathbf{f}_k \tag{3.40}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot [(\rho E + p)\mathbf{u} - \boldsymbol{\tau} \cdot \mathbf{u} + \mathbf{q}] = \dot{\omega}_T + \dot{Q} + \rho \sum_{k=1}^{N} Y_k \mathbf{f}_k \cdot (\mathbf{u} + \mathbf{V}_k) \tag{3.41}$$

$$\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot [\rho Y_k (\mathbf{u} + \mathbf{V}_k)] = \dot{\omega}_k \tag{3.42}$$

where p is the pressure, $\boldsymbol{\tau}$ the viscous stress tensor, $\mathbf{f}_k$ the mass force acting on species k, $\mathbf{q}$ the heat flux, $\dot{\omega}_T$ the heat release rate due to combustion, $\dot{Q}$ an external energy source, $\dot{\omega}_k$ the mass production rate of species k and $\mathbf{V}_k$ the diffusion velocity of species k into the mixture. The energy equation is derived from the total energy equation and several other formulations are possible [Poinsot and Veynante, 2012, Chap. 1.1.5]. The total non chemical energy equation is the one that is used throughout the simulations of this thesis.

3.3.2 Diffusion modelling

Diffusive effects can be found in the momentum, energy and species k conservation equations and their modelling is discussed in this section. In the Newtonian approximation, the viscous stress is given by Toro [2009], Nishikawa [2017]

$$\boldsymbol{\tau} = 2\eta \mathbf{d} + \lambda (\nabla \cdot \mathbf{u}) \mathbf{I} \tag{3.43}$$

where $\mathbf{d} = \frac{1}{2}(\nabla \mathbf{u} +^t \nabla \mathbf{u})$ is the deformation tensor, η the viscosity coefficient and λ the second viscosity coefficient. The bulk velocity is defined as

$$\eta_b = \frac{2}{3}\eta + \lambda \tag{3.44}$$

so that molecular theory tells us that for monoatomic gases $\eta_b = 0$ which agrees well with experiments whereas for polyatomic gases $\eta_b \neq 0$. Setting $\eta_b = 0$ is called Stokes' hypothesis (1845). The viscosity coefficient can be approximated by the Sutherland formula:

$$\eta = C_1 \left[1 + \frac{C_2}{T}\right]^{-1} \sqrt{T} \tag{3.45}$$

where $C_1 = 1.46 \times 10^{-6}$ and $C_2 = 112\,\mathrm{K}$ for air.
For a reactive mixture the energy flux is

$$\mathbf{q} = -\lambda \nabla T + \rho \sum_{k=1}^{N} h_k Y_k \mathbf{V}_k \tag{3.46}$$

where λ is the thermal conductivity. Molecular theory tells us that λ is directly proportional

to η so that the Prandtl number

$$P_r = \frac{c_p \eta}{\lambda} \tag{3.47}$$

is very nearly constant. For air and $200 \leq T \leq 1000$ K, its value is close to 0.7.

The diffusion velocities of species k into the mixture are given by the Hirschfelder and Curtiss approximation [Hirschfelder et al., 1954]:

$$\mathbf{V}_k = -D_k \frac{\nabla X_k}{X_k} + \mathbf{V}_c \tag{3.48}$$

where $\mathbf{V}_c = \sum_k W_k D_k \nabla X_k / W$ is a correction velocity to ensure mass conservation. This is a simplified model as a complete description of species diffusion is very costly [Giovangigli, 1999]. The Lewis number Le_k and Schmidt number Sc_k are defined to compare species diffusions with respectively the thermal diffusion and momentum diffusion:

$$Le_k = \frac{D_{th}}{D_k} \tag{3.49}$$

$$Sc_k = \frac{\nu}{D_k} = P_r Le_k \tag{3.50}$$

3.3.3 Chemical reactions and equilibrium

3.3.3.1 Chemistry description

The N species considered here react following M reactions denoted by:

$$\sum_{k=1}^{N} \nu'_{kj} \mathcal{M}_k = \sum_{k=1}^{N} \nu''_{kj} \mathcal{M}_k \tag{3.51}$$

where ν'_{kj} and ν''_{kj} are the reactive and products stochiometric coefficients. The net stochiometric coefficients are $\nu_{kj} = \nu''_{kj} - \nu'_{kj}$ and by mass conservation the following equations are enforced:

$$\sum_{k=1}^{N} \nu_{kj} W_k = 0 \tag{3.52}$$

The mass production rate of species k can now be expanded over all reactions j

$$\dot{\omega}_k = \sum_{j=1}^{M} \dot{\omega}_{kj} = \sum_{j=1}^{M} \nu_{kj} W_k \dot{Q}_j \tag{3.53}$$

where $\dot{Q}_j$ is the rate of reaction j given by a balance between the forward and backward reactions

$$\dot{Q}_j = K_{fj} \prod_{k=1}^{N} [X_k]^{\nu'_{kj}} - K_{rj} \prod_{k=1}^{N} [X_k]^{\nu''_{kj}} \tag{3.54}$$

The forward reaction rates K_{fj} often take the form of the Arrhenius law

$$K_{fj} = A_{fj} T^{\beta_j} \exp\left(-\frac{E_a}{RT}\right) \tag{3.55}$$

where the A_{fj} is preexponential factor, β_j the temperature exponent and E_a the activation energy. Given a forward reaction rate K_{fj}, the backward or reverse reaction rate K_{rj} can be computed through

$$K_{rj} = \frac{K_{fj}}{\left(\frac{p_a}{RT}\right)^{\sum_{k=1}^{N} \nu_{kj}} \exp\left(\frac{\Delta S_j^0}{R} - \frac{\Delta H_j^0}{RT}\right)} \tag{3.56}$$

where ΔS_j^0 is the reaction j entropy change and ΔH_j^0 the reaction j enthalpy change.

3.3.3.2 Chemical equilibrium

The meaning of chemical equilibrium is discussed in this section. The steps leading to Eq. (3.56) are detailed with the introduction of the Gibbs free energy. For a chemical reaction the equilibrium constant is defined as the balance between forward and backward rates:

$$K_{\text{eq}} = \frac{K_{fj}}{K_{rj} \left(\frac{p_a}{RT}\right)^{\sum_{k=1}^{N} \nu_{kj}}} \tag{3.57}$$

To express the rate constant K_{eq}, we introduce the Gibbs free energy G of the system as

$$G = H - TS \tag{3.58}$$

where H is the total enthalpy of the system and S the total entropy. For a constant pressure process $\delta Q = \mathrm{d}H$ and from the second law assuming furthermore an isothermal process [Lieberman and Lichtenberg, 2005, Chap. 7] such that

$$\mathrm{d}G = \mathrm{d}H - T\mathrm{d}S \leq 0 \tag{3.59}$$

If we let $G = G(T, p, n_k)$ define the state of the system, we obtain by differentiation

$$\mathrm{d}G = \left(\frac{\partial G}{\partial T}\right)_{p,n_k} \mathrm{d}T + \left(\frac{\partial G}{\partial p}\right)_{T,n_k} \mathrm{d}p + \sum_{k=1}^{N} \left(\frac{\partial G}{\partial n_k}\right)_{T,p,n_{i\neq k}} \mathrm{d}n_k \tag{3.60}$$

where the first partial derivative can be identified and the chemical potential μ_k is defined

$$\left(\frac{\partial G}{\partial T}\right)_{p,n_k} = -S \tag{3.61}$$

$$\left(\frac{\partial G}{\partial p}\right)_{T,n_k} = V \tag{3.62}$$

$$\left(\frac{\partial G}{\partial n_k}\right)_{T,p,n_{i\neq k}} = \mu_k \tag{3.63}$$

From Eqs. (3.61) and (3.62) and using the Euler theorem for a single substance the chemical potential for a gas is found to be

$$\mu_k(T) = \mu_k^0(T) + RT \log\left(\frac{x_k p}{p_a}\right) \tag{3.64}$$

Finally chemical equilibrium is considered at constant pressure and temperature so that in the end

$$\mathrm{d}G = 0 \implies \sum_{k=1}^{N} \mu_k \mathrm{d}n_k = 0 \tag{3.65}$$

$$\implies K_{\mathrm{eq}}(T) = \exp\left(-\frac{\Delta G^0(T)}{RT}\right)\left(\frac{p_a}{RT}\right)^{\sum_{k=1}^{N}\nu_{kj}} \tag{3.66}$$

In the end the backward rates are computed from the forward rates using

$$K_{rj} = \frac{K_{fj}}{(\frac{p_a}{RT})^{\sum_{k=1}^{N}\nu_{kj}} \exp\left(\frac{\Delta S_j^0}{R} - \frac{\Delta H_j^0}{RT}\right)} \tag{3.67}$$

where ΔS_j^0 is the reaction j entropy change and ΔH_j^0 the reaction j enthalpy change.

3.4 Combustion

Flames may be classified into four categories shown in Fig. 3.2. These categories are based on two criterions: mixing and turbulence. In a premixed flame, reactants are mixed before entering the combustion chamber whereas mixing occur inside the combustion chamber for non-premixed or diffusion flames. A brief overview of premixed and non-premixed flames is recalled here. More details can be found in [Poinsot and Veynante, 2012, Chap. 2 to 6].

3.4.1 Premixed flames

Considering the global combustion reaction of fuel and oxidizer as follows

$$\nu'_F \mathrm{F} + \nu'_O \mathrm{O} \rightarrow \text{Products} \tag{3.68}$$

the mass stochiometric ratio s is defined

$$s = \left(\frac{Y_O}{Y_F}\right)_{\mathrm{st}} = \frac{\nu'_O W_O}{\nu'_F W_F} \tag{3.69}$$

and the mixture equivalence ratio is then

$$\phi = s\left(\frac{Y_F}{Y_O}\right) \tag{3.70}$$

The equivalence ratio indicates if the mixture is lean ($\phi < 1$), *i.e.* the oxidizer is in excess, or rich ($\phi > 1$), *i.e.* the fuel is in excess. Most burners operate in lean or close to stochiometric conditions.

(a) Laminar premixed flame.

(b) Turbulent premixed flame.

(c) Laminar diffusion flame.

(d) Turbulent diffusion flame.

Figure 3.2: Four flame configurations.

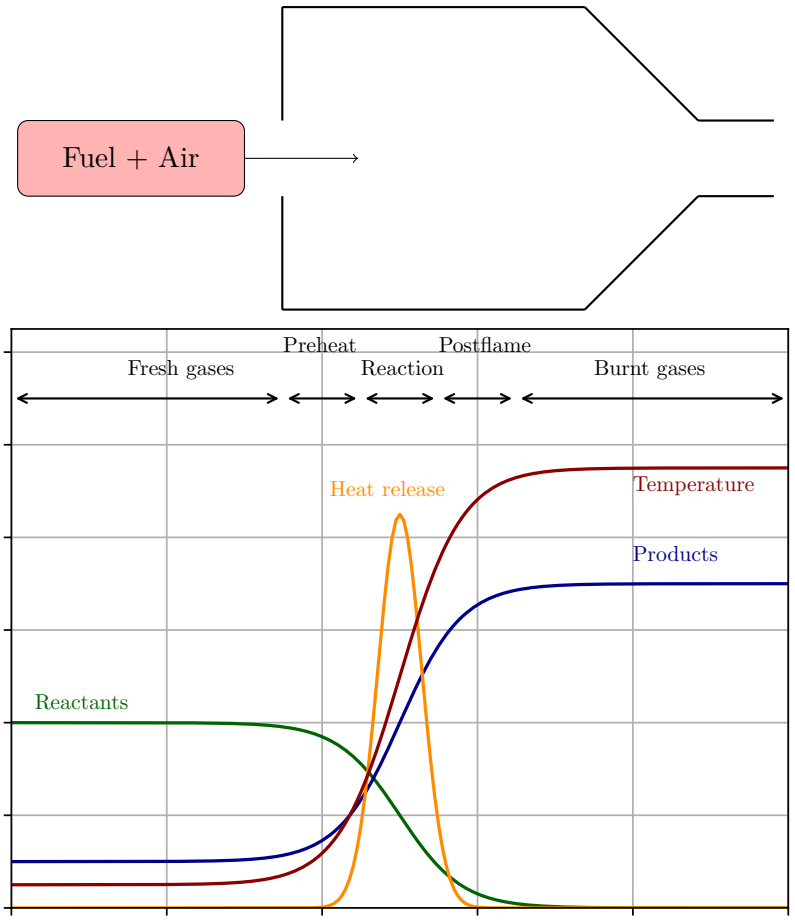


Figure 3.3: Canonical premixed flame configuration (top) and structure (bottom).

One dimensional laminar premixed flames are the canonical flames of combustion. It is one of the few configurations where theoretical, computational and experimental results exist and are compared thoroughly [Poinsot and Veynante, 2012, Chap. 2]. They also represent the building block of turbulent flames with the flamelet theory. One-dimensional laminar flames represent a propagating front between the burnt gases to the fresh gases which reaches a constant propagation velocity s_L^0 at steady state. In the flame reference frame the governing equations yield

$$\rho u = \text{cst} = \rho_1 s_L \tag{3.71}$$

$$\rho_1 s_L^0 \frac{\mathrm{d}Y_F}{\mathrm{d}x} = \frac{\mathrm{d}}{\mathrm{d}x}\left(\rho D \frac{\mathrm{d}Y_F}{\mathrm{d}x}\right) + \dot{\omega}_F \tag{3.72}$$

$$\rho_1 c_p s_L^0 \frac{\mathrm{d}T}{\mathrm{d}x} = \frac{\mathrm{d}}{\mathrm{d}x}\left(\lambda \frac{\mathrm{d}T}{\mathrm{d}x}\right) - Q\dot{\omega}_F \tag{3.73}$$

A typical premixed flame structure is given in Fig. 3.3. Several characteristic zones can be identified from left (fresh gases) to right (burnt gases):

- the preheat zone, chemically inert, where heat diffuses progressively from the reaction zone and increases gas temperature
- the reaction layer where the fuel is decomposed in smaller hydrocarbons which react with radicals to create secondary species
- the post-flame region where secondary species are transformed in final products of the reaction

The laminar flame speed is an important quantity to assess the quality of simulations compared to experiments. Simplifying assumptions allow to obtain theoretical expressions of the laminar flame speed [Poinsot and Veynante, 2012, Chap. 2] which exhibits the following relationship

$$s_L^0 \propto \sqrt{D_{th} A} \tag{3.74}$$

where A is the preexponential factor of the global reaction considered. Various flame thicknesses can be defined in premixed flames:

- the diffusion thickness
$$\delta = \frac{D_{th}}{s_L^0} \propto \sqrt{\frac{D_{th}}{A}} \tag{3.75}$$
- the thermal layer thickness
$$\delta_L^0 = \frac{T_b - T_f}{\max(|\nabla T|)} \tag{3.76}$$
- the reaction layer thickness
$$\delta_r = \frac{\delta_L^0}{\beta} \tag{3.77}$$

3.4.2 Diffusion flames

Premixing can sometimes be unpractical or unsafe to do and in this case combustion must be done in a non-premixed way: these are the diffusion or non-premixed flames. Diffusion flames are not studied in this work and only a brief explanation is given for completeness.

Mixing occurs inside the combustion chamber for a diffusion flame as shown in Fig. 3.4 and therefore two states must be considered: the fuel injection and the oxidizer injection. On each side (diffusion zones in Fig. 3.4) the gas is either too rich or too lean to burn and only when the two streams interact combustion can occur. The most-efficient combustion is obtained when local stochiometry is close to one (reaction zone in Fig. 3.4). Contrary to premixed flames, no reference speed or thickness can be defined for diffusion flames. They are simpler to design and safer to use but at the cost of lower combustion efficiency.

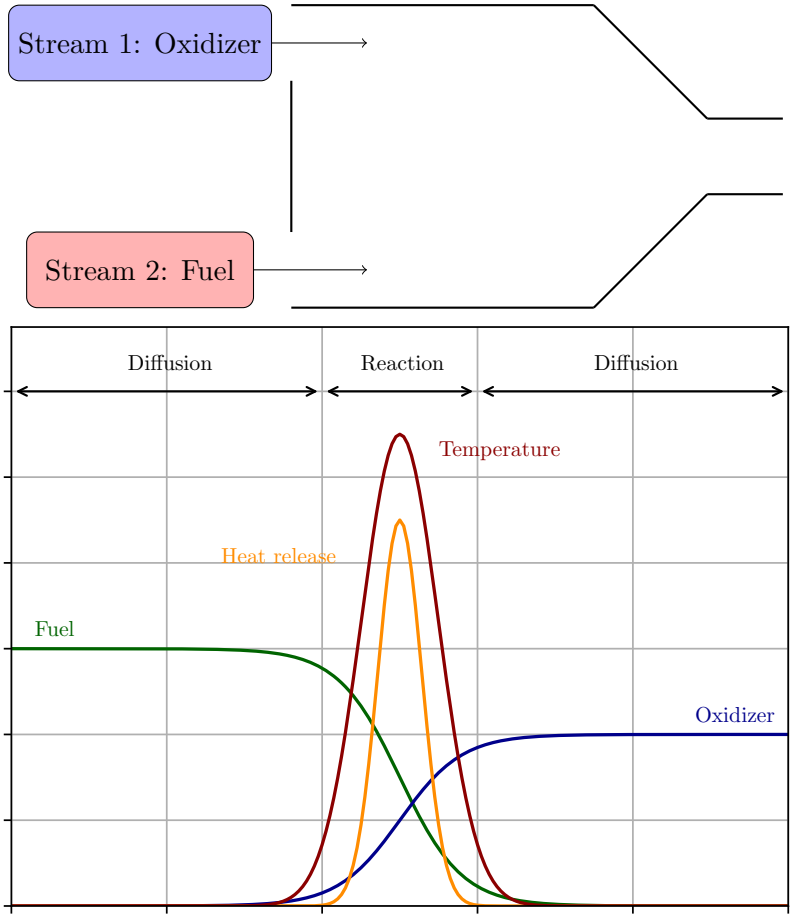


Figure 3.4: Canonical diffusion flame configuration (top) and structure (bottom).

3.5 Turbulent Combustion

A quick overview of turbulent combustion is given here, although no simulation involving turbulence is performed in this thesis. The models developed for PAC in the thesis are

however applicable to turbulent combustion and could be the subject of future works.

3.5.1 Basics of turbulence

In the Navier-Stokes equation the Reynolds number compares the contribution of the inertia forces and the viscours forces in the momentum equation

$$Re = \frac{||\mathbf{u}||L}{\nu} \tag{3.78}$$

where L is a characteristic length of the flow. As this number increases the Navier Stokes equations become chaotic and their sensitivity to perturbations such as small changes in initial conditions or boundary conditions is increased [Pope, 2000]. Systems much simpler than the Navier-Stokes equations like the Lorentz system also share this sensitivity and their behavior is well understood in the study of dynamical systems.

In a turbulent flow compared to a laminar flow, the velocity field $\mathbf{u}(\mathbf{x}, t)$ becomes random because of the chaotic behavior of the Navier Stokes equations. A multitude of vortices of varying size, called eddies, are convected by the mean flow. These eddies interact with each other and can exchange energy. Kinetic energy is dissipated through a cascade process shown in Fig. 3.5 where three distincit regions can be identified when looking at the energy spectra of those eddies:

1. The **integral zone** corresponds to the eddies of largest size l_t with characteristic velocity u'_t. Turbulent kinetic energy is produced at these scale. The size of these eddies is of the order of magnitude of the characteristic length of the problem.

2. The largest eddies break down into smaller ones in the **inertial zone** where there is a transfer of energy from low to high frequency.

3. In the **dissipation zone**, the viscous forces dominate and the energy is dissipated as heat by the kinematic viscosity ν. The typical length scale is the Kolmogorov length scale η_K and the following relationships hold

$$\eta_K = \left(\frac{\nu^3}{\varepsilon}\right)^{1/4} \qquad u'_K = (\nu\varepsilon)^{1/4} \tag{3.79}$$

3.5.2 Turbulent premixed flames

Turbulent premixed combustion is the result of the interaction between the turbulent eddies and the premixed flame front. To assess the interaction between both characteristic times are used. The premixed flame has a chemical characteristic time $\tau_c = \delta_L^0/s_L^0$ whereas turbulence can be characterized by two typical time scales: the time scale of the largest eddies $\tau_t = l_t/u'_t$ and smallest eddies $\tau_K = \eta_K/u'_K$. Two dimensionless numbers are used to describe turbulent/flame interactions:

- The Damköhler number Da compares the integral time scale τ_t with the chemical time scale τ_c:

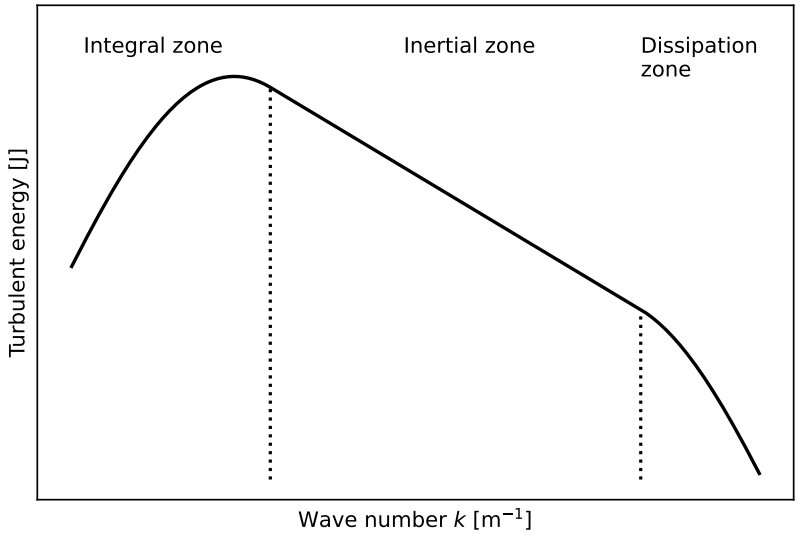


Figure 3.5: Turbulent spectra for homogeneous isotropic turbulence.

$$Da = \frac{\tau_t}{\tau_c} \tag{3.80}$$

For $Da \gg 1$, the time scale of turbulence is much larger than the chemical time step so that turbulence does not affect the flame structure: it is the so-called flamelet regime where the turbulent flame front can be assimilated to a collection of laminar flame elements called *flamelets*. On the other hand, for $Da \ll 1$, the turbulent time scale is much smaller than the chemical time scale so that the reactants and products are mixed by turbulent motion before reacting. This corresponds to the well stirred reactor limit.

- The Karlovitz number Ka compares the smallest turbulence time scale with the chemical time scale:

$$Ka = \frac{\tau_c}{\tau_K} = \frac{\delta_L^0}{\eta_K} \frac{u'_K}{s_L^0} \tag{3.81}$$

The Reynolds, Damköhler and Karlovitz numbers are related through $Re = Da^2 Ka^2$ [Maestro, 2018] so that a diagram of the various combustion regimes can be constructed following Peters [2000] shown in Fig. 3.6.

3.5.3 Turbulent diffusion flames

Differently from premixed flames, diffusion flames do not have intrinsic properties, their structure is governed by the scalar dissipation rate which is directly linked to the turbulent strain rate. Establishing a universal combustion diagram is thus more difficult compared to the premixed flame case.

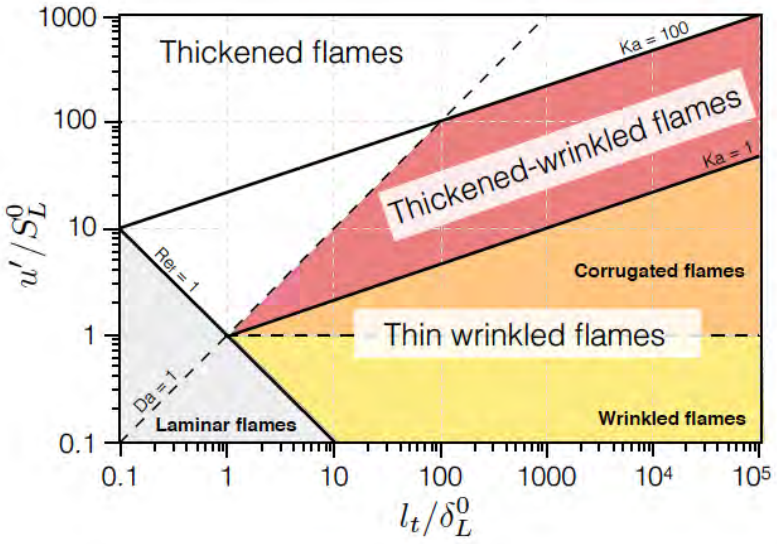


Figure 3.6: Combustion regime diagram for premixed combustion from Peters [2000].

Chapter 4

Plasma Assisted Combustion Modeling

Contents

4.1 NRP discharges

As stated in the introduction, NRP discharges are non-equilibrium discharges which are *thermally* and *statistically* in non-equilibrium. Both parts of this statement are made clear in this section: thermal non-equilibrium is taken into account by computing the electron energy distribution function (EEDF) which is presented in the first section. Excited states, which play a central role in NRP discharges, are then described. This finally leads to explain the meaning of statistical non-equilibrium of excited populations.

4.1.1 Thermal non-equilibrium and EEDF

In NRP discharges, a high-voltage difference is applied between two electrodes for a short amount of time, on the order of tens of nanoseconds. The electric field induced accelerates electrons and collisions with neutrals change the electron distribution function so that it is not a Maxwellian distribution anymore. Electrons are associated to a set of cross-sections each representing a collision process. We will now detail the governing equations of the electron energy distribution function (EEDF). Starting from the Boltzmann equation for the electron distribution f_e:

$$\frac{\partial f_e}{\partial t} + \mathbf{v}_e \cdot \frac{\partial f}{\partial \mathbf{r}} + \mathbf{a}_e \cdot \frac{\partial f}{\partial \mathbf{v}_e} = \left(\frac{\delta f_e}{\delta t}\right)_c \tag{4.1}$$

4.1.1.1 Two-term approximation

We assume a two-term expansion (isotropic and anisotropic part) of the electron distribution function [Raimbault, 2018, Chap. 1]:

$$f_e(\mathbf{r}, \mathbf{v}_e, t) = f_0(\mathbf{r}, v_e, t) + \frac{\mathbf{v}_e}{v_e} \cdot \mathbf{f}_1(\mathbf{r}, v_e, t) \tag{4.2}$$

Reinjecting Eq. (4.2) into Eq. (4.1) and integrating over the solid angle the first and second moments of Boltzmann's equation yields the following coupled equations for the isotropic f_0 and anisotropic $\mathbf{f}_1$ parts of the electron distribution function:

$$\frac{\partial f_0}{\partial t} + \nabla_r \cdot \left(\frac{v_e}{3}\mathbf{f}_1\right) - \frac{1}{v_e^2}\frac{\partial}{\partial v_e}\left(v_e^2 \frac{e\mathbf{E}}{3m} \cdot \mathbf{f}_1\right) = \left(\frac{\delta f_0}{\delta t}\right)_c \tag{4.3}$$

$$\frac{\partial \mathbf{f}_1}{\partial t} + \nabla_r (v_e f_0) - \frac{\partial}{\partial v_e}\left(\frac{e\mathbf{E}}{m} f_0\right) = \left(\frac{\delta \mathbf{f}_1}{\delta t}\right)_c \tag{4.4}$$

Three types of processes are dominant for electrons: elastic collisions with neutrals whose mass M are way larger than that of electrons m. Electrons can change the internal structure of the neutral particules so that excitation and ionization are possible after a collision at sufficient speed. Starting from Eq. (2.31), elastic processes yield the following form for the collision parts [Raimbault, 2018, Chap. 2]:

$$\left(\frac{\delta \mathbf{f}_1}{\delta t}\right)_c^{\text{el}} = -\nu_m\left(v_e\right)\mathbf{f}_1(\mathbf{r}, v_e, t) \tag{4.5}$$

$$\left(\frac{\delta f_0}{\delta t}\right)_c^{\text{el}} = \frac{m}{M}\frac{1}{v_e^2}\frac{\partial}{\partial v_e}\left[v_e^2 \nu_m\left(v_e\right)\left(v_e f_0 + \frac{kT_n}{m}\frac{\partial f_0}{\partial v_e}\right)\right] \tag{4.6}$$

where $\nu_m = n_n \sigma_{en}^t$ is the momentum frequency collision, σ_{en}^t the momentum cross-section, n_n the neutral density, T_n the neutral temperature. Inelastic collisions are derived the same way starting from Eq. (2.31):

$$\left(\frac{\delta f_0}{\delta t}\right)_c^{\text{exc}} = \frac{v_e'}{v_e}\nu_{\text{exc}}(v_e') f_0(\mathbf{r}, v_e', t) - \nu_{\text{exc}}\left(v_e\right) f_0(\mathbf{r}, v_e, t) \tag{4.7}$$

$$\left(\frac{\delta f_0}{\delta t}\right)_c^{\text{ion}} = 4\frac{v_e'}{v_e}\nu_{\text{ion}}\left(v_e'\right) f_0(\mathbf{r}, v_e', t) - \nu_{\text{ion}}\left(v_e\right) f_0(\mathbf{r}, v_e, t) \tag{4.8}$$

where ν_{ion} and ν_{exc} are respectively the ionization and excitation collision frequencies. The practical resolution of Eqs. (4.3) and (4.4) requires additional simplifications [Hagelaar and Pitchford, 2005]: the electric field and the collision terms are uniform, the anisotropic part of the distribution function is directed along the electric field. Performing a change of variable to go to electron volt energy space ε:

$$\mathbf{E} = E\mathbf{e}_z, \quad \mathbf{f}_1 = f_1\mathbf{e}_z$$
$$\varepsilon = \frac{1}{2}mv^2/e = \left(\frac{v}{\gamma}\right)^2 \quad \text{with} \quad \gamma = \sqrt{\frac{2e}{m_e}}$$
$$f(v, \cos\theta, z, t) = f_0(v, z, t) + f_1(v, z, t)\cos\theta$$

Eqs. (4.3) and (4.4) become [Hagelaar and Pitchford, 2005]

$$\frac{\partial f_0}{\partial t} + \frac{\gamma}{3}\varepsilon^{1/2}\frac{\partial f_1}{\partial z} - \frac{\gamma}{3}\varepsilon^{-1/2}\frac{\partial}{\partial \varepsilon}(\varepsilon E f_1) = C_0 \tag{4.9}$$
$$\frac{\partial f_1}{\partial t} + \gamma\varepsilon^{1/2}\frac{\partial f_0}{\partial z} - E\gamma\varepsilon^{1/2}\frac{\partial f_0}{\partial \varepsilon} = -N\sigma_m\gamma\varepsilon^{1/2}f_1 \tag{4.10}$$

where $C_0 = (\delta f_0/\delta t)_c$ represents the change in f_0 due to collisions. The right hand side of Eq. (4.10) contains the total gas density N and the total momentum-transfer cross-section σ_m consisting of contributions from all possible collision processes k with gas particles:

$$\sigma_m = \sum_k x_k\sigma_k \tag{4.11}$$

where x_k is the mole fraction of the target species of the collision process. For elastic collisions, σ_k is the effective momentum transfer (same as σ_{en}^t above). For inelastic collisions, σ_k is the total cross section.

We assume furthermore a decoupling of energy and time/space dependences for both f_0 and f_1:

$$f_{0,1}(\varepsilon, z, t) = \frac{1}{2\pi\gamma^3}F_{0,1}(\varepsilon)n(z, t) \tag{4.12}$$

where F_0 is the *electron energy distribution function*. The EEDF F_0 is moreover normalized following:

$$\int_0^\infty \varepsilon^{1/2}F_0\,\mathrm{d}\varepsilon = 1 \tag{4.13}$$

4.1.1.2 Growth of the electron density

Three different assumptions can be made regarding the dependence of the electron density. In all cases the Eq.4.9 reduces to:

$$-\frac{\gamma}{3}\frac{\partial}{\partial\varepsilon}\left(\left(\frac{E}{N}\right)^2\frac{\varepsilon}{\tilde{\sigma}_\mathrm{m}}\frac{\partial F_0}{\partial\varepsilon}\right) = \tilde{C}_0 + \tilde{R} \tag{4.14}$$

where $\tilde{R}$ is the growth-renormalization term, it ensures that F_0 remain normalized in the case of net electron production.

- No growth model: the electron density is supposed to be independent of time and

space.

$$\tilde{\sigma}_{\mathrm{m}} = \sigma_{\mathrm{m}} \quad \text{and} \quad \tilde{R} = 0 \tag{4.15}$$

- Temporal growth: the electron density is supposed to be independent of space.

$$\tilde{\sigma}_{\mathrm{m}} = \sigma_{\mathrm{m}} + \frac{\bar{\nu}_i}{N\gamma\varepsilon^{1/2}} \quad \text{with} \quad \frac{1}{n_{\mathrm{e}}}\frac{\partial n_{\mathrm{e}}}{\partial t} = \bar{\nu}_i \tag{4.16}$$

$$\tilde{R} = -\frac{\bar{\nu}_i}{N}\varepsilon^{1/2}F_0 \tag{4.17}$$

- Spatial growth: the electron density is supposed to be independent of time.

$$\tilde{R} = \frac{\alpha}{N}\frac{\gamma}{3}\left[\frac{\varepsilon}{\sigma_{\mathrm{m}}}\left(2\frac{E}{N}\frac{\partial F_0}{\partial\varepsilon} + \frac{\alpha}{N}F_0\right) + \frac{E}{N}F_0\frac{\partial}{\partial\varepsilon}\left(\frac{\varepsilon}{\sigma_{\mathrm{m}}}\right)\right] \quad \text{with} \quad \alpha \equiv -\frac{1}{n}\frac{\partial n}{\partial z} = -\frac{\bar{\nu}_i}{w} \tag{4.18}$$

$$\tilde{\sigma}_{\mathrm{m}} = \sigma_{\mathrm{m}} \tag{4.19}$$

4.1.1.3 Collision terms

The collisions are separated between between neutral-electron (subscript k for the neutral k) and coulomb collisions (subscript e):

$$\tilde{C}_0 = \sum_k \tilde{C}_{0,k} + \tilde{C}_{0,e} \tag{4.20}$$

Elastic collisions The simplifying assumptions are applied to Eq. (4.6) so that the collision term in BOLSIG+ formulation reads

$$\tilde{C}_{0,k=\text{ elastic}} = \gamma x_k \frac{2m}{M_k}\frac{\partial}{\partial\varepsilon}\left[\varepsilon^2\sigma_k\left(F_0 + \frac{k_{\mathrm{B}}T}{e}\frac{\partial F_0}{\partial\varepsilon}\right)\right] \tag{4.21}$$

Inelastic collisions In the same way as for the elastic collisions, Eqs. (4.7) and (4.8) are adapted to the BOLSIG+ formulation to yield

$$\begin{aligned}\tilde{C}_{0,k=\text{ inelastic}} = -\gamma x_k \big[&\varepsilon\sigma_k(\varepsilon)F_0(\varepsilon) \\ &- (\varepsilon + u_k)\,\sigma_k\,(\varepsilon + u_k)\,F_0\,(\varepsilon + u_k)\big]\end{aligned} \tag{4.22}$$

$$\begin{aligned}\tilde{C}_{0,k=\text{ ionization}} = -\gamma x_k \big[&\varepsilon\sigma_k(\varepsilon)F_0(\varepsilon) \\ &-2\,(2\varepsilon + u_k)\,\sigma_k\,(2\varepsilon + u_k)\,F_0\,(2\varepsilon + u_k)\big]\end{aligned} \tag{4.23}$$

$$\tilde{C}_{0,k=\text{ attachment}} = -\gamma x_k \varepsilon\sigma_k(\varepsilon)F_0(\varepsilon) \tag{4.24}$$

Electron-electron collisions A detailed account of Coulomb collisions and the derivation of the corresponding cross-section can be found in [Bittencourt, 2004, Chap. 21]. This yield in the end the following term for electron-electron Coulomb collisions:

$$\tilde{C}_{0,e} = a\frac{n}{N}\frac{\partial}{\partial\varepsilon}\left[3A_1F_0 + 2\left(A_2 + \varepsilon^{3/2}A_3\right)\frac{\partial F_0}{\partial\varepsilon}\right] \tag{4.25}$$

4.1.1.4 Equation for the EEDF

In the end the actual solved equation for the EEDF is a stationary drift-diffusion equation in energy, which reads

$$\frac{\partial}{\partial\varepsilon}\left(\tilde{W}F_0 - \tilde{D}\frac{\partial F_0}{\partial\varepsilon}\right) = \tilde{S} + \tilde{R} \tag{4.26}$$

where

$$\begin{aligned}
\tilde{W} &= -\gamma\varepsilon^2\sigma_\varepsilon - 3a\frac{n}{N}A_1 \\
\tilde{D} &= \frac{\gamma}{3}\left(\frac{E}{N}\right)^2\frac{\varepsilon}{\tilde{\sigma}_\mathrm{m}} + \frac{\gamma k_\mathrm{B}T}{e}\varepsilon^2\sigma_\varepsilon + 2a\frac{n}{N}\left(A_2 + \varepsilon^{3/2}A_3\right) \\
\sigma_\varepsilon &= \sum_{k=\text{elastic}}\frac{2m}{M_k}x_k\sigma_k \\
\tilde{S} &= \sum_{k=\text{inelastic}}\tilde{C}_{0,k}
\end{aligned}$$

4.1.2 Excited states

Excited states of molecules and atoms play a major role in plasma assisted combustion. Recalling Fig. 1.2, the kinetic pathway of plasma assisted combustion relies heavily on the creation of excited species which in turn produce heat and radicals. The different possible excitations of molecules are recalled: rotational excitation, vibrational excitation and electronic excitation where for atoms only electronic excitation is possible. Simple canonical models for rotational and vibration are laid out for diatomic molecules to build some intuition about them. Reminders on quantum mechanics and statistical physics can be found in Appendix B.

4.1.2.1 Rotational excitation - the rigid rotor

A diatomic molecule can be represented by a system of two atoms of mass M separated by a distance R linked by a potential. In classical mechanics, if the interatomic distance R is at an equilibrium position, the molecule has rotation energy:

$$E_\mathrm{rot} = \frac{L^2}{2I} \tag{4.27}$$

where $I = MR^2/2$ is the moment of intertia of the system and L is the angular momentum with respect to the center of gravity. We consider for quantum mechanics the angular momentum operator $\hat{L}$ to deduce the energy levels and eigenfunctions of rotational levels.

Energy levels and eigenfunctions The spherical harmonics are the eigenfunctions of the observable $\hat{L}^2$ associated to the eigenvalue $J(J+1)\hbar^2$ with degeneracy $g_J = 2J+1$:

$$\hat{L}^2 Y_{J,m}(\theta,\phi) = J(J+1)\hbar^2 Y_{J,m}(\theta,\phi) \tag{4.28}$$

Hence the energy levels are retrieved as

$$E_{\text{rot}}(J) = J(J+1)\frac{h^2}{8\pi^2 I} = J(J+1)k_B\theta_r \tag{4.29}$$

where I is the momentum of inertia of the rotor (considered as a constant), k is the Boltzmann constant, J is the rotational quantum number and

$$\theta_r = \frac{h^2}{8\pi^2 I k_B} \tag{4.30}$$

is the characteristic rotational temperature.

Partition function The statistical weight (the degeneracy) of a rotational level is $g_J = 2J+1$ so that the rotational partition function writes

$$\mathcal{Q}^{\text{rot}} = \sum_{J=0}^{\infty}(2J+1)\mathrm{e}^{-J(J+1)\theta_r/T} \tag{4.31}$$

The rotational characteristic temperature of diatomic species is usually small, and therefore, for $T \gg \theta_r$, the sum can be approximated by an integral yielding

$$\mathcal{Q}^{\text{rot}} \approx \int_0^{\infty}(2J+1)\mathrm{e}^{-J(J+1)\theta_r/T}\mathrm{d}J = \int_0^{\infty}\frac{T}{\theta_r}\frac{\mathrm{d}}{\mathrm{d}J}\mathrm{e}^{-J(J+1)\theta_r/T}\mathrm{d}J = \frac{T}{\theta_r} \tag{4.32}$$

This approach has to be slightly changed for homo-nuclear diatomic molecule due to the Pauli's exclusion principle to yield where a factor 2 appear in the denominator so that in the end

$$\mathcal{Q}^{\text{rot}} = \frac{T}{\sigma\theta_r} \tag{4.33}$$

with $\sigma = 1$ for heteronuclear molecules and $\sigma = 2$ for homonuclear molecules.

4.1.2.2 Vibrational excitation

For a diatomic molecule the two atoms can move relative to each other around their equilibrium position. Quantum description of such movement yields the vibrational levels of such molecules.

Energy levels and eigenfunctions A harmonic oscillator is a system which consists of a particle of mass m elastically connected to a center x_0 by a restoring force $F = -K(x - x_0)$. The reference position and potential are chosen such that $x_0 = 0$ and $V_0 = 0$. Hence total energy is given by:

$$E = \frac{1}{2}m\dot{x}^2 + \frac{1}{2}m\omega^2 x^2 \tag{4.34}$$

This energy is always positive and since the potential tends to infinity when $|x| \to \infty$, there are only bound states. In quantum mechanics, the Hamiltonian has the following form:

$$\hat{H} = \frac{\hat{p}_x^2}{2m} + \frac{1}{2}m\omega^2\hat{x}^2, \tag{4.35}$$

We solve the stationary Schrödinger equation to find the eigenvalues and eigenfunctions of our problem:

$$\left(-\frac{\hbar^2}{2m}\frac{d^2}{dx^2}+\frac{1}{2}m\omega^2x^2\right)\psi(x)=E\psi(x). \tag{4.36}$$

We are only interested in square-integrable functions since there are only bound states. Natural units appear in this problem (the length $a=\sqrt{\hbar/(mw)}$ and energy $\hbar\omega$) which leads to the following change of variables:

$$\epsilon=\frac{E}{\hbar\omega}\quad y=\frac{x}{a} \tag{4.37}$$

and the time-independent Schrödinger's equation becomes:

$$\frac{1}{2}\left(y^2-\frac{d^2}{dy^2}\right)\phi(y)=\epsilon\phi(y) \tag{4.38}$$

Solutions to this differential equation are Hermite functions:

$$\phi_v(y)=c_v\mathrm{e}^{-y^2/2}H_v(y)\quad\text{with}\quad\begin{cases}c_v=(\sqrt{\pi}\,2^n\,n!)^{-1/2}\\ H_v(y)=(-1)^n e^{y^2}\frac{d^n}{dy^n}\left(e^{-y^2}\right)\end{cases} \tag{4.39}$$

with the corresponding eigenvalues:

$$\epsilon_v=n+\frac{1}{2} \tag{4.40}$$

Finally the dimensional solutions are

$$\psi_v(x)=\frac{\pi^{-1/4}}{\sqrt{2^v\,v!\,a}}\mathrm{e}^{-x^2/2a^2}H_v(x/a) \tag{4.41}$$

$$E_v=\left(v+\frac{1}{2}\right)\hbar\omega \tag{4.42}$$

Partition function The eigenfunctions have no degeneracy and from Eq. (4.42) the partition function writes

$$Q^{\mathrm{vib}}=\left(\mathrm{e}^{\theta_\nu/2T}-\mathrm{e}^{-\theta_\nu/2T}\right)^{-1}=\frac{\mathrm{e}^{-\theta_\nu/2T}}{1-\mathrm{e}^{-\theta_\nu/T}} \tag{4.43}$$

where $\theta_\nu=\frac{h\nu}{k_B}$ is the characteristic vibrational temperature. The internal energy is retrieved from the partition function as

$$U^{\mathrm{vib}}=\frac{1}{2}\mathcal{N}h\nu+\frac{\mathcal{N}h\nu}{\mathrm{e}^{h\nu/kT}-1} \tag{4.44}$$

It should be noted that energy of vibrational levels included in the partition function are referred to the bottom of the potential curve. On the other hand, we can refer the energy level to the ground state ($v=0$), obtaining the new partition function and vibrational internal

energy:

$$\mathcal{Q}^{\text{vib}} = (1 - \mathrm{e}^{-h\nu/kT})^{-1} \tag{4.45}$$

$$U^{\text{vib}} = \frac{\mathcal{N} h\nu}{\mathrm{e}^{h\nu/kT} - 1} \tag{4.46}$$

Towards more realistic vibrational models The harmonic oscillator model is a good approximation near the equilibrium position of the diatomic molecule but more complex potential shapes are needed when considering higher energy levels. A better approximation is given by the Morse potential Capitelli et al. [2012]

$$V(r) = D_e[1 - \exp\{-\beta(r - r_e)\}]^2 \tag{4.47}$$

where D_e is the depth of the potential well and r_e is the equilibrium position. The energy levels and eigenfunctions of the vibrational levels of nitrogen are given in Fig. 4.1. Near the bottom of the potential curve the eigenfunctions and energy levels are quite similar but the higher the energy the larger the difference between the two models.

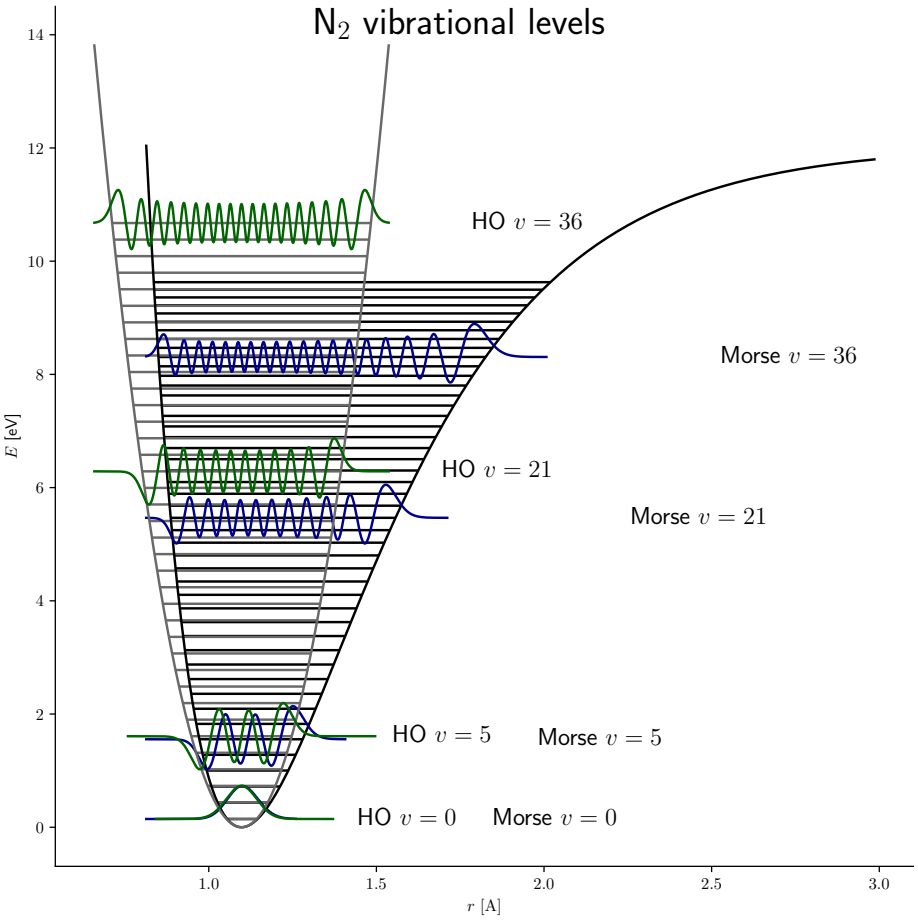


Figure 4.1: Ground state N_2 vibrational levels for Harmonic Oscillator and Morse potentials.

4.1.2.3 Electronic excitation

Electron excitation is the transfer of a bound electron to a more energetic, but still bound state. For diatomic molecules, each electronically excited state is composed of vibrational states which are in turn composed of rotational states. This is illustrated for N_2 in Fig. 4.2. In plasma discharges, this electron excitation is the result of electrons colliding with neutral at high velocities induced by the electric field. These electronically excited states are very reactive and can relax to the ground states through several mechanisms: photon emission or collisional quenching. The latter will be of central importance for plasma assisted combustion using NRP discharges.

4.1.3 Statistical non-equilibrium

In classical combustion chemistry, the excited species populations are in a state of statistical equilibrium described by Eq. (B.32). In NRP discharges, the out-of-thermal equilibrium electrons collide with neutral particles and are able to excite them in proportions far above their equilibrium values: on top of thermal non-equilibrium we also have therefore *statistical non-equilibrium*. By using the simple rigid rotor and harmonic oscillator for a diatomic molecule the thermodynamic properties of the ideal diatomic gas can be retrieved [Kee et al., 2017, Chap. 10]:

$$H = \frac{7}{2}Nk_BT + Nk_B\theta_v\frac{\exp(-\theta_v/T)}{1-\exp(-\theta_v/T)} - ND_0 \tag{4.48}$$

$$C_p = \left(\frac{\partial H}{\partial T}\right)_P = \frac{7}{2}Nk_B + Nk_B\frac{\theta_v^2}{T^2}\frac{e^{-\theta_v/T}}{(1-e^{-\theta_v/T})^2} \tag{4.49}$$

$$S = -\left(\frac{\partial F}{\partial T}\right)_{V,N} = Nk_B\left[\ln\frac{V}{N\lambda_T^3} + \frac{7}{2} + \frac{\theta_v}{T}\frac{e^{-\theta_v/T}}{1-e^{-\theta_v/T}} - \ln\left(1-e^{-\theta_v/T}\right) + \ln\left(\frac{T}{\sigma\theta_r}\right)\right]. \tag{4.50}$$

where D_0 is the dissociation energy of the ground state of the diatomic molecule. The activation of the vibration levels of a diatomic molecule are seen in the second term of C_p where it transitions from $7/2Nk_B$ tp $9/2Nk_B$. Hence the statistical equilibrium assumption is central in the thermodynamic description of gas mixtures and special care must be taken when the excited species are in substantial amounts as in NRP discharges.

4.2 Governing equations in AVIP

AVIP is a non-structured code written in Fortran for the numerical simulation of weakly-ionized plasmas. It is based on the same structure as the combustion code AVBP and has been developed at CERFACS since 2015.

A first version of the code was introduced in Joncquieres [2019] to simulate Hall-Effect (HE) thrusters plasmas. During this PhD, the code has been rewritten to take into account streamer simulations for PAC, leveraging object-oriented paradigms in Fortran for integration of different types of plasma modeling. Hence AVIP is now a non-structured code for fluid

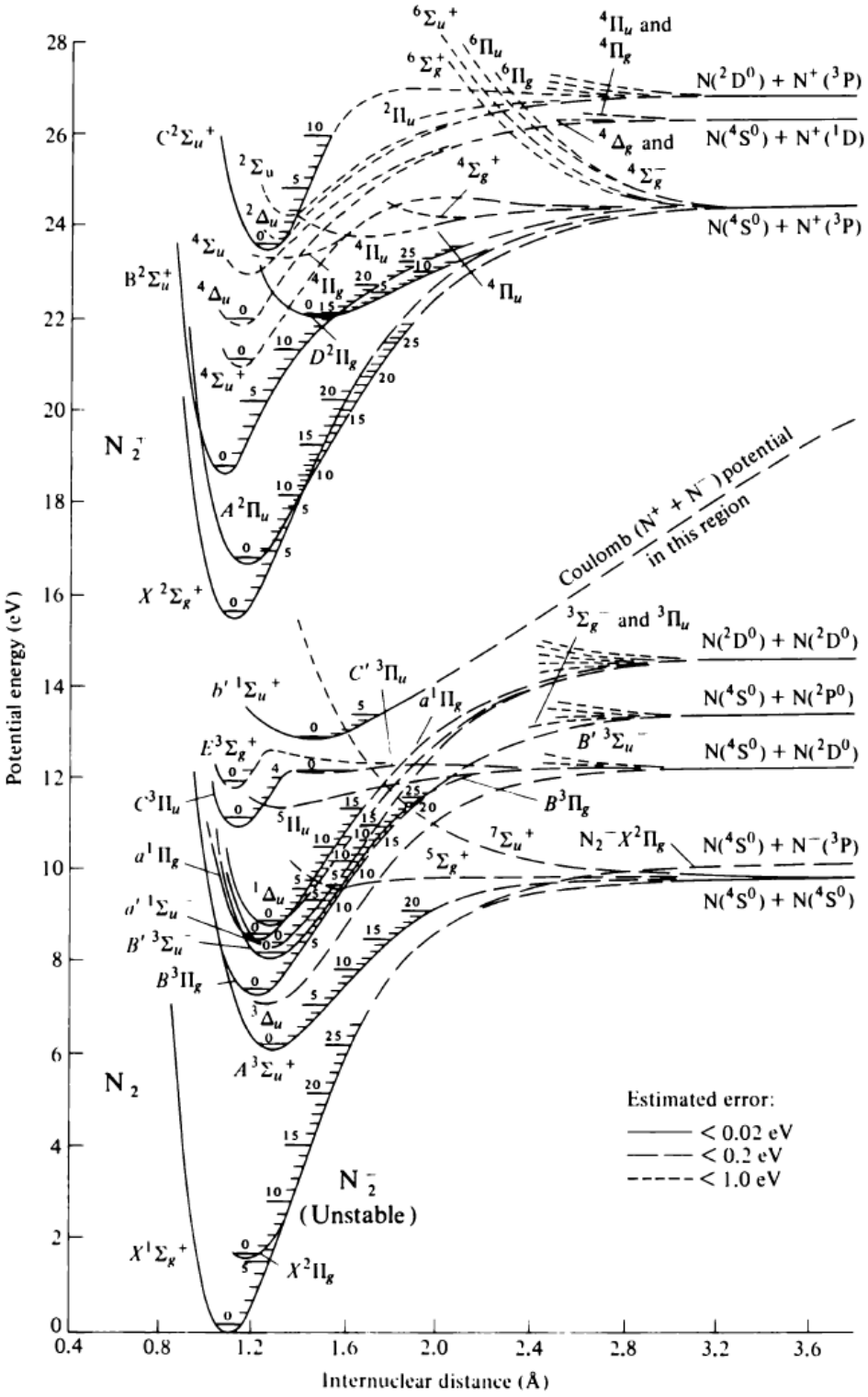


Figure 4.2: Quantum levels of N_2 taken from Steinfeld [1986].

simulations of both Nanosecond Repetitively Pulsed (NRP) discharges in Plasma Assisted Combustion (PAC) and Hall-Effect (HE) thrusters.

Governing equations from previous chapters are recalled with the two main fluid formulations for simulations of weakly ionized plasmas: plasma Euler equations [Bittencourt, 2004] and plasma drift-diffusion equations [Raimbault, 2018]. The plasma species equations are coupled to the previous EEDF and to the Poisson equation for computation of the eletric field. These equations are supplemented by the reactive Navier-Stokes equations for the mixture which will be relevant for PAC.

4.2.1 Electromagnetism

In plasma simulations, the electric $\mathbf{E}$ and magnetic $\mathbf{B}$ fields appear in the transport equations and need to be computed. Magnetic field $\mathbf{B}$ is only used in HE thrusters simulation where it is constant and is negligible for NRP discharges simulations so that the electric field can be computed from the electromagnetic potential $\mathbf{E} = -\nabla\phi$. The electromagnetic potential is governed by the Poisson equation supplemented by boundary conditions [Jackson, 1999]:

$$\nabla^2\phi = -\frac{\rho}{\varepsilon_0} \quad \text{in} \quad \dot{\Omega} \tag{4.51}$$

$$\phi = \phi_D \quad \text{on} \quad \partial\Omega_D \tag{4.52}$$

$$\nabla\phi \cdot \mathbf{n} = -E_n \quad \text{on} \quad \partial\Omega_N \tag{4.53}$$

where $\dot{\Omega}$ denotes the interior of the domain, $\partial\Omega_D$ the Dirichlet boundary, ϕ_D its imposed potential, $\partial\Omega_N$ the Neumann boundary and E_n its associated impsoed normal electric field.

In a numerical simulation (4.51) becomes a linear system $A\ \phi = b$ and a linear system solver must be used to get the potential.

Some models such as the ambipolar diffusion give the electric field from different parameters and also need to be incorporated in the code [Celestin, 2008].

4.2.2 Transport equations

Depending on the level of precision wanted, each charged species i has its own set of transport equations. Three kind of transport equations are wanted, with decreasing precision: plasma Euler equations, drift-energy and drift-diffusion equations [Raimbault, 2018]. For plasma assisted combustion the gaseous flow also needs to be solved.

4.2.2.1 Plasma Euler equations

The plasma species can be described using Euler equations presented in Chap. 2:

$$\frac{\partial \mathbf{U}_i}{\partial t} + \nabla \cdot \mathbf{F}_i = \mathbf{S}_i \tag{4.54}$$

with

$$\mathbf{U}_i = \begin{bmatrix} \rho_i \\ \rho_i \mathbf{V}_i \\ \rho_i E_i \end{bmatrix} \qquad \mathbf{F}_i = \begin{bmatrix} \rho_i \mathbf{V}_i \\ \rho_i \mathbf{V}_i \mathbf{V}_i + P_i \mathbf{I} \\ (\rho_i E_i + P_i)\mathbf{V}_i \end{bmatrix} \qquad \mathbf{S}_i = \begin{bmatrix} S_{0i} \\ \mathbf{S}_{1i} \\ S_{2i} \end{bmatrix} \tag{4.55}$$

and

$$S_{0i} = \int m_i \left(\frac{\delta f_i}{\delta t} \right)_c \mathrm{d}^3 \mathbf{v}_i \tag{4.56}$$

$$\mathbf{S}_{1i} = n_i q_i (\mathbf{E} + \mathbf{V}_i \times \mathbf{B}) + \int m_i \mathbf{v}_i \left(\frac{\delta f_i}{\delta t} \right)_c \mathrm{d}^3 \mathbf{v}_i \tag{4.57}$$

$$S_{2i} = n_i q_i \mathbf{E} \cdot \mathbf{V}_i + \int \frac{m_i v_i^2}{2} \left(\frac{\delta f_i}{\delta t} \right)_c \mathrm{d}^3 \mathbf{v}_i \tag{4.58}$$

where $\mathbf{V}_i = \langle \mathbf{v}_i \rangle$ is the mean velocity over phase space. These sets of equations are relevant in Hall-effect thrusters modeling Joncquieres [2019] and are not further discussed.

4.2.2.2 Drift-diffusion and drift-energy equations

For the modeling of plasma discharges the simplified version of the plasma Euler equations neglecting inertia forces, the drift-diffusion equations, are considered:

$$\frac{\partial n_i}{\partial t} + \nabla \cdot \mathbf{\Gamma}_i = S_{0i} \qquad \text{with } \mathbf{\Gamma}_i = n_i \mu_i \mathbf{E} - D_i \nabla n_i \tag{4.59}$$

The energy equation can be added and simplified in a similar manner yielding the drift-energy model which is the above equation supplemented by [Raimbault, 2018, Chap. 3]:

$$\frac{\partial n_i \varepsilon_i}{\partial t} + \nabla \cdot \mathbf{H}_i = S_{2i} \qquad \text{with } \mathbf{H}_i = n_i \varepsilon_i \beta_i \mathbf{E} - G_i \nabla n_i \varepsilon_i \tag{4.60}$$

Inclusion of the non-equilibrium electrons mean that the drift-diffusion equations are coupled with the EEDF equations which must be solved.

4.2.2.3 Gas mixture of neutral particules

The reactive Navier Stokes equation are solved for a mixture of N species:

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \nabla \cdot [\rho \mathbf{u}] = 0 \\
&\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot [\rho \mathbf{u}\mathbf{u} + p\mathbf{I} - \boldsymbol{\tau}] = \rho \sum_{k=1}^{N} Y_k \mathbf{f}_k \\
&\frac{\partial \rho E}{\partial t} + \nabla \cdot [(\rho E + p)\mathbf{u} - \boldsymbol{\tau} \cdot \mathbf{u} + \mathbf{q}] = \dot{\omega}_T + \dot{Q} + \rho \sum_{k=1}^{N} Y_k \mathbf{f}_k \cdot (\mathbf{u} + \mathbf{V}_k) \\
&\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot [\rho Y_k (\mathbf{u} + \mathbf{V}_k)] = \dot{\omega}_k
\end{aligned} \tag{4.61}$$

The gas mixture and drift-diffusion equations are coupled through the chemistry: $\dot{\omega_k}$ in the NS equations and S_{0i} in the drift-diffusion equations are taken from a common chemistry. More details are given in Chap. 11 when the fully coupled PAC simulations will be presented.

The above set of equations (plasma and reactive flow) are solved with specific numerical methods that are presented in the next chapters.

Part II

Numerical Methods for PAC

Chapter 5

Introduction to Numerical Methods

Contents

In this thesis, two types of equations are solved numerically: elliptic equations (the Poisson equation) and hyperbolic equations (the drift-diffusion and reactive Navier-Stokes equations). This chapter provides general definitions about numerical schemes and discretizations that are common to the next three chapters. The concepts of consistency, stability and convergence are recalled in a way that is correct for all the types of equations studied. Then general definitions about numerical methods are recalled and the choices made in AVIP are detailed. Some toy cases and coding of fundamental algorithms relevant to all the numerical developments of AVIP can be found at `https://github.com/lionelchg/NumMaths`.

A publication related to the AVIP code that summarizes the main features of the code presented in the subsequent chapters can be found on ArXiV: Cheng et al. [2021]. It is meant to be updated with any major development of the code.

5.1 Fundamentals of Numerical Methods and Numerical Mathematics

5.1.1 Consistency, stability and convergence

The general definitions of consistency, stability and convergence shown here are taken from [Quarteroni et al., 2007, Chap. 2]. Let us consider the problem to find x such that

$$F(x, d) = 0 \tag{5.1}$$

where d is the set of data on which the solution depends and F is the functional relationship between x and d. We restrict the study to direct problems where F and d are given and x is the unkwown. In the case of the Poisson equation d is the charge density and F the Laplacian whereas for hyperbolic problems F is a non-linear differential operator and d the flow parameters.

Definition 5.1 (Well-posedness or stability of a problem)**.** *The problem is well-posed if it admits a unique solution which depends continuously on the data. Continuous dependence on the data means that small perturbations of d leads to small perturbations in the solution x. For a given δd, the subsequent change δx such that*

$$F(x + \delta x, d + \delta d) = 0 \tag{5.2}$$

satisfies

$$\forall \eta > 0 \ \exists K(\eta, d) \ ||\delta d|| < \eta \implies ||\delta x|| < K(\eta, d)||\delta d|| \tag{5.3}$$

A numerical method for solving a well-posed problem will consist in a sequence of approximate problems

$$F_n(x_n, d_n) = 0 \qquad n \geq 1 \tag{5.4}$$

and the expectation is that $x_n \to x$, $d_n \to x$ and F_n approximates F as $n \to +\infty$. n refers to the discretization of the problem which for elliptic differential equations is only a spatial discretization but for hyperbolic differential equations is a space and time discretization.

Definition 5.2 (Consistency of a numerical method)**.** *The numerical method is said to be consistent if*

$$F_n(x, d) = F_n(x, d) - F(x, d) \to 0 \qquad \text{when } n \to +\infty \tag{5.5}$$

The numerical method is said to be strongly consistent if $F_n(x, d) = 0$ for any value of n.

Definition 5.3 (Convergence)**.** *The numerical method is said to be convergent if*

$$\forall \varepsilon > 0 \ \exists n_0(\varepsilon) \ \exists \delta(n_0, \varepsilon) > 0 : \tag{5.6}$$

$$\forall n > n_0(\varepsilon) \ \forall ||\delta d_n|| < \delta(n_0, \varepsilon) \implies ||x(d) - x_n(d + \delta d_n)|| < \varepsilon \tag{5.7}$$

where d is an admissible data for the problem, $x(d)$ is the corresponding solution and $x_n(d_n + \delta d_n)$ is the solution of the numerical problem with data $d_n + \delta d_n$.

Convergence is achieved if the solution of the numerical discretization of the problem tends to the actual exact solution of the problem. The link between convergence and stability is found in the famous *Lax-Richtmeyer* theorem:

Theorem 5.1 (Lax-Richtmeyer)**.** *For a consistent numerical method, stability is equivalent to convergence.*

5.1.2 Discretization methods and data structures

5.1.2.1 Finite Difference for structured grids

Finite difference discretization is the most straightforward way of discretizing derivatives using Taylor expansions. It requires however structured grids where each point, in an n-dimensional space, is located at the intersections of n families of lines and must lie on one, and only one line, of each family [Hirsch, 2007, Chap. 4].

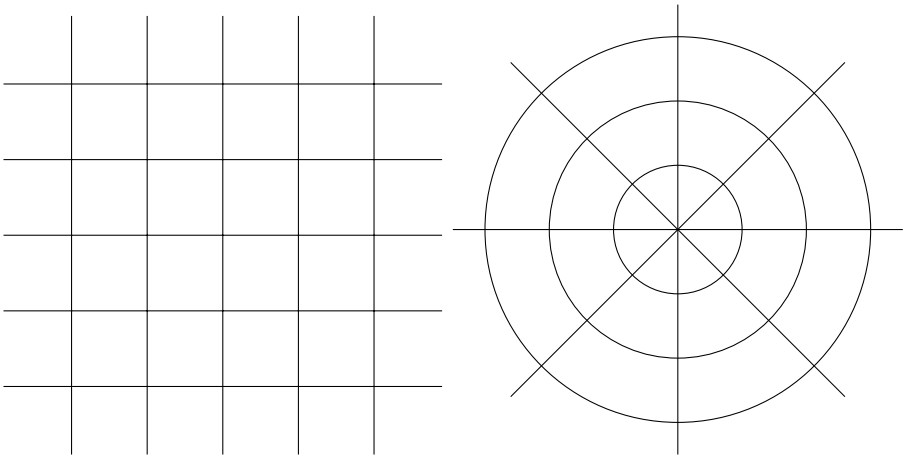


Figure 5.1: Examples of structured grids.

In a structured grid, one index per dimension is used to store the data, *e.g.* ijk in three dimensions, so that access to neighboring points and use of high-order discretizations is easy. Examples of structured grids are given in Fig. 5.1.

5.1.2.2 Finite Volume for structured and unstructured grids

The Finite Volume (FV) Method is applicable on both structured and unstructured grids. This method uses an integral formulation of the partial differential equations of interest and is directly conservative [Lamarque, 2007, Hirsch, 2007].

Unstructured grids work natively on triangles in 2D and tetrahedra in 3D where an example is given in Fig. 5.2. No point can be found at the intersections of families of lines and the one-index per dimension storing of data is thus not possible. Points need to be numbered separately and connectivity tables, *i.e.* the nodes composing each element, are defined. Interaction between distant nodes is more involved and costly in this case, making the development of high-order schemes harder than for finite difference methods.

5.1.2.3 Finite Element for unstructured grids

In the Finite Element Method, the physical variables are expanded using trial functions and the equations are recast in a weak formulation. More mathematically involved, this method produces higher order schemes than the Finite Volume Method but is also more computationally expensive. The Finite Element Method reduces to the Finite Volume Method for P1 trial functions with lumping of the mass matrix [Lamarque, 2007, Chap. 4].

5.2 Numerical Methods for Plasmas and Combustion equations in AVIP

5.2.1 Data structure and type of methods

AVIP shares the same datastructure and metrics as AVBP which are detailed in depth in [Lamarque, 2007, Chap. 4]. Hence unstructured meshes are considered with data stored at

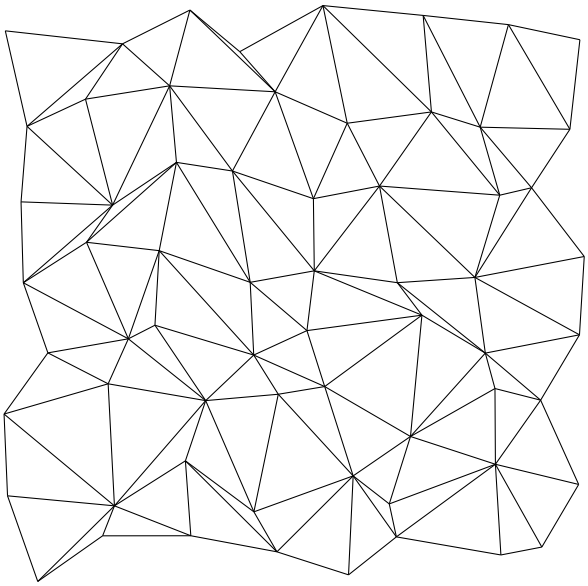


Figure 5.2: Examples of an unstructured grid.

the vertices of the mesh and Finite Volume schemes have been developed. We thus consider the integral formulation of a conservation law in its general form without source terms:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega}\mathbf{U}\,\mathrm{d}V+\int_{\partial\Omega}\mathbf{F}\cdot\mathbf{n}\mathrm{d}S=0 \tag{5.8}$$

The possible control volumes associated to such a discretization is now discussed. There are three choices: cell-centered, vertex-centered or cell-vertex methods. In cell-centered methods, the variables are stored in the cell centers and the flux are applied at the cell faces, hence vertices are only present to mark off the different control volumes. In vertex-centered methods, the variables are stored at the vertices and the conservation laws are applied in the dual volumes of the mesh. Finally a cell-vertex methods used in AVBP computes part of the residuals at the cells before scattering them to the vertices which has been shown to be more robust to mesh deformations [Suli, 1989].

5.2.2 Nomenclature and notations

All the nomenclature and notations used in the following Chaps. 6 and 7 concerning the computational grid or mesh are summarized below:

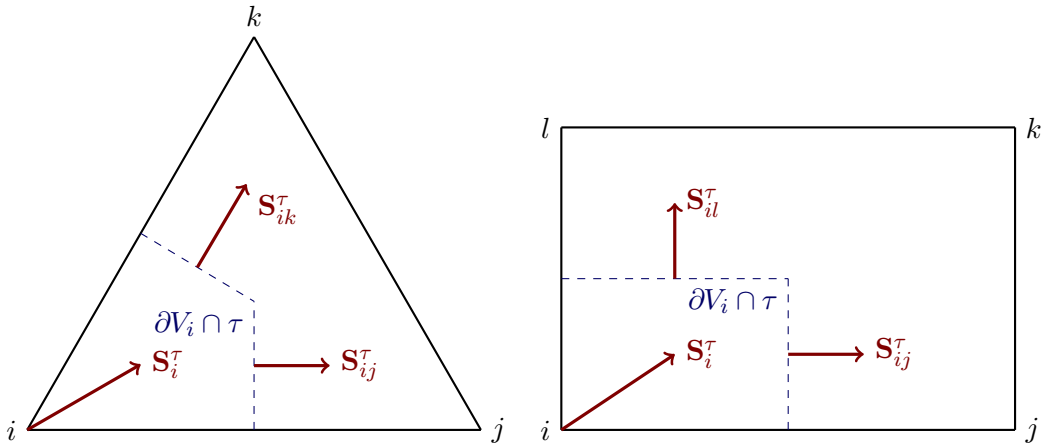
- Ω is the open set which corresponds to the domain of integration. Nomenclature from topology is adopted to indicate the interior of the domain $\mathring{\Omega}$ and the boundary of the domain $\partial\Omega=\overline{\Omega}/\mathring{\Omega}$ where $\overline{\Omega}$ is the closure of Ω
- τ is a cell of the domain, V_τ is the cell volume, n_v^τ the number of vertex of the cell
- i,j,k are nodes of the domain. To each node i is associated a volume V_i, ∂V_i is the boundary of the nodal volume, $\partial\mathring{V}_i=\mathring{\Omega}\cap\partial V_i$ is the interior boundary of the nodal volume, $\partial V_i^b=\partial\Omega\cap\partial V_i$ is the boundary of the nodal volume which is also a boundary of the domain

- $E(i)$ is the set of elements (or cells) for which i is a vertex
- f is a face of a nodal surface or an element
- $\mathbf{S}_{i,j}^{\tau}$ is the surface normal of the edge (i,j) in cell τ which is oriented from i to j. $\mathbf{S}_i^{\tau}$ is the surface associated to node i in cell τ, pointing inward towards the cell center and defined as a linear combination of the adjacent faces:

$$\mathbf{S}_i^{\tau} = -\frac{n_d}{n_v^f} \sum_{f \subset i} \mathbf{S}_f^{\tau} \tag{5.9}$$

  where n_d is the number of dimensions of the computational grid, n_v^f the number of vertices on the face f. The cell dependence τ of the normals is sometimes omitted to alleviate notations.

Individual triangular and quadrangular cells with the different normals defined above are shown in Figures 5.3a and 5.3b. Triangular and quadrangular domains are depicted in Figures 5.4a and 5.4b where nodal volumes are drawn.



(a) Triangular cell (b) Quadrangular cell

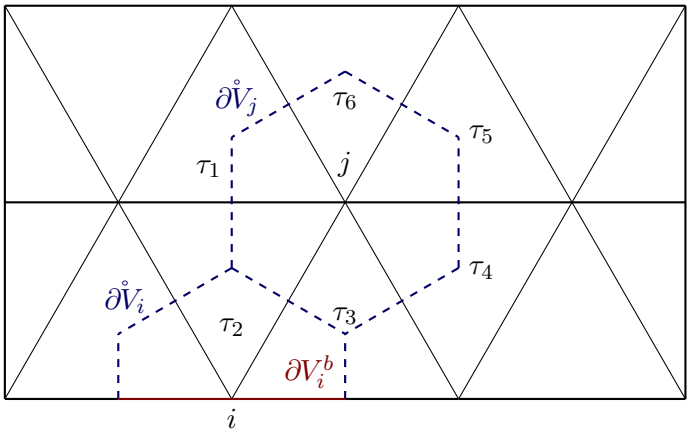
Figure 5.3: Metrics definitions in 2D cells.

Finally for a vector variable $\mathbf{U}(\mathbf{x},t)$ the nodal average, which is the physical quantities that the scheme is working on, is defined as

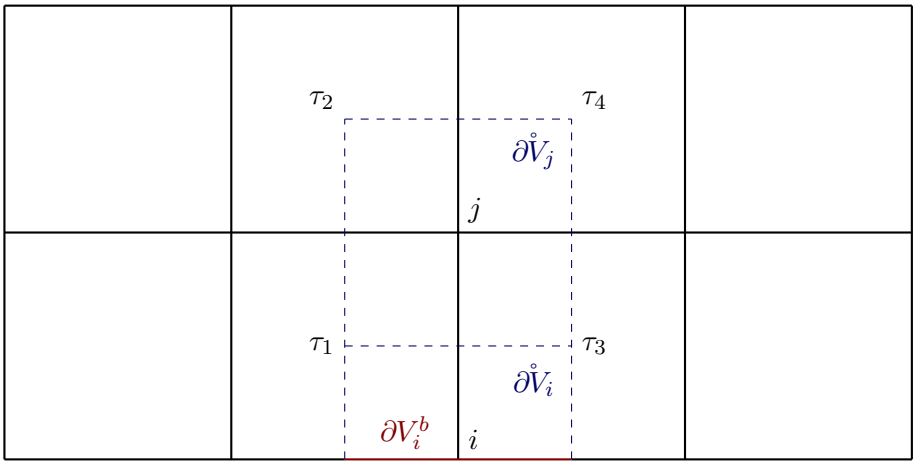
$$\mathbf{U}_i(\mathbf{x},t) = \frac{1}{V_i} \int_{V_i} \mathbf{U}\, dV \tag{5.10}$$

The order of convergence of the different numerical schemes studied throughout the following three chapters are assessed by computing the $\mathcal{L}^2$ norm of the error of the nodal average for a scalar variable u defined by:

$$L^2(\varepsilon_u) = \sqrt{\frac{\sum_i V_i(\tilde{u}_i - u_i)^2}{\sum_i V_i}} \tag{5.11}$$



(a) Triangular elements



(b) Quadrangular elements

Figure 5.4: 2D meshes with nodes i and j belonging respectively to the interior and the boundary of the domain for triangular and quadrangular elements. Neighboring cells belonging to $E(j)$ are shown for both cases.

where $\tilde{u}$ is the exact solution of the problem at hand. This formulation corresponds to Option 1 of Wang et al. [2013] and is the most natural error estimation in the framework of AVIP.

5.2.3 Cylindrical frames

In this work, axisymmetric conditions, *i.e.* $\frac{\partial}{\partial \theta} = 0$, are often considered. This saves a lot computational time as a three-dimensional simulation can be carried out using a two-dimensional mesh. A typical cylindrical frame in the (z, r) plane is shown in Fig. 5.5. In the following numerics chapters, the Poisson and plasma transport equations discretizations in cylindrical coordinates are thus presented along with their classical cartesian coordinates discretizations. Geometrical source terms appear for the Euler and Navier-Stokes equations which require special care for proper integration.

The elements and surfaces considered in this 2D mesh are actually tores in this setting as illustrated in Fig. 5.5 and it should be always kept in mind that although we are solving the equations on a 2D mesh the geometry is in fact three dimensional.

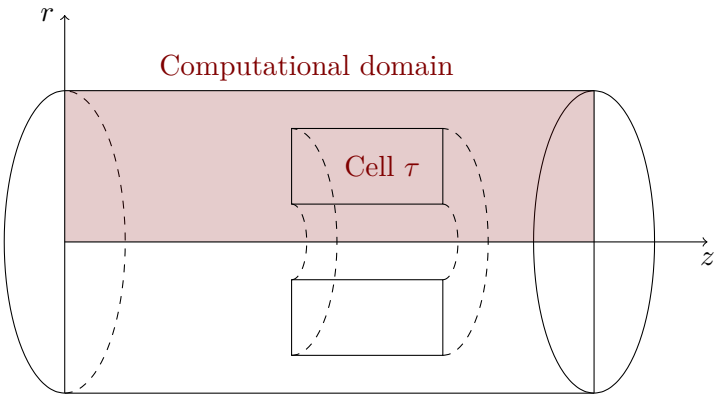


Figure 5.5: Axisymmetric nodal volumes.

Two important results relevant to the finite volume method in cylindrical geometries are the following Pappus-Guldinus theorems illustrated in Figs. 5.6 and 5.7 which deal with volumes and surfaces of revolution [Harris and Stoecker, 1998]. These theorems are used to compute fluxes and volumes in axisymmetric formulations of the Poisson equation and transports equations.

Theorem 5.2 (First Pappus–Guldinus theorem)**.** *The surface area A of a surface of revolution generated by rotating a plane curve C about an axis external to C and on the same plane is equal to the product of the arc length s of C and the distance d traveled by the geometric centroid of C:*

$$A = s \times d \tag{5.12}$$

Theorem 5.3 (Second Pappus–Guldinus theorem)**.** *The volume V of a solid of revolution generated by rotating a plane figure F about an external axis is equal to the product of the area A of F and the distance d traveled by the geometric centroid of F. (Note that the centroid of F is usually different from the centroid of its boundary curve C.) That is:*

$$V = A \times d \tag{5.13}$$

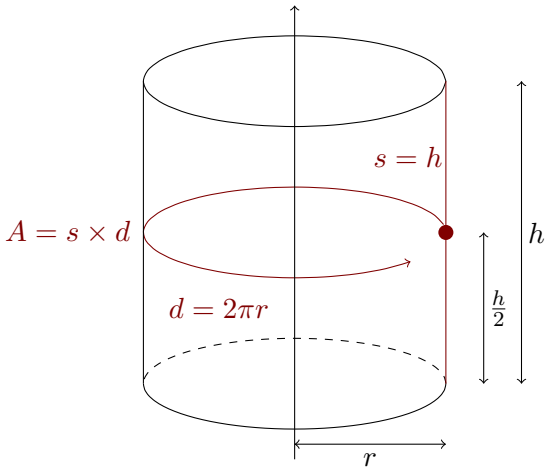


Figure 5.6: Illustration of the first Pappus-Guldinus theorem with a cylinder surface.

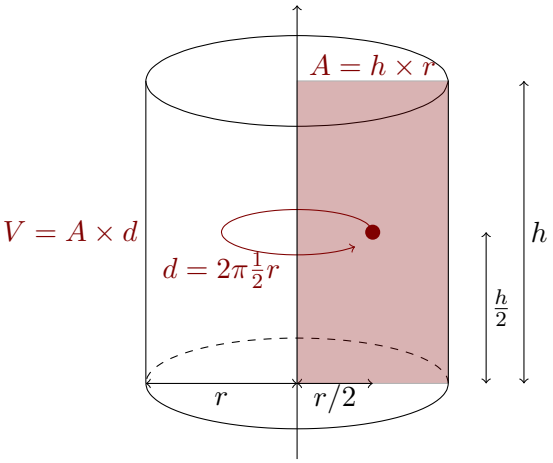


Figure 5.7: Illustration of the first Pappus-Guldinus theorem with a cylinder volume.

Chapter 6

The Poisson equation

Contents

The Poisson equation, necessary to simulate plasmas, is presented in this chapter. The first section is a theoretical presentation of the equation, where some analytical solutions are derived to give insights about the behavior of the equation on canonical geometries. The second section deals with the discretization of the Poisson equation which becomes a linear system on computational grids. In a third section, a brief and concise presentation of linear solvers is given with an emphasis on iterative solvers. Finally test cases to validate the implementation of the Poisson equation in AVIP are detailed in the last section.

6.1 An overview of electrostatics

Most of the material presented in this section is taken from Jackson [1999]. The Poisson equation relates the electromagnetic potential ϕ to the charge distribution ρ:

$$\nabla^2\phi = -\frac{\rho}{\epsilon_0} \tag{6.1}$$

Without source terms $\rho = 0$, the equation reduces to the Laplace equation.

6.1.1 General solution to the free space problem

Without boundary surfaces a general solution of the Poisson equation is given by:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0}\int_{V'}\frac{\rho(\mathbf{x}')}{|\mathbf{x}-\mathbf{x}'|}dV' \tag{6.2}$$

A key equality for the proof is the following relationship which will be relevant in the following sections:

$$\nabla^2\left(\frac{1}{|\mathbf{x}-\mathbf{x}'|}\right) = \delta(\mathbf{x}-\mathbf{x}') \tag{6.3}$$

where δ is the Dirac delta function. The proof of this relationship along with some definitions and properties of distributions relevant to the Dirac delta function (which is in fact a distribution) are given in Appendix. E.

6.1.2 General solution of the Poisson equation with boundary conditions

In numerical simulations, the domain is delimited in space so that appropriate boundary conditions are imposed through Dirichlet or Neumann boundary conditions so that the problem becomes:

$$\begin{cases} \nabla^2\phi = -\dfrac{\rho}{\epsilon_0} \quad \text{in} \quad \mathring{\Omega} & (6.4)\\ \phi = \phi_D \quad \text{on} \quad \partial\Omega_D & (6.5)\\ \dfrac{\partial\phi}{\partial n} = -E_n \quad \text{on} \quad \partial\Omega_N & (6.6) \end{cases}$$

where $\mathring{\Omega}$ denotes the interior of the domain, $\partial\Omega_D$ denotes the Dirichlet boundary, $\partial\Omega_N$ the Neumann boundary. These boundary conditions are the appropriate ones to ensure that a unique and well-behaved solution will exist in the bounded region Ω.

6.1.2.1 Uniqueness of the solution with Dirichlet or Neumann boundary conditions

To show the uniqueness of the solution of the potential with Dirichlet or Neumann boundary conditions, we start from Green's first identity [Jackson, 1999, Chap. 1.8] with arbitrary well behaved functions ψ and ϕ:

$$\int_\Omega(\psi\nabla^2\phi + \nabla\psi\cdot\nabla\phi)dV = \oint_{\partial\Omega}\psi\frac{\partial\phi}{\partial n}dS \tag{6.7}$$

Let's suppose that there are two potentials ϕ_1 and ϕ_2 that satisfy the same Poisson equation with the same boundary conditions:

$$\nabla^2 \phi_i = -\frac{\rho}{\epsilon_0} \quad \text{in} \quad \Omega \tag{6.8}$$

$$\phi_i = \phi_d \quad \text{or} \quad \mathbf{E}_i \cdot \mathbf{n} = E_d \quad \text{on} \quad \partial\Omega \tag{6.9}$$

Let us take the difference $A = \phi_2 - \phi_1$ which is solution of:

$$\nabla^2 A = 0 \quad \text{in} \quad \Omega \tag{6.10}$$

$$A = 0 \quad \text{or} \quad \nabla A \cdot \mathbf{n} = 0 \quad \text{on} \quad \partial\Omega \tag{6.11}$$

Applying the first Green identity [Jackson, 1999, Chap. 1] on $\phi = \psi = A$:

$$\int_\Omega |\nabla A|^2 dV = 0 \implies A = \text{constant} \tag{6.12}$$

For Dirichlet boundary conditions this constant is then null since $A = 0$ on the boundary $\partial\Omega_D$, therefore the potentials ϕ_1 and ϕ_2 are strictly equal. For Neumann boundary conditions, potentials are equal up to a constant.

6.1.2.2 Formal solution of electrostatic boundary-value problem with Green function

Starting from the solution in free space which rests heavily upon the following equality Eq. (6.3) we generalize this equality by introducing Green functions $G(\mathbf{x}, \mathbf{x}')$ that satisfies:

$$\nabla^2 G(\mathbf{x}, \mathbf{x}') = \delta(\mathbf{x} - \mathbf{x}') \tag{6.13}$$

with

$$G(\mathbf{x}, \mathbf{x}') = \frac{1}{|\mathbf{x} - \mathbf{x}'|} + F(\mathbf{x}, \mathbf{x}') \tag{6.14}$$

$$\nabla^2 F(\mathbf{x}, \mathbf{x}') = 0 \tag{6.15}$$

Hence $G(\mathbf{x}, \mathbf{x}')$ is the sum of the free space solution with an added function which is solution to the Laplace equation and accounts for the boundary conditions. Using Green's theorem on ϕ and G yields:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0} \int \rho(\mathbf{x}') G(\mathbf{x}, \mathbf{x}') dV' + \frac{1}{4\pi} \int \left(G \frac{\partial \phi}{\partial n'} - \phi \frac{\partial G}{\partial n'} \right) dS' \tag{6.16}$$

This equation has the problem of having to specify Neumann and Dirichlet boundary conditions on the same boundaries. This is solved by setting appropriate conditions on $G(\mathbf{x}, \mathbf{x}')$. For example for Dirichlet boundary conditions we impose that:

$$G_D(\mathbf{x}, \mathbf{x}') = 0 \quad \text{on} \quad \partial\Omega \tag{6.17}$$

so that the general solution is:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0}\int \rho(\mathbf{x}')G_D(\mathbf{x},\mathbf{x}')dV' - \frac{1}{4\pi}\int \phi_D \frac{\partial G_D}{\partial n'}dS' \tag{6.18}$$

For Neumann boundary conditions, the mean value of the potential over the boundary surface is included in the solution [Jackson, 1999, Chap. 1.10]:

$$\phi(\mathbf{x}) = \langle\phi\rangle_S + \frac{1}{4\pi\epsilon_0}\int \rho(\mathbf{x}')G_N(\mathbf{x},\mathbf{x}')\mathrm{d}V' - \frac{1}{4\pi}\int G_N E_{n'}\mathrm{d}S' \tag{6.19}$$

6.1.2.3 The Poisson equation as a functional minimum

The Poisson equation can be deduced from variational principles using the electric energy. Let us consider the following functional:

$$I(\psi) = \frac{1}{2}\int_V \nabla\psi\cdot\nabla\psi dV - \int_V g\psi dV \tag{6.20}$$

Making an infinitesimal change in ψ, $\psi \to \psi+\eta$, the functional I changes by an amount:

$$\delta I = \int_V \nabla\psi\cdot\nabla\eta dV - \int_V g\eta dV \tag{6.21}$$

$$\implies \delta I = \int_V (-\nabla^2\psi - g)\eta dV + \int_{\partial V}\eta\frac{\partial\psi}{\partial n}dS \tag{6.22}$$

Hence this functional is at an extremum provided that:

$$\eta = 0 \quad \text{on} \quad \partial V \tag{6.23}$$

$$\nabla^2\psi = -g \tag{6.24}$$

Setting $g = \rho/\epsilon_0$, $\psi = \phi$ and multiplying the functional I by ϵ_0 we get the energy functional:

$$E(\phi) = \int_V \left[\frac{\epsilon_0}{2}(\nabla\phi)^2 - \rho\phi\right]dV \tag{6.25}$$

This energy is at an extremum (it is in fact a minimum) if the potential ϕ satisfies the Poisson equation. Two terms are competing against each other in this energy functional: the electric energy $\epsilon_0\mathbf{E}^2/2$ and the potential energy.

6.1.3 General solution of the Poisson equation in a rectangular cartesian geometry

The general solution of the Poisson equation in a canonical case is derived here in order to get insights about how the Poisson equation behaves: from the propagation of boundary conditions to the diffusion of charge density. A simple three-dimensional rectangular box of size (L_x, L_y, L_z) shown in Fig. 6.1 is considered (note that directions are either written in (x, y, z) or (x_1, x_2, x_3) form, the second form allowing to index directions). On each of the boundary faces Dirichlet and Neumann boundary conditions are considered. We separate

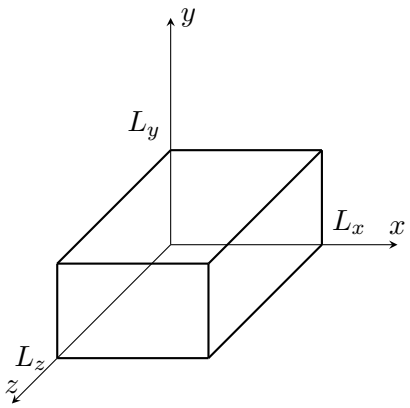


Figure 6.1: Rectangular box considered for the Poisson problem.

the general problem in two sub-problems by linearity: the RHS problem with zero boundary conditions, *i.e.* the values of the Dirichlet and/or Neumann boundary conditions are set to zero with a non-zero charge density, on one hand and the zero RHS boundary conditions, *i.e.* a zero charge density with non-zero Dirichlet and/or Neumann boundary conditions on the other hand. The resolution of each problem is carried out in the paragraphs below.

6.1.3.1 Zero Dirichlet and Neumann interior problem

From Eqs. (6.4) to (6.6) we consider the zero boundary conditions problem:

$$\begin{cases} \nabla^2\phi = -\dfrac{\rho}{\epsilon_0} \quad \text{in} \quad \mathring{\Omega} & (6.26) \\ \phi = 0 \quad \text{on} \quad \partial\Omega_D & (6.27) \\ \dfrac{\partial\phi}{\partial n} = 0 \quad \text{on} \quad \partial\Omega_N & (6.28) \end{cases}$$

We need to find the Green function associated to this problem. From Eqs. (6.18) and (6.19), $\phi_D = 0$ and $E_n = 0$ is imposed and by setting the average value of the potential to zero the following general form is retrieved for both Dirichlet and Neumann boundary conditions:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0}\int \rho(\mathbf{x}')G(\mathbf{x},\mathbf{x}')dV' \tag{6.29}$$

The expansion in an orthonormal basis of the Green function of the problem is now sought. Starting from a slightly more general problem of finding the Green function of the following elliptic equation [Jackson, 1999, Chap. 3.10]:

$$\nabla_x^2 G(\mathbf{x},\mathbf{x}') + [f(x) + \lambda]G(\mathbf{x},\mathbf{x}') = -4\pi\delta(\mathbf{x}-\mathbf{x}') \tag{6.30}$$

the eigenvalues and eigenvectors (λ_n, ψ_n) of the problem are

$$\nabla^2\psi_n + [f(x) + \lambda_n]\psi_n(\mathbf{x}) = 0 \tag{6.31}$$

The expansion of the Green function in the orthonormal basis of the eigenvectors is now

given by [Jackson, 1999, Chap. 3.12]:

$$G(\mathbf{x},\mathbf{x}') = 4\pi \sum_n \frac{\psi_n^*(\mathbf{x}')\psi_n(\mathbf{x})}{\lambda_n - \lambda} \tag{6.32}$$

For the Poisson equation $f(x) = 0$ and $\lambda = 0$. The orthonormal eigenfunctions of the problem now need to be found:

$$(\nabla^2 + k_{nml}^2)\psi_{nml} = 0 \tag{6.33}$$

The three-dimensional eigenfunction is the product of one-dimensional eigenfunctions such that $\psi_{nml} = \psi_n\psi_m\psi_l$ (again indices for each direction are either denoted n, m, l or n_1, n_2, n_3). In each direction, three configurations are possible in the two limits $(0, L_i)$ of the boundaries: Dirichlet boundary conditions, Neumann boundary conditions and mixed boundary conditions. The orthonormal eigenfunctions for each type of boundary conditions are as follows:

$$\text{Dirichlet} \quad \psi_n(x) = \sqrt{\frac{2}{L_x}} \sin\left(\frac{n\pi x}{L_x}\right) \tag{6.34}$$

$$\text{Neumann} \quad \psi_n(x) = \sqrt{\frac{2}{L_x}} \cos\left(\frac{n\pi x}{L_x}\right) \tag{6.35}$$

$$\text{Dirichlet} - \text{Neumann} \quad \psi_n(x) = \sqrt{\frac{2}{L_x}} \sin\left[\frac{n\pi}{L_x}\left(x - \frac{L_x}{2}\right)\right] \tag{6.36}$$

Injecting the general eigenfunction in each direction ψ_n into Eq. (6.32) yields

$$G(\mathbf{x},\mathbf{x}') = 4\pi \sum_{n_i=1}^{+\infty} \frac{\prod_{i=1}^{n_d} \frac{2}{L_i}\psi_{n_i}(x_i')\psi_{n_i}(x_i)}{\sum_{i=1}^{n_d}\left(\frac{\pi n_i}{L_i}\right)^2} \tag{6.37}$$

6.1.3.2 Boundary conditions problem

We now turn to the Laplace problem associated to Eqs. (6.4) to (6.6) where the effect of non-zero boundary conditions can be observed:

$$\begin{cases} \nabla^2\phi = 0 \quad \text{in} \quad \mathring{\Omega} & (6.38) \\ \phi = \phi_D \quad \text{on} \quad \partial\Omega_D & (6.39) \\ \dfrac{\partial\phi}{\partial n} = -E_n \quad \text{on} \quad \partial\Omega_N & (6.40) \end{cases}$$

We thus need to solve the Laplace equation with non-trivial boundary conditions in cartesian coordinates. We first describe the resolution in two dimensions with the specific case of a non-zero Dirichlet boundary at the bottom $\phi(x, L_y) = V_u(x)$ and zero potential on the three remaining boundaries shown in Fig. 6.2. The 2D Laplace equation reads:

$$\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} = 0 \tag{6.41}$$

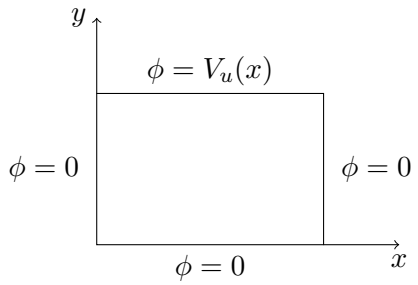


Figure 6.2: 2D model Laplace problem considered.

Separating variables for the potential, the following equalities are obtained:

$$\phi(x,y) = X(x)Y(y) \tag{6.42}$$

$$\implies \frac{1}{X}\frac{\partial^2 X}{\partial x^2} + \frac{1}{Y}\frac{\partial^2 Y}{\partial y^2} = 0 \tag{6.43}$$

$$\implies \frac{1}{X}\frac{\partial^2 X}{\partial x^2} = -\alpha^2 \quad \text{and} \quad \frac{1}{Y}\frac{\partial^2 Y}{\partial y^2} = \alpha^2 \tag{6.44}$$

$$\implies \phi = e^{\pm i\alpha x} e^{\pm \alpha y} \tag{6.45}$$

With these boundary conditions, the following choice of basis functions for ϕ satisfying the boundary conditions is:

$$\phi_n = \sin(\alpha_n x)\sinh(\alpha_n y) \quad \text{with} \quad \alpha_n = \frac{n\pi}{L_x} \tag{6.46}$$

Hence the general form of the potential is given by the following series expansion:

$$\phi(x,y) = \sum_{n=1}^{+\infty} A_n \sin\left(\frac{n\pi x}{L_x}\right)\sinh\left(\frac{n\pi y}{L_x}\right) \tag{6.47}$$

The sequence (A_n) is determined by the potential at the top V_u since $\phi = V_u$ is imposed at this boundary:

$$\phi(x,L_y) = V_u(x) = \sum_{n=1}^{+\infty} A_n \sin\left(\frac{n\pi x}{L_x}\right)\sinh\left(\frac{n\pi L_y}{L_x}\right) \tag{6.48}$$

By orthogonality of the Fourier series functions the coefficients are given by the following integral:

$$A_n = \frac{2}{L_x \sinh\left(\frac{n\pi L_y}{L_x}\right)} \int_0^{L_x} V_u(x)\sin\left(\frac{n\pi x}{L_x}\right)dx \tag{6.49}$$

To solve the problem with other settings of boundary conditions the base functions Eq. (6.46) need to be adapted. We separate the total boundary conditions problems into four sub-problems in two dimensions and six in three dimensions setting one boundary at the prescribed value and all the other ones at zero. In the direction of the non-zero boundary

condition sinh is taken if it is a non-trivial Dirichlet boundary condition (just like above) and cosh is taken if it is a non-trivial Neumann boundary condition. For the other directions the choices of the basis functions are made according to Eqs. (6.34)-(6.36).

6.2 Numerical integration of the Poisson equation

The discretization of the Poisson equation in structured meshes using finite differences is presented first in cartesian and cylindrical geometries. In a second part the unstructured discretization of the Poisson equation implemented in AVIP is detailed. The Laplacian cells and matrices of the underlying linear systems are also exhibited for canonical configurations in both structured and unstructured meshes.

6.2.1 Finite difference formulations

This section is dedicated to the discretization of the Poisson equation on structured grids. First the discretization of the Laplacian on interior nodes is considered in cartesian and axisymmetric geometries and then the boundary conditions are dealt with. In this section the index i refers to the x direction and the index j refers to either the y (Cartesian coordinates) or the r (axisymmetric coordinates) direction. The Poisson equation reduces to a linear systems $A\phi = b$ in numerical simulations whose coefficients depend on the chosen discretization, *i.e.* the formula used for the Laplacian.

6.2.1.1 Cartesian

In Cartesian coordinates, the Laplace operator is simply a sum of second order derivatives so that

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} \tag{6.50}$$

Using second order centered schemes for both derivatives, the Laplace operator for an interior node is

$$\nabla^2\phi_{i,j} = \frac{\phi_{i+1,j} + \phi_{i-1,j} - 2\phi_{i,j}}{\Delta x^2} + \frac{\phi_{i,j+1} + \phi_{i,j-1} - 2\phi_{i,j}}{\Delta y^2} \tag{6.51}$$

6.2.1.2 Cylindrical

We denote by (r, θ, x) the cylindrical coordinates. In axisymmetric conditions, the partial derivative with respect to the angle θ in cylindrical coordinates of all the physical values are equal to zero. Hence we are considering variables in the (r, x) plane. The operators on a scalar fields $\phi(r, x)$ an vector field $u(r, x)$ in axisymmetric conditions become:

$$\boldsymbol{\nabla}\phi = \frac{\partial\phi}{\partial x}\mathbf{e}_x + \frac{\partial\phi}{\partial r}\mathbf{e}_r \tag{6.52}$$

$$\boldsymbol{\nabla}\cdot\mathbf{u} = \frac{\partial u_x}{\partial x} + \frac{1}{r}\frac{\partial r u_r}{\partial r} \tag{6.53}$$

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\phi}{\partial r}\right) \tag{6.54}$$

The radius $r_{i,j}$ associated to each node in two-dimensions has to be corrected for the axis nodes as the radius of the center of gravity associated to the nodal volume (which in the case of axis nodes is not the same point as the node itself).

6.2.1.3 Boundary conditions

After construction of the matrix for all the interior nodes, boundary nodes need to be treated separately. Dirichlet boundary conditions are enforced in the matrix by simply putting 1.0 coefficients for the Dirichlet nodes. As an illustration in 1D, the discretized Poisson matrix with Dirichlet boundary conditions is given by:

$$A = \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ 1 & -2 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 1 & -2 & 1 \\ 0 & \dots & \dots & 0 & 1 \end{bmatrix} \tag{6.55}$$

Neumann boundary conditions are imposed by using a ghost node method for second order accuracy and the resulting one-dimensional matrix for a left-Neumann and right-Dirichlet boundary Poisson problem is

$$A = \begin{bmatrix} -2 & 2 & \dots & \dots & 0 \\ 1 & -2 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 1 & -2 & 1 \\ 0 & \dots & \dots & 0 & 1 \end{bmatrix} \tag{6.56}$$

6.2.2 Vertex-Centered Finite Volume formulation

6.2.2.1 Cartesian geometry

We integrate the Poisson equation Eq. (6.4) in the nodal volume V_i for an interior node i (*i.e.* such that $\partial V_i^b = \emptyset$):

$$\int_{V_i}\nabla^2\phi \mathrm{d}V = \int_{\partial V_i}\nabla\phi\cdot\mathbf{n}\mathrm{d}S = \sum_{\tau\in E(i)}\int_{\partial V_i\cap\tau}\nabla\phi\cdot\mathbf{n}\mathrm{d}S \tag{6.57}$$

The gradient is supposed to be constant inside each cell τ so that it can be extracted from the integral yielding

$$\int_{V_i} \nabla^2 \phi dV = \sum_{\tau \in E(i)} \nabla \phi_\tau \cdot \int_{\partial V_i \cap \tau} \mathbf{n} dS = \sum_{\tau \in E(i)} \nabla \phi_\tau \cdot \sum_{j \in S(i) \cap \tau} \mathbf{S}_{i,j}^\tau \tag{6.58}$$

The value of this constant gradient inside cell τ is computed from the Green-Ostrogradski theorem:

$$\nabla \phi_\tau = \frac{1}{V_\tau} \int_{\partial V_\tau} \phi \, \mathbf{n} \, dS \tag{6.59}$$

$$= \frac{1}{V_\tau} \sum_{f \in F(\tau)} \phi_f \mathbf{S}_f \tag{6.60}$$

where $F(\tau)$ are the set of faces of an element τ and $\mathbf{n}$ denotes undimensionalized normals. Note that $\mathbf{S}_i$ has the dimension of a surface in 3D or a length in 2D. Following the definition of nodal normals Eq.(5.9) the gradient can be rewritten as [Lamarque, 2007, Chap. 4]:

$$\nabla \phi_\tau = -\frac{1}{V_\tau n_d} \sum_{k \in \tau} \phi_k \mathbf{S}_k \tag{6.61}$$

which in the end yields:

$$\int_{V_i} \nabla^2 \phi dV = \sum_{\tau \in E(i)} -\frac{1}{V_\tau n_d} \sum_{k \in \tau} \phi_k \mathbf{S}_k \cdot \sum_{j \in S(i) \cap \tau} \mathbf{S}_{i,j}^\tau \tag{6.62}$$

The following relation is exact in the case of triangular, tetrahedral and holds for regular quadrangular and hexahedral elements [Auffray, 2007]:

$$\sum_{j \in S(i) \cap \tau} \mathbf{S}_{i,j}^\tau = \mathbf{S}_i^\tau / n_d \tag{6.63}$$

which yields

$$\int_{V_i} \nabla^2 \phi dV = \sum_{\tau \in E(i)} \sum_{k \in \tau} \left[-\frac{\mathbf{S}_k \cdot \mathbf{S}_i}{V_\tau n_d^2} \right] \phi_k. \tag{6.64}$$

Finally with the integration of the charge density the following equation is obtained from the integration of both sides of the Poisson equation:

$$\sum_{\tau \in E(i)} \sum_{k \in \tau} \left[\frac{\mathbf{S}_k \cdot \mathbf{S}_i}{V_\tau n_d^2} \right] \phi_k = \frac{\rho_i V_i}{\varepsilon_0} \tag{6.65}$$

Eq. (6.65) defines a linear system, i being the line and k the column of the matrix for all i and k taking the values of all the inner nodes of the partition. One interesting feature of this matrix in cartesian coordinates is its symmetry. The Laplacian operator matrix A is computed with node normal vectors and primal cell volumes where each coefficient a_{ij} stands as:

$$a_{ij} = \sum_{ij \in \tau} \left[-\frac{\mathbf{S}_i \cdot \mathbf{S}_j}{V_\tau n_d^2} \right] \tag{6.66}$$

where the sum is performed over the cells for which ij is an edge.

Remark on the implementation: The linear system solved in AVIP is in fact the one obtained from:

$$-\nabla^2\phi = \frac{\rho}{\epsilon_0} \tag{6.67}$$

6.2.2.2 Axisymmetric configuration

The solution of Poisson's equation in an $r-z$ axisymmetry geometry is needed in streamer simulations. First integration over the nodal volume is performed, here the volume is $dV = r\mathrm{d}r\mathrm{d}\theta\mathrm{d}z$:

$$\int_{V_i} \nabla^2\phi \,\mathrm{d}V = \int_{V_i} \nabla^2\phi \, r\mathrm{d}r\mathrm{d}\theta\mathrm{d}z. \tag{6.68}$$

By assuming axisymmetric conditions *i.e.* $\partial/\partial\theta = 0$, using Green's theorem, summing over the neighboring cells and assuming constant gradient within a cell, it yields:

$$\begin{aligned}
\int_{V_i} \nabla^2\phi dV &= 2\pi \int_{A_i} \nabla\cdot(r\nabla\phi) dA && (6.69)\\
&= 2\pi \int_{\partial A_i} r\nabla\phi\cdot\mathbf{n} dl && (6.70)\\
&= \sum_{\tau\in E(i)} 2\pi \int_{\partial A_i\cap\tau} r\nabla\phi\cdot d\mathbf{l} && (6.71)\\
&= \sum_{\tau\in E(i)} 2\pi\nabla\phi_\tau \cdot \int_{\partial A_i\cap\tau} r d\mathbf{l} && (6.72)
\end{aligned}$$

In this section, all the normals denoted $\mathbf{n}$ are homogeneous to a length. The cell gradient is discretized as in the last section with nodal normals (note here that the gradient is really 2D here, all the axisymmetric information is contained with the radius):

$$\nabla\phi_\tau = -\frac{1}{A_\tau n_d}\sum_{k\in\tau}\phi_k\mathbf{n}_k. \tag{6.73}$$

Using the notations from Fig. 5.3a, for a triangle the remaining integral yields:

$$\int_{\partial A_i\cap\tau} r d\mathbf{l} = \frac{r_\tau + r^\tau_{ij}}{2}\mathbf{n}^\tau_{ij} + \frac{r_\tau + r^\tau_{ik}}{2}\mathbf{n}^\tau_{ik} \tag{6.74}$$

From Auffray [2007]:

$$\mathbf{n}^\tau_{ij} = \frac{1}{6}(\mathbf{n}_i - \mathbf{n}_j) \tag{6.75}$$

In the end:

$$\int_{\partial A_i\cap\tau} r d\mathbf{l} = \frac{1}{24}\left[(4r_\tau + 2r_i + r_j + r_k)\mathbf{n}_i - (2r_\tau + r_i + r_j)\mathbf{n}_j - (2r_\tau + r_i + r_k)\mathbf{n}_k\right] \tag{6.76}$$

For the quadrangular cell from Fig. 5.3b, using similar relations the resulting integral

yields:

$$\int_{\partial A_i \cap \tau} r d\mathbf{l} = \frac{1}{16}\left[(4r_\tau + 2r_i + r_j + r_l)\mathbf{n}_i - (2r_\tau + r_i + r_j)\mathbf{n}_j - (2r_\tau + r_i + r_l)\mathbf{n}_l\right] \tag{6.77}$$

From these relations an equation similar to Eq. (6.65) can be obtained yielding a different linear system: the radiuses introduced break the symmetry of the linear system matrix.

6.2.3 Boundary conditions

Two kind of boundary conditions are implemented in AVIP for the Poisson equation: Dirichlet and Neumann boundary conditions. In practice the laplacian discretization is computed for all nodes and then boundary conditions are applied on the Poisson matrix.

6.2.3.1 Neumann boundary conditions

For a Neumann boundary condition node i, *e.g.* in Figs. 5.4a and 5.4b, integration of the Poisson equation yields

$$\int_{V_i} \nabla^2 \phi \mathrm{d}V = \int_{\partial \mathring{V}_i} \nabla\phi \cdot \mathbf{n} \mathrm{d}S + \int_{\partial V_i^b} \nabla\phi \cdot \mathbf{n} \mathrm{d}S \tag{6.78}$$

The first term in the right hand side is discretized as detailed in the sections above. Introducing the node normal boundary $\mathbf{S}_i^b$, which is the weighted sum of the face boundary normals shown in Fig. 6.3, the second term reduces to

$$\int_{\partial V_i^b} \nabla\phi \cdot \mathbf{n} \mathrm{d}S = \nabla\phi \cdot \mathbf{n} ||\mathbf{S}_i^b|| \tag{6.79}$$

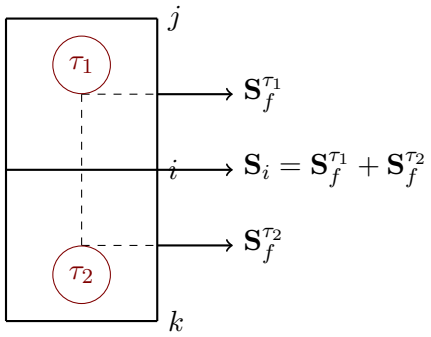


Figure 6.3: Boundary conditions normals.

The Neumann boundary condition is imposed by modifying the right hand side of the Poisson equation with the normal electric field E_n from Eq. (6.6). Using Eq. (6.65), the complete discretized line for a Neumann boundary condition node i in the Poisson matrix is thus

$$\sum_{\tau \in E(i)} \sum_{k \in \tau} \frac{\mathbf{S}_k \cdot \mathbf{S}_i}{V_\tau n_d^2} \phi_k = \frac{\rho_i V_i}{\varepsilon_0} - E_n^i ||\mathbf{S}_i^b|| \tag{6.80}$$

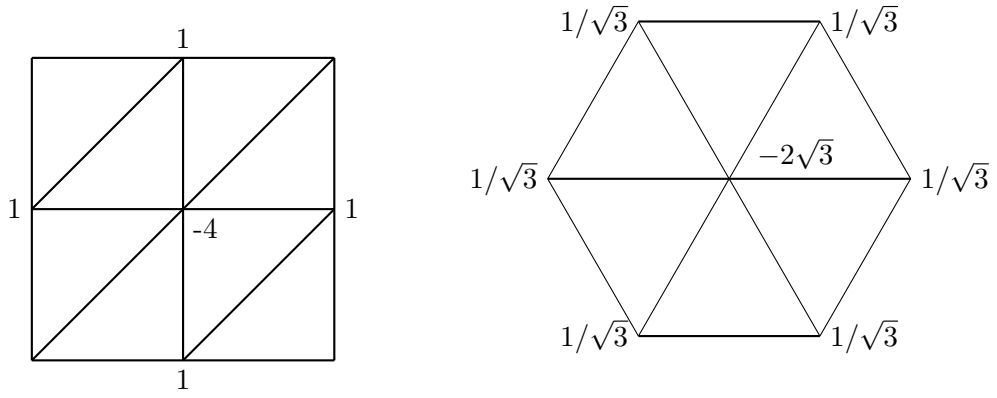
Neumann boundary conditions are thus natural conditions for the Poisson equation in this vertex-centered formulation. Indeed when a boundary node is discretized using the laplacian discretization and nothing else is applied a zero Neumann boundary condition is implied.

6.2.3.2 Dirichlet boundary conditions

Dirichlet boundary conditions are enforced the same way as for the structured meshes in Section. 6.2.1.3, by imposing a one coefficient on the matrix and setting the imposed value in the right hand side of the linear system.

6.2.4 Laplacian cells for canonical meshes

The discretization of the Poisson equation is studied by looking at the coefficients of the Poisson matrix for the cartesian and axisymmetric formulations. We first look at the discretization coefficients given by Eq. (6.65) in 2D for two different kinds of triangles: triangles cutted from squares and equilateral triangles. The classical five point molecule is retrieved for the first case, which is exactly the same discretization as the structured discretization Eq. (6.51), whereas a seven point molecule is retrieved for equilateral triangles.



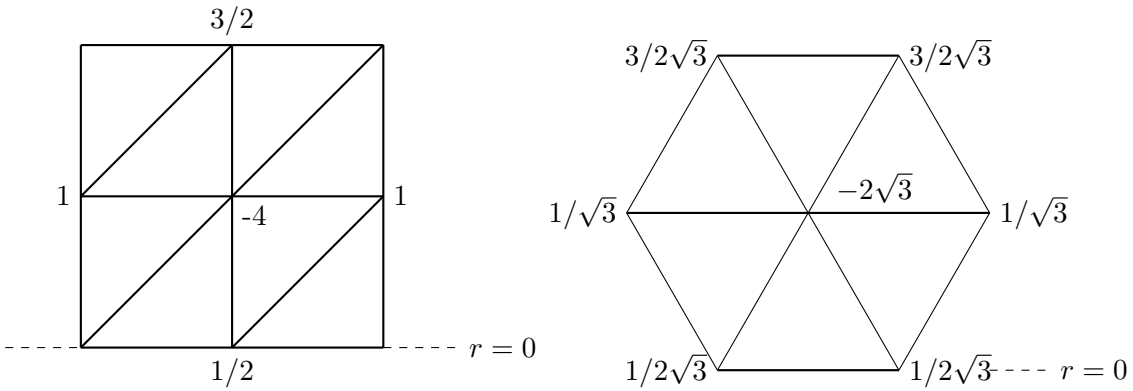
(a) Triangular elements

(b) Equilateral triangles

Figure 6.4: Laplacian cells for the 2D cartesian discretization of the Poisson equation Eq. (6.65).

The matrices for the two kind of triangles studied above yield for the first cell at the axis the coefficients show in Fig. 6.5. The top-bottom symmetry is broken in this case so that the matrix is clearly non-symmetric for axisymmetric conditions.

The Laplacian cell for quadrangular elements using Eq. (6.65) is shown in Fig. 6.6. This discretization is not the same as for the cutted squares shown in Fig. 6.4a and there is an odd-even decoupling of the solution. This is problematic as two groups of nodes are separated by this discretization that never interact with each other which can lead to instabilities. This discretization is however valid and the proof can be carried out using the multi-dimensional Taylor expansion below



(a) Triangular elements at first node (b) Equilateral triangles at first node

Figure 6.5: Laplacian cells in axysymmetric formulation.

$$\phi_{i+n,j+m} = \sum_{k=1}^{+\infty} \frac{1}{k!} \left[i\Delta x \frac{\partial}{\partial x} + i\Delta y \frac{\partial}{\partial y} \right]^k \phi_{i,j} \tag{6.81}$$

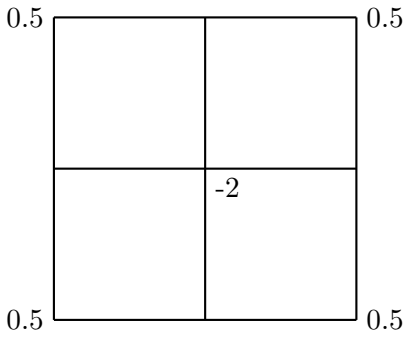


Figure 6.6: Quadrangular elements Laplacian cells in 2D.

To circumvent this odd-even decoupling a procedure is applied whenever quadrangular elements are present in the mesh by creating virtual elements to discretize the quadrangular as cutted quadrangular so that the resulting Laplacian cell is Fig. 6.4a. The procedure is summarized in Fig. 6.7. The quadrangular element τ is cutted in half along the diagonal into two virtual triangles τ_1 and τ_2. The four vertices of τ, originally numerotated from 1 to 4, are given virtual indices shown in the corresponding color of the virtual element in Fig. 6.7.

6.3 Linear systems

The Poisson equation is discretized following the numerical methods described in the previous section to yield a linear system of the form

$$Ax = b \tag{6.82}$$

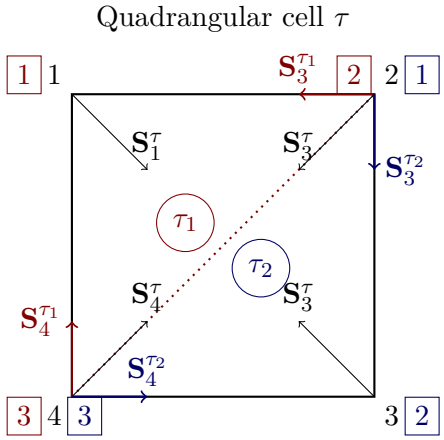


Figure 6.7: Stabilized version of quadrangular discretization.

where A is the Poisson matrix with boundary conditions, x is the potential vector and b the charge density with boundary conditions.

A brief overview of linear system solvers is given in this section with emphasis on Krylov solvers which are used in practice: Conjugate Gradient and GMRES. Some linear algebra material such as basic definitions, properties and theorems can be found in Appendix. D. Proofs are omitted throughout this chapter but the interest reader can find them in Quarteroni et al. [2007], Saad [2003].

6.3.1 Preconditioning

Preconditioning is an important block of linear system solvers. It is a step that is applied before and/or after the solver to stabilize the system. We first define the condition number of a matrix.

Definition 6.1 (Condition number)**.** *Given a matrix norm $||\cdot||$, the condition number of a matrix A is given by*

$$K(A) = ||A||\,||A^{-1}|| \tag{6.83}$$

A condition number is defined for each norm that we choose and the condition number for the 2-norm is called the spectral condition number because it is related to the singular values of the matrix:

$$K_2(A) = \frac{\sigma_1(A)}{\sigma_n(A)} \tag{6.84}$$

where σ_1 and σ_n are respectively the highest and lower singular values of the matrix.

Property 6.1 (Distance singular matrices)**.** *The distance of matrix A to the set of singular matrices is*

$$\text{dist}(A) = \min\{||\delta A||/||A|| : \det(A + \delta A) = 0\} \tag{6.85}$$

This distance is related to the condition number as

$$\text{dist}(A) = 1/K(A) \tag{6.86}$$

and hence all invertible matrices $A + \delta A$ satisfy

$$||\delta A||\,||A^{-1}|| < 1 \tag{6.87}$$

Hence the condition number of a matrix tells us how far the system is from being ill-conditioned, the lower the condition number the more stable the system.

This statement is made more precise by the following forward analysis. The floating point representation of number implies that all numbers are represented with a given degree of accuracy represented by the machine precision. For a designed linear system this mean that the matrix A and right hand side b are perturbed by δA and δb, respectively. We consider the perturbation on the solution δx that is solution of the perturbed system:

$$(A + \delta A)(x + \delta x) = b + \delta b \tag{6.88}$$

The perturbation on the solution given perturbed matrix and right-hand side are governed by the following inequality:

$$\frac{||\delta x||}{||x||} \leq \frac{K(A)}{1 - K(A)||\delta A||/||A||}\left(\frac{||\delta b||}{||b||} + \frac{||\delta A||}{||A||}\right) \tag{6.89}$$

We would expect for a stable system to have a perturbed solution close to the actual solution, *i.e.* a low value of $||\delta x||/||x||$. The higher the condition number, the higher the norm of the perturbation can be so that the actual computed solution can diverge from the actual solution of the linear system.

Hence for a given linear system $Ax = b$ the goal of preconditioning is to reduce the condition number of the system by solving an equivalent linear system which has the same solution. Three kinds of preconditioning are possible:

1. Left preconditioning $P^{-1}Ax = P^{-1}b$

2. Right preconditioning $AP^{-1}y = b$ and $y = Px$

3. Centered preconditioning $P_L^{-1}AP_R^{-1}y = b$ and $y = P_R x$

In each case we expect to have $K(P^{-1}A) < K(A)$ so that the system would be more stable.

6.3.2 Condition numbers of cartesian and cylindrical matrices

From the above properties and relations, the condition number is a critical quantity to assess the difficulty to solve a linear system. The comparison of cartesian and cylindrical matrices of the Poisson equation discretization is carried out here to understand what the difference is between the two formulations.

A 1D-domain with n nodes is considered with a Neumann condition at one end and a Dirichlet boundary condition at the other end. The cartesian matrix reads

$$A = \begin{bmatrix} -2 & 2 & \dots & \dots & 0 \\ 1 & -2 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 1 & -2 & 1 \\ 0 & \dots & \dots & 0 & 1 \end{bmatrix} \tag{6.90}$$

whereas the equivalent cylindrical matrix reads

$$A = \begin{bmatrix} -4 & 4 & \dots & \dots & 0 \\ 1/2 & -2 & 3/2 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & (n-3/2)/n & -2 & (n-1/2)/n \\ 0 & \dots & \dots & 0 & 1 \end{bmatrix} \tag{6.91}$$

These discretizations are obtained by the square-cutted triangles shown in Fig. 6.4a and 6.5a in the y direction. The spectral condition number $K_2(A)$ for these two matrices is shown in Fig. 6.8. The spectral condition numbers increase as the number of nodes increase which is expected: the bigger the size of the system the harder it is to solve. The cylindrical formulation (red plain curve) seems to be slightly more stable than the cartesian formulation (blue dashed curve) as the cylindrical spectral condition number is smaller for a given resolution. An important property that is lost however for the cylindrical matrix is the symmetry property for which fast solvers exist as shown in the following section.

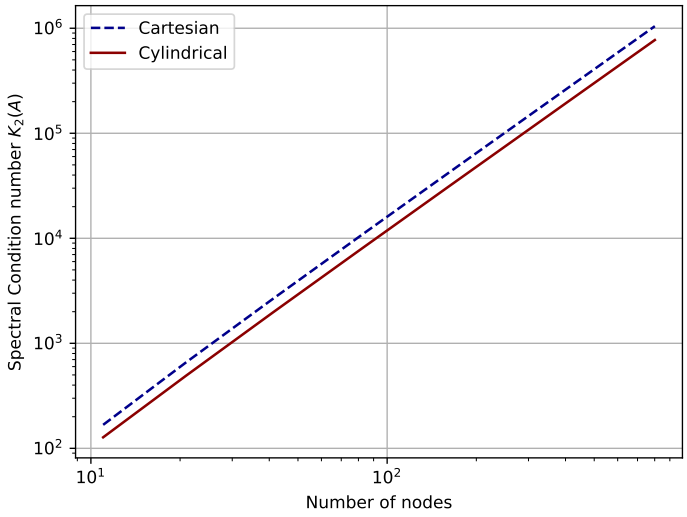


Figure 6.8: Spectral condition number at different resolutions for cartesian and cylindrical matrices.

6.3.3 Iterative solvers

Two kind of linear system solvers exist: direct solvers and iterative solvers. Iterative solvers are less memory intensive and faster than their direct counterparts. In the context of a HPC

code these methods are preferred although direct methods are more robust and accurate but more memory intensive.

As will be seen from the results shown below iterative solvers require a deeper analysis of the considered linear system to be efficient. This is because convergence is certain only assuming certain matrix properties whereas for a non-singular matrix a simple direct method such as LU always yields a solution.

In iterative methods we try to find a sequence of vectors that tend to the desired one, solution of the system $Ax = b$

$$x = \lim_{k \to +\infty} x^{(k)} \tag{6.92}$$

where k is iteration number. Given an initial guess $x^{(0)}$ the following iteration procedure is performed where B is called the iteration matrix:

$$x^{(k+1)} = Bx^{(k)} + f \quad k \geq 0 \tag{6.93}$$

For consistency we must have $x = Bx + f$ or equivalently $f = (I - B)A^{-1}b$. This procedure converges if and only if $\rho(B) < 1$. A general method to devise consistent linear method consists in splitting the matrix $A = P - N$ where P is called the preconditioner. From this splitting the iteration procedure writes

$$Px^{(k+1)} = Nx^{(k)} + b \quad k \geq 0 \tag{6.94}$$

This decomposition can be directly linked with the iteration matrix as $B = P^{-1}N$. Finally another common rewriting of the iteration procedure with such a decomposition is the following one:

$$x^{(k+1)} = x^{(k)} + P^{-1}(b - Ax^{(k)}) = x^{(k)} + P^{-1}r^{(k)} \tag{6.95}$$

Method	Iteration Matrix	P	N
Jacobi	$B_J = I - D^{-1}A$	D	$E + F$
JOR	$B_{J\omega} = \omega B_J + (1 - \omega)I$	-	-
Gauss-Seidel	$(D - E)^{-1}F$	$D - E$	F
SOR	$B = (I - \omega D^{-1}E)^{-1}(\omega D^{-1}F + (1 - \omega)I)$	-	-

Table 6.1: Common iterative methods following the $A = P - N$ decomposition.

Eqs. (6.93), (6.94) and (6.95) are all equivalent and all iterative methods can be written done in either of these forms and some of them are recalled in Table. 6.1.

6.3.3.1 Basic iterative methods

The Jacobi, Gauss-Seidel and their relaxation counterparts JOR and SOR are recalled in Tab. 6.1. For these methods convergence results exist and the main ones are laid out next.

Theorem 6.1. *If A is a strictly diagonally dominant matrix by rows, then the Jacobi and Gauss-Seidel methods are convergent.*

Theorem 6.2. *If A is symmetric positive definite, the Gauss-Seidel method is monotonically convergent with respect to $||\cdot||_A$.*

Theorem 6.3. *If the Jacobi method is convergent then JOR method is convergent for $0 < \omega \leq 1$.*

6.3.3.2 The Gradient and Conjugate Gradient methods

For a positive definite matrix A we define the energy of the linear system

$$E(y) = \frac{1}{2} y^T A y - y^T b \tag{6.96}$$

The gradient of the energy is $\nabla E = Ay - b$ so that for the solution of the system x we have $\nabla E = 0$, *i.e.* the solution is at an extremum of the energy functional. From a Taylor expansion in multiple dimensions

$$E(y) = E(x + (y - x)) = E(x) + \frac{1}{2}||y - x||_A^2 \tag{6.97}$$

so that the extremum is in fact a minimum. The goal is to update the residuals in the direction of steepest gradient.

The Gradient method [Quarteroni et al., 2007, Chap. 4] uses a Richardson iteration method as follows

$$x^{(k+1)} = x^{(k)} + \alpha_k r^{(k)} \tag{6.98}$$

where the direction of update is simply the gradient direction since $r = \nabla E$ and the optimal parameter for convergence is

$$\alpha_k = \frac{r^{(k)T} r^{(k)}}{r^{(k)T} A r^{(k)}} \tag{6.99}$$

The Conjugate Gradient Method is an improvement of the Gradient Method which finds the optimal direction along which the solution is updated at each iteration. The algorithm reads for each iteration k

$$\alpha_k = \frac{p^{(k)T} r^{(k)}}{p^{(k)T} A p^{(k)}} \tag{6.100}$$

$$x^{(k+1)} = x^{(k)} + \alpha_k p^{(k)} \tag{6.101}$$

$$r^{(k+1)} = r^{(k)} - \alpha_k A p^{(k)} \tag{6.102}$$

$$\beta^{(k+1)} = \frac{(Ap^{(k)})^T r^{(k+1)}}{(Ap^{(k)})^T p^{(k)}} \tag{6.103}$$

$$p^{(k+1)} = r^{(k+1)} - \beta_k p^{(k)} \tag{6.104}$$

A comparison of the Gradient Method and the Conjugate Gradient Method is shown in Fig. 6.9 for a simple 2×2 matrix. The Conjugate Gradient corrects the direction towards the global minimum directly whereas the Gradient Method zigzags around.

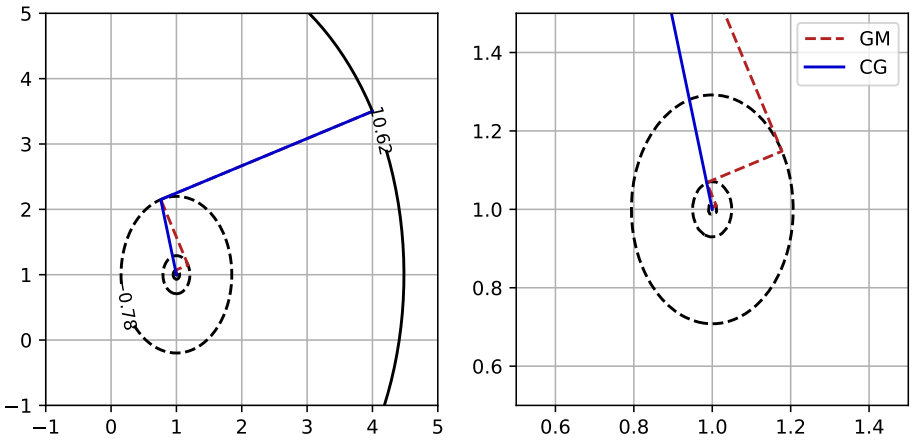


Figure 6.9: Gradient Method (GM) and Conjugate Gradient (CG) comparisons.

As for the convergence of the CG method, we have the following theorem.

Theorem 6.4. *Let A be a symmetric and positive definite matrix. Any method which employs conjugate directions to solve the linear system $Ax = b$ terminates after at most n steps, yielding the exact solution.*

Hence for symmetric positive definite matrices, the CG method should be the first option to consider. For not too non-symmetric, *i.e.* symmetry broken by the boundary conditions only, it is also reasonable to consider CG as the best linear solver. However for matrices whose inner part is non-symmetric such as in the cylindrical case it is not clear whether or not CG is the best option.

6.3.3.3 The Multigrid method

We now turn to a particular class of methods, the multigrid methods, which solve differential equations using a hierarchy of discretizations. Solving linear systems using multiple scales have proven to be very effective especially on elliptic problems: they are scalable as their complexity on specific cases is $\mathcal{O}(n)$ where n is the number of unknowns [Falgout, 2006].

Multigrid methods is built upon two complementary processes: *smoothing* (or relaxing) and *coarse-grid corrections*. The application of a *smoother*, which is a simple iterative method such as the Gauss-Seidel method presented in Section. 6.3.3.1, is carried out in each different resolution. The transfer from a finer resolution to a coarser one is done through *restriction* operators and from coarser to finer resolutions through *interpolation* or *prolongation* operators. This process is illustrated in Fig. 6.10 where the restriction operators are denoted by $\mathbf{R}^i$ and the prolongation operators by $\mathbf{P}^i$.

6.3.3.4 The PETSc solver

In AVIP, we have decided to use the PETSc library (Version 13.4 at the time of writing) [Balay et al., 2021] to solve linear systems. The previously used Maphys software in Joncquieres [2019] has been discarded due to unadapted data structures. PETSc has, on

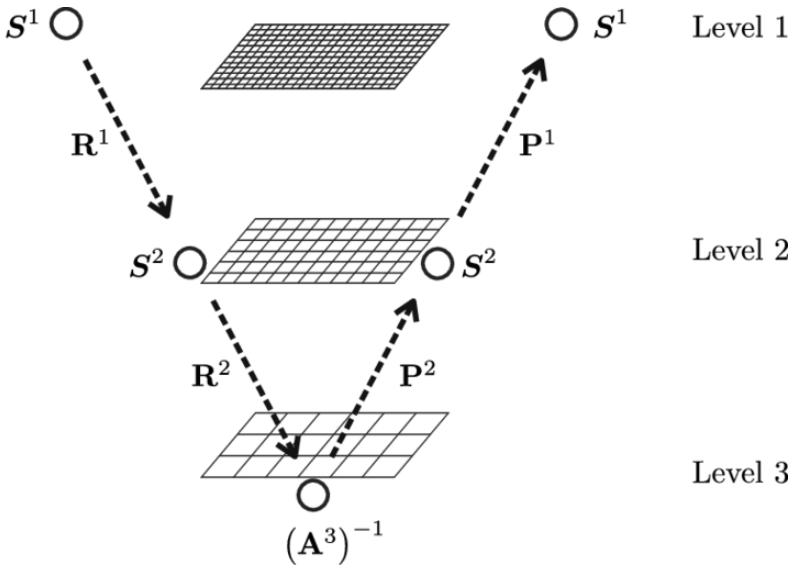


Figure 6.10: Illustration of a multigrid V cycle with 3 levels taken from Verdugo and Wall [2016].

Krylov solvers	Preconditioners
Richardson	Jacobi
Chebychev	SOR
Conjugate Gradient	Algebraic Multigrid
BiCGStab	Boomer AMG
GMRES	LU

Table 6.2: Krylov solvers and preconditioners (partial list).

the other hand, an easy-to-use interface with a data structure easily adaptable to that of AVIP. Various Krylov solvers and preconditioners can be accessed through PETSc shown in Tab. 6.2. It interfaces in particular with Hypre [Falgout and Yang, 2002] which allows to use its well-reknown BoomerAMG multigrid method.

6.4 Test cases

Test cases are performed on the Poisson solver implemented in AVIP with the discretization laid out in Sec. 6.2 with PETSc [Balay et al., 2021].

6.4.1 Convergence

The convergence of the Poisson equation discretization of AVIP is considered in this section. The test case is a simple $[0, 1]^2$ square with zero Dirichlet boundary conditions labeled case A shown in Fig. 6.11. The anaytical solution of the problem is given by Eqs. (6.29) and (6.37). The fundamental modes are sine modes so that we consider a single-mode charge density of the form

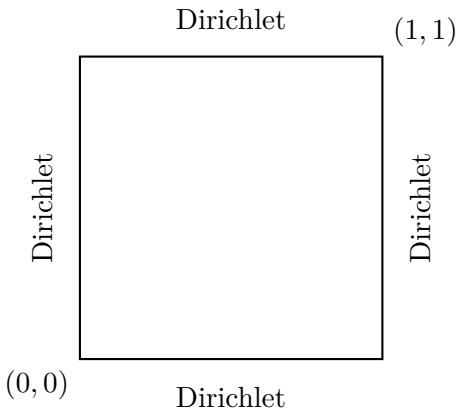


Figure 6.11: Cartesian geometry for Poisson validation of case A.

$$R(x,y) = R_{nm} \sin\left(\frac{n\pi x}{L_x}\right) \sin\left(\frac{m\pi y}{L_y}\right) \tag{6.105}$$

The potential solution for this problem has the same shape as the charge density. Higher frequencies are damped by a factor that is proportional to the square of n and m:

$$\phi(x,y) = \frac{R_{nm}}{\left(\frac{n\pi}{L_x}\right)^2 + \left(\frac{m\pi}{L_y}\right)^2} \sin\left(\frac{n\pi x}{L_x}\right) \sin\left(\frac{m\pi y}{L_y}\right) \tag{6.106}$$

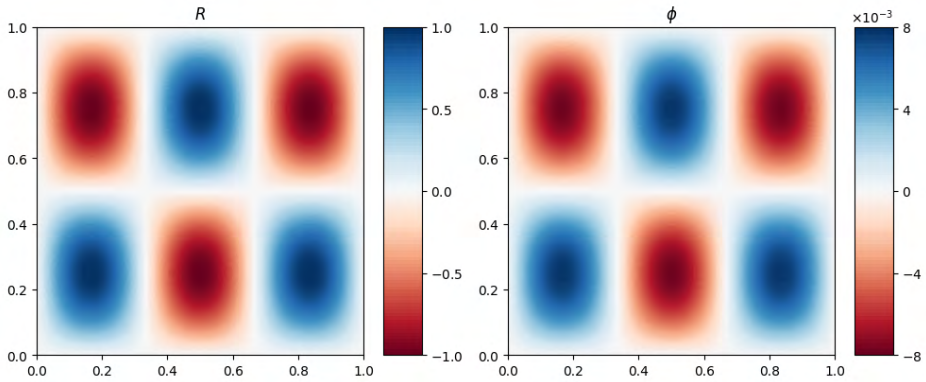


Figure 6.12: Charge density and potential (non-dimensional) with only the $(3,2)$ mode so that $R_{3,2} = 1$ for the convergence study for case A at 101×101 resolution.

We choose the $(n,m) = (3,2)$ mode with $R_{nm} = 1$ for the convergence testing of the Poisson formulation which is shown in Fig. 6.12 where the same direct linear system solver, LU, has been applied on both cases. The potential and charge density indeed have the same shape with a lower amplitude for the potential in this case. The convergence of the solution on both triangular and quadrangular meshes without correction is shown in Fig. 6.13 by taking the L^2 norm of the potential. Both element types have a second order accuracy which is satisfactory. However due to the odd-even decoupling of the quadrangular elements the

error is a factor of 5 higher for quadrangular meshes. Since the same method is used for both element types, this error is thus a result of the bad discretization for this type of elements in the chosen formulation which is more stable for triangles. The quadrangular stabilized discretization described earlier solves this issue by introducing virtual cells, resulting in the same discretization for quadrangular and triangles given by Fig. 6.4a.

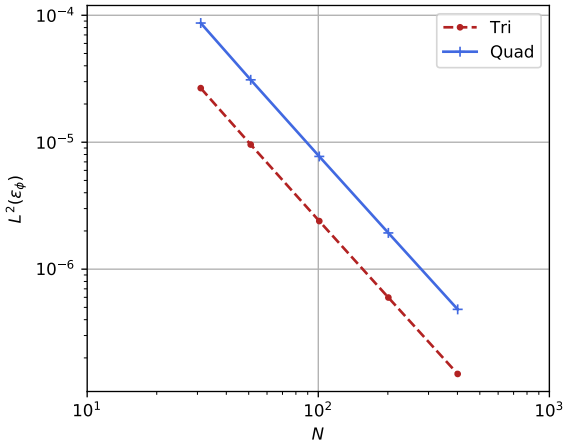


Figure 6.13: Poisson error convergence for case A.

6.4.2 Cartesian geometry

Case B considers mixed boundary conditions tested on the $[0, 1]^2$ square by switching to zero Dirichlet or zero Neumann boundary conditions for the bottom side in Fig. 6.14.

In the case where the Dirichlet boundary conditions are in the x-direction and the Neumann boundary conditions are in the y-direction the modes of the Poisson equation are

$$R(x, y) = R_{nm} \sin\left(\frac{n\pi x}{L_x}\right) \cos\left(\frac{m\pi y}{L_y}\right) \tag{6.107}$$

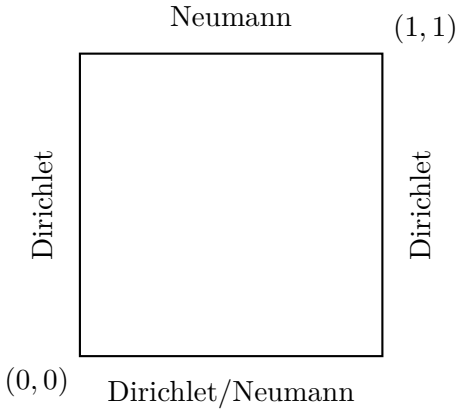


Figure 6.14: Cartesian geometry for Poisson validation of case B.

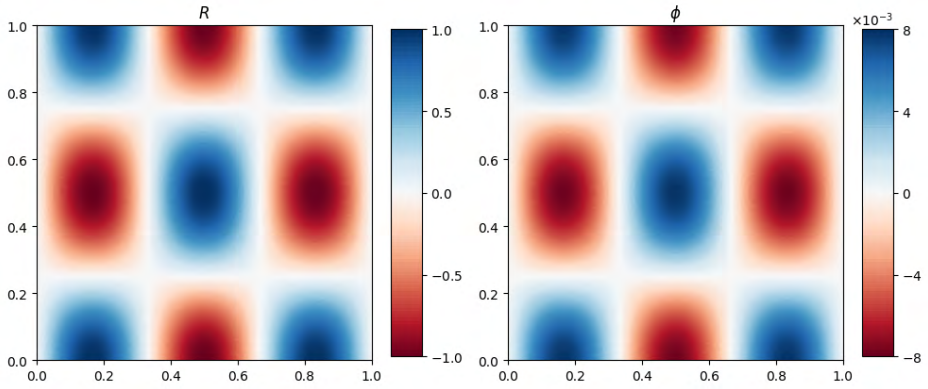


Figure 6.15: Charge density and potential for the 2-Dirichlet (left-right) 2-Neumann (bottom-top) mode for case B.

where the results for this case with $n = 3$ and $m = 2$ are shown in Fig. 6.15. Switching the bottom boundary condition to Dirichlet the modes of the Poisson equation become

$$R(x, y) = R_{nm} \sin\left(\frac{n\pi x}{L_x}\right) \sin\left((m + 1/2)\frac{\pi y}{L_y}\right) \tag{6.108}$$

and results for the potential and charge density are shown in Fig. 6.16. Both modes with mixed boundary conditions are correctly captured as their shape is left unchanged validating the implementation of mixed boundary conditions in cartesian geometries.

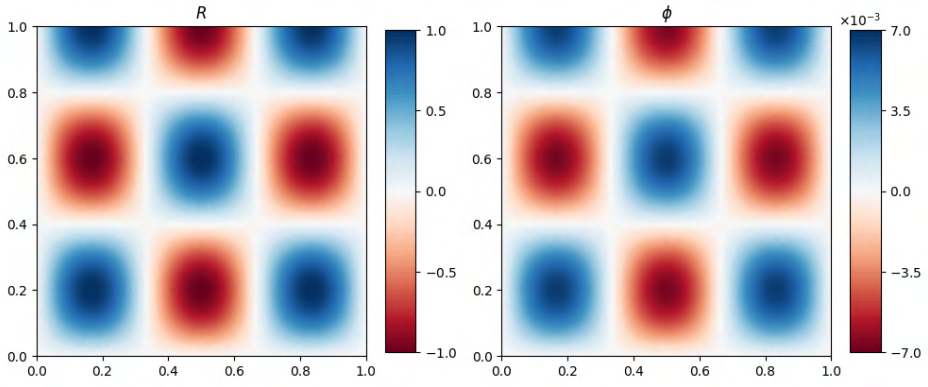


Figure 6.16: Charge density and potential for the 3-Dirichlet 1-Neumann (top) mode for case B.

We finally consider non-zero boundary conditions propagation of the AVIP Poisson solver: a non-zero potential is applied without any charge density so that it is the Laplace equation that is actually solved. The simple test case of a constant electric field in the x direction is shown in Fig. 6.17 where the potential is indeed linear in the x direction.

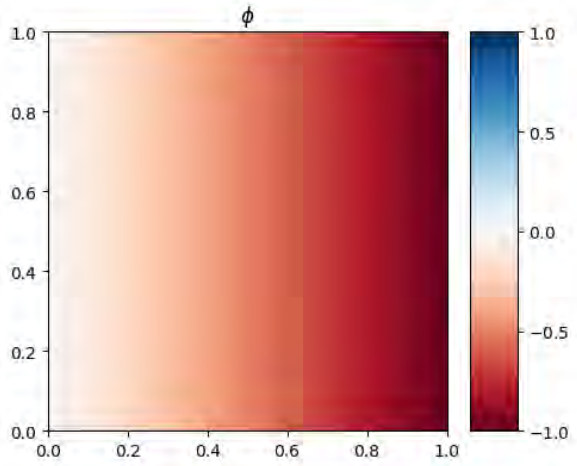


Figure 6.17: Potential for a constant electric field.

6.4.3 Cylindrical geometry

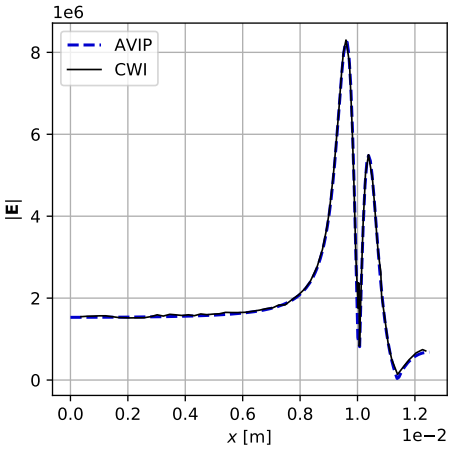
6.4.3.1 Bagheri benchmark

Axisymmetric conditions are assumed to simulate NRP discharges so that the cylindrical formulation of the Poisson equation needs to be checked. The first case is a simple square domain of 1.25×1.25 cm^2 used in the streamer code benchmark Bagheri et al. [2018] (more details in Chap. 9) where an initial gaussian seed of positive charge is applied. The solution at the first iteration is shown in Fig. 6.18 with a map of electric field norm and a 1D comparison between AVIP and the reference code from CWI in the benchmark paper. Only the first iteration is compared because different numerical schemes for plasma species are used and the solution at later times is closed but not supposed to be exactly the same. Tests have been carried out in three kind of meshes shown in Fig. 10.2: triangular, quadrangular and hybrid meshes for a number of nodes ranging from 10^5 to 10^6. The agreeement found between AVIP and the benchmark code validates the axisymmetric formulation of the Poisson equation in rectangular geometries.

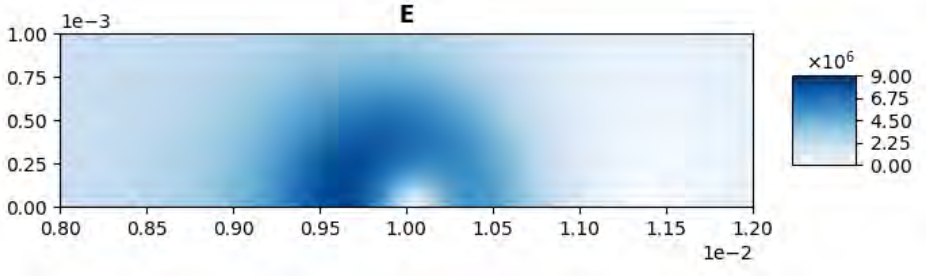
6.4.3.2 Hyperbole electrodes

We now turn to hyperbole electrodes that are more representative of the targeted configurations of this thesis. The propagation of isopotentials inside the domain in cylindrical geometry is thus evaluated. A sketch of such a pin-pin configuration as well as the boundary conditions applied is shown in Fig. 6.19: Dirichlet boundary conditions are applied on both electrodes and Neumann boundary conditions are imposed on the outlet and axis. To take such electrodes into account in structured meshes immersed boundaries can be used [Celestin, 2008, Chap. 4]. In the unstructured framework of AVIP, these electrodes are body-fitted and the mesh around one tip of the electrodes is shown in Fig. 6.19b.

The results of the AVIP Poisson solver are shown in Figs. 6.20 and 6.21 for the potential and the electric field, respectively. The equipotentials are well propagated inside the domain and result in a fan-like shape. Near the tip of the electrodes the electric field is amplified

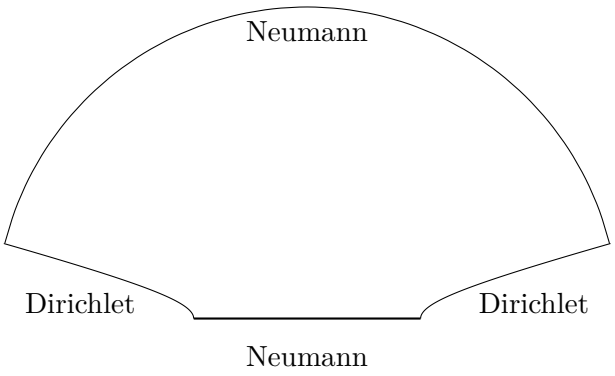


(a) 1D cut

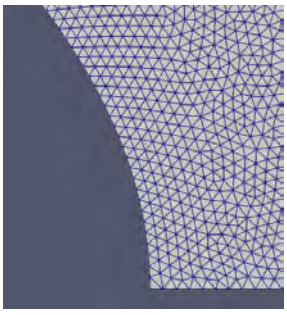


(b) 2D map (Zoom)

Figure 6.18: Electric field for the first iteration of the benchmark Bagheri et al. [2018]. Comparison of the 1D cut is taken from Fig. 4 of the paper.



(a) Poisson boundary conditions.



(b) Mesh around a hyperbolic electrode.

Figure 6.19: Hyperbole electrodes configuration.

due to the shortest distance between the equipotentials as shown in Fig. 6.21.

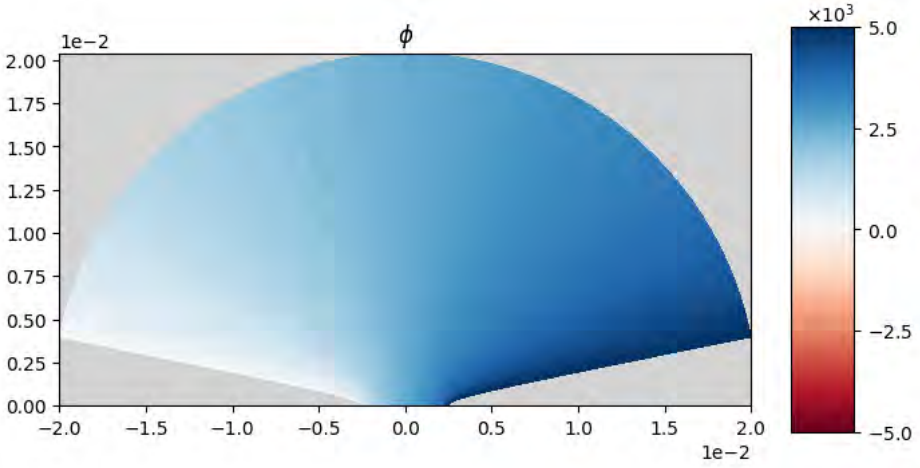


Figure 6.20: Potential for a pin-pin hyperboles configuration.

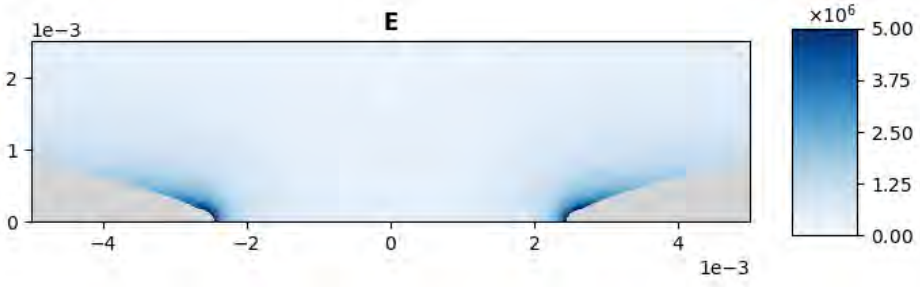


Figure 6.21: Electric field for hyperbole electrodes.

In Celestin [2008], Tholin [2012], Dirichlet conditions from the analytical solution of the Laplace equation for hyperbolic electrodes are imposed on the boundaries of a structured rectangular mesh which uses immersed boundaries to model the electrodes. Real pin electrodes are not perfect hyperboles and have a constant height so that this approach is limited to a particular geometry: for a real electrode exact solutions of the Laplace equation are not easily retrieved. On the other hand, in AVIP we leverage the unstructured framework of the code and impose Neumann boundary conditions in the farfield region allowing to adapt to any shape of electrodes. That step would be very costly for a structured code as coarsening the mesh is less effective. By coarsening the mesh thanks to triangles, the farfield region can be set far away to not have any influence between the electrodes at a low cost. This is illustrated in Fig. 6.22 where more realistic delta-shaped electrodes are used in the simulation: the fan-shape potential is retrieved as well.

6.4.4 PETSc solvers

As described earlier in this chapter, a variety of preconditioners and linear solvers are available in PETSc and a partial list is shown in Tab. 6.2. The most performant option for each

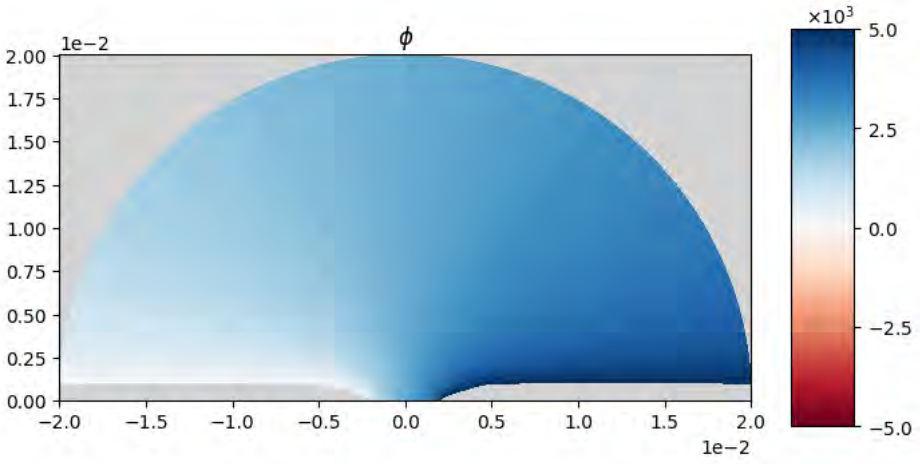


Figure 6.22: Potential for a pin-pin delta electrodes configuration.

configuration is now discussed.

When the quadrangular discretization is not stabilized as shown in Fig. 6.7, iterative solvers are not able to converge for high number of nodes and only a direct LU solver.

For the stabilized quadrangular discretization and triangle discretization (which are the same) in cartesian coordinates, the inner part of the matrix is symmetric. Boundary conditions break this symmetry and this is not directly corrected in AVIP. However since the matrix is non-symmetric only due to boundary conditions it can be considered not too non-symmetric so that in practice iterative methods like CG are able to converge to the solution really fast. The most performant combination of preconditioner and iterative solver in our case are the native PETSc GAMG preconditioner with CG iterative solver where the default Chebychev solver for the multigrid levels is used.

In axisymmetric conditions, the symmetry of the inner matrix is broken and the matrix is completely non-symmetric. The combination of CG with GAMG is still performant if the iterative solver for the multigrid levels is switched to a Richardson solver since the Chebychev solver only works for symmetric matrices. The GAMG preconditioner seems to stabilize the system a lot since the CG method is able to converge to the solution although the system is non-symmetric. Other preconditioners applied with CG do not lead to convergence and we must switch to the BiCG and BiCGStab [Saad, 2003] which are versions of CG that get rid of the symmetry constraint to get convergence.

6.5 Photoionization

Photoionization, as discussed in Section. 2.4.4, can be modeled by a set of Helmholtz equations. By doing so the photoionization source term becomes a sum of three source terms:

$$S_{ph}(\mathbf{r}) = \sum_j S_{ph}^j(\mathbf{r}) \tag{6.109}$$

$$S_{ph}^j(\mathbf{r}) = \int_{V'} \frac{I(\mathbf{r}')A_j p_{O_2} e^{-\lambda_j p_{O_2} R}}{4\pi R} dV' \tag{6.110}$$

where $R = |\mathbf{r} - \mathbf{r}'|$ and each $S_p h^j$ satisfies the following equation:

$$\nabla^2 S_{ph}^j(\mathbf{r}) - (\lambda_j p_{O_2})^2 S_{ph}^j(\mathbf{r}) = -A_j p_{O_2}^2 I(\mathbf{r}) \tag{6.111}$$

The resolution of these equations follow exactly the same methodology as for the Poisson equation. The number of linear systems solved is equal to two or three depending on the model used and resolution is performed every 10 iterations to keep the cost low while still preserving accuracy [Celestin, 2008].

Chapter 7

Scalar transport equations

Contents

This chapter concerns the numerical integration of the scalar transport equations relevant to the drift-diffusion modeling of plasma species where each species i is governed by the following equation:

$$\frac{\partial n_i}{\partial t} + \nabla \cdot \mathbf{\Gamma}_i = S_{0i} \qquad \text{with } \mathbf{\Gamma}_i = n_i \mu_i \mathbf{E} - D_i \nabla n_i \tag{7.1}$$

General properties and canonical solutions of the drift-diffusion equations are first recalled in Section. 7.1. Robust numerical integration of these equations is then presented in Section. 7.2 with the introduction of limiters. Finally the validation of the implemented schemes is carried out in Section. 7.3.

7.1 Canonical solutions

7.1.1 Convection

Let us consider a multidimensional scalar field $u(\mathbf{x}, t)$. For a constant advection velocity $\mathbf{a}$, the convection problem reads:

$$\begin{cases} \dfrac{\partial u}{\partial t} + \boldsymbol{\nabla} \cdot (\mathbf{a}u) = 0 & (7.2) \\ u(\mathbf{x}, 0) = u_0(\mathbf{x}) & (7.3) \end{cases}$$

The method of characteristics [Toro, 2009, Chap .2] allows to retrieve the solution to the constant linear advection problem:

$$u(\mathbf{x}, t) = u(\mathbf{x} - \mathbf{a}t, 0) = u_0(\mathbf{x} - \mathbf{a}t) \tag{7.4}$$

7.1.2 Diffusion

Let us consider a one-dimensional scalar field $u(x, t)$. For a constant diffusion coefficient ν the diffusion problem reads

$$\begin{cases} \dfrac{\partial u}{\partial t} = \nu \dfrac{\partial^2 u}{\partial x^2} & (7.5) \\ u(x, 0) = u_0(x) & (7.6) \end{cases}$$

1D Resolution Applying the Fourier transform on Eq. (7.5):

$$\frac{\partial \hat{u}}{\partial t} = -\nu k^2 \hat{u} \tag{7.7}$$

$$\implies \hat{u}(k, t) = \hat{u}(k, 0)\mathrm{e}^{-\nu k^2 t} \tag{7.8}$$

$$\implies u(x, t) = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} \hat{u}(k, 0)\mathrm{e}^{-\nu k^2 t}\mathrm{e}^{\mathrm{i}kx} dk = \mathcal{F}^{-1}(\hat{u}(k, 0)\mathrm{e}^{-\nu k^2 t}) \tag{7.9}$$

From this formula if the initial solution is a Gaussian:

$$u(x, 0) = u_0 \exp\left(-\frac{x^2}{\sigma^2}\right) \tag{7.10}$$

then the solution of the diffusion problem is

$$u(x, t) = \frac{u_0 \sigma}{\sqrt{\sigma^2 + 4\nu t}} \exp\left(-\frac{x^2}{\sigma^2 + 4\nu t}\right). \tag{7.11}$$

Multidimensional resolution Adapting the preceding results, we start from the multidimensional diffusion problem:

$$\frac{\partial u}{\partial t} = \nu \nabla^2 u \tag{7.12}$$

The multidimensional Fourier transform for $f \in \mathcal{L}^2(\mathbb{R}^d)$ (where d is the dimension) is:

$$\mathcal{F}(f) : \mathbf{k} \mapsto \hat{f}(\mathbf{k}) = \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} f(\mathbf{x})\mathrm{e}^{-\mathrm{i}\mathbf{k}\cdot\mathbf{x}} dx. \tag{7.13}$$

Eq. 7.9 becomes:

$$u(\mathbf{x},t) = \mathcal{F}^{-1}(\hat{u}(\mathbf{k},0)\mathrm{e}^{-\nu\|\mathbf{k}\|^2 t}) \tag{7.14}$$

7.1.3 Convection-diffusion-reaction

We include reaction in the form of a sink term with frequency k. The multidimensional convection-diffusion-reaction equation then reads:

$$\begin{cases} \dfrac{\partial u}{\partial t} + \boldsymbol{\nabla}\cdot(\mathbf{a}u) = \nu\nabla^2 u - ku & (7.15) \\ u(\mathbf{x},0) = u_0(\mathbf{x}) & (7.16) \end{cases}$$

where the convection speed $\mathbf{a}$ and diffusion coefficient ν are constant. Making the following change of variables:

$$\tau = t \tag{7.17}$$
$$\mathbf{y} = \mathbf{x} - \mathbf{a}t \tag{7.18}$$

the advection part of the equation vanishes so that the equation reduces to

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial \tau} - \mathbf{a}\cdot\boldsymbol{\nabla}_y u - ku \tag{7.19}$$
$$\boldsymbol{\nabla}_x u = \boldsymbol{\nabla}_y u \tag{7.20}$$

We split the remaining diffusion and reaction terms by setting $u = \tilde{u}\exp(-kt)$. Hence the convection-diffusion-reaction equation becomes a diffusion equation:

$$\frac{\partial \tilde{u}}{\partial \tau} = \nu\nabla_y^2 \tilde{u} \tag{7.21}$$

Formally the solution of such a problem is then:

$$u(\mathbf{y},t) = e^{-kt}\mathcal{F}^{-1}(\hat{\tilde{u}}(\mathbf{k},0)\mathrm{e}^{-\nu\|\mathbf{k}\|^2 t}) \tag{7.22}$$

For a Gaussian initial condition the following expression can be retrieved

$$u(\mathbf{x},t) = u_0 e^{-kt}\prod_{i=1}^{d}\frac{\sigma_i}{\sqrt{\sigma_i^2 + 4\nu t}}\exp\left[-\frac{(x_i - a_i t)^2}{\sigma_i^2 + 4\nu t}\right] \tag{7.23}$$

This solution shows the interplay between the advection term which transports the solution, the diffusion term which damps the amplitude and increase the width of the solution the sink source term which exponentially damps the solution.

7.2 Numerical integration of scalar transport equations

7.2.1 Consistency, stability and convergence for transport equations

The general definitions of consistency, stability and convergece presented in Section. 5.1.1 are adapted in the particular context of transport equations which are partial differential equations in time and space.

7.2.1.1 Consistency

A numerical scheme $N(U_i^n) = 0$ is said to be consistent if it tends to the differential equation $D(U) = 0$ when time and space steps tend to zero:

$$\lim_{\Delta x \to 0, \Delta t \to 0} N(U_i^n) = D(U_i^n) \tag{7.24}$$

7.2.1.2 Stability

Stability is the requirement that all errors must remain bounded when the temporal iteration process advances. Denoting by u_i^n the computed solution and $\bar{u}_i^n$ the exact solution of the numerical scheme, we define the error as $\bar{\varepsilon}_i^n = u_i^n - \bar{u}_i^n$ and require:

$$\lim_{n \to +\infty} |\bar{\varepsilon}_i^n| \leq K \tag{7.25}$$

at fixed Δt with K independent of n.

7.2.1.3 Convergence

Convergence is a condition on the numerical solution: when time and space steps tend to zero, the numerical solution u_i^n must converge to the exact solution $\tilde{u}_i^n$ of the differential equation. Defining the error as $\tilde{\varepsilon}_i^n = u_i^n - \tilde{u}_i^n$ we require that

$$\lim_{\Delta x \to 0, \Delta t \to 0} \max_{n \in [\![0, T/\Delta t]\!], i \in [\![0, L_x/\Delta x]\!]} |\tilde{\varepsilon}_i^n| = 0 \tag{7.26}$$

7.2.2 Advection-diffusion schemes

The Scharfetter-Gummel [Scharfetter and Gummel, 1969] advection-diffusion scheme and its improved version [Kulikovsky, 1995] are presented in this section. These schemes are widely used in plasma streamer physics Celestin [2008], Tholin [2012], Sharma et al. [2018], Bagheri et al. [2018], Kantner [2020] for their combined robustness and accuracy.

7.2.2.1 Derivation

We consider the general advection diffusion equation

$$\frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} = 0 \quad \text{where} \quad F = Vu - D\frac{\partial u}{\partial x} \tag{7.27}$$

Placing ourselves in a mesh between x_i and x_{i+1} we assume that the velocity V, diffusion coefficient D and flux F are constant along the line segment $(i, i+1)$. Hence the drift-

diffusion flux definition $F = Vu - D\frac{\partial u}{\partial x}$ becomes a first order differential equation between x_i and x_{i+1}:

$$\frac{\partial u}{\partial x} - \frac{V_{i+1/2}}{D_{i+1/2}}u = -\frac{F_{i+1/2}}{D_{i+1/2}} \tag{7.28}$$

The solution along the edge $(i, i+1)$ is given by

$$u(x) = \left[u_i - \frac{h_i F}{D}\int_0^{\xi} \exp(-\alpha\xi')\mathrm{d}\xi'\right]\exp(\alpha\xi) \tag{7.29}$$

where

$$\xi = \frac{x - x_i}{h_i} \quad \alpha = \frac{h_i V_{i+1/2}}{D_{i+1/2}} \quad h_i = x_{i+1} - x_i \tag{7.30}$$

Finally evaluating the solution at x_{i+1} gives the desired unknown flux at the middle of the edge:

$$F_{i+1/2} = \frac{V_{i+1/2}}{1 - e^{\alpha}}[u_{i+1} - e^{\alpha}u_i] \tag{7.31}$$

This flux definition is the one from the original Scharfetter and Gummel [1969] scheme. This combined convection-diffusion scheme has a non-dimensional parameter α which compares the amount of convection against diffusion at a given edge. Asymptotic regimes of this edge flux are given in the following for limit values of α:

Dominant convection $\alpha \gg 1$. The flux reduces to a first order upwind for the convection part

$$F_{i+1/2} = V_{i+1/2}u_i \tag{7.32}$$

Dominant diffusion $\alpha \ll 1$. The flux reduces to centered difference for the diffusive flux and convective flux

$$F_{i+1/2} = -\frac{D_{i+1/2}}{h_i}(u_{i+1} - u_i) + V_{i+1/2}\frac{u_{i+1} + u_i}{2} - \frac{\alpha}{12}V_{i+1/2}(u_{i+1} - u_i) \tag{7.33}$$

7.2.2.2 Improved version of the SG scheme

From this asymptotic analysis the scheme reduces to first order when convection is dominant. To gain order an improved version [Kulikovsky, 1995] can be obtained by first assuming a linear velocity profile between x_i and x_{i+1}

$$V(x) = V_i(1 + \frac{\Delta V_i}{V_i}\xi) = V_i(1 + 2\beta\xi) \tag{7.34}$$

with $\beta = \Delta V_i/2V_i$. Solving the first order differential equation as above with the linear profile and assuming $|\alpha\beta| \ll 1$, *i.e.* a sufficiently small speed variation, allows to get the

following flux expression [Kulikovsky, 1995]:

$$F_{i+1/2} = \frac{D_{i+1/2}}{h_i I_1}\left[u_i - \frac{\exp(-\alpha\xi)}{1+\alpha\beta}u_{i+1}\right] \tag{7.35}$$

with

$$\xi = \frac{x - x_i}{h_i} \tag{7.36}$$

$$f(\xi) = \alpha(1+\beta\xi)\xi \tag{7.37}$$

$$\alpha = \frac{V_i h_i}{D_{i+1/2}} \tag{7.38}$$

$$I_1 = \int_0^1 \exp(-\alpha\xi')(1-\alpha\beta\xi'^2)\mathrm{d}\xi'. \tag{7.39}$$

7.2.2.3 Virtual nodes

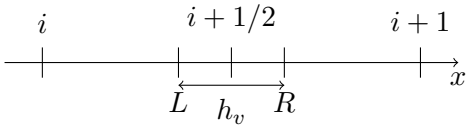


Figure 7.1: Sketch of the virtual nodes used for the improved SG scheme.

To satisfy $|\alpha\beta| \ll 1$, *i.e.* a small enough velocity fluctuation, virtual nodes are defined on the edge. Velocity is assumed to vary linearly and the interpolation distance is controled by a paremeter ε: the lower the value the more the values are interpolated close to the middle of the edge. A sketch of the interpolation procedure is drawn in Fig. 7.1 and the different interpolated variables read

$$x_{L,R} = x_{i+1/2} \mp \frac{h_v}{2} \tag{7.40}$$

$$V_{L,R} = V_{i+1/2} \mp \frac{\Delta V_v}{2} \tag{7.41}$$

$$\Delta V_v = \frac{h_v}{h_i}\Delta V_i \tag{7.42}$$

$$h_v = \sqrt{\frac{\varepsilon 2 D_{i+1/2} h_i}{|\Delta V_i|}} \tag{7.43}$$

The interpolated scalar values $u_{L,R}$ are not linear functions of the position but are exponentially interpolated to yield:

$$u(x) = u_i \exp(a(x - x_i)), \quad a = \frac{1}{h_i}\ln\left(\frac{n_{i+1}}{n_i}\right) \tag{7.44}$$

7.2.3 Total Variation Diminushing High-Resolution Schemes

The SG scheme is an advection-diffusion scheme that uses an interpolation procedure to capture stiffness. As will be seen in the following sections, its applicability on non-regular elements, such as distorted triangles, is not possible and we turn to other classes of the schemes for this reason. The general development of Total Variation Diminushing (TVD) High-Resolution Schemes is detailed for 1D structured meshes following [Hirsch, 2007, Chap. 8] for the linear advection equation in this section. Conditions for high-resolution schemes are first detailed for a two-step numerical scheme. Monotonicity is then defined and the important Godunov theorem [Godunov and Bohachevsky, 1959] is recalled leading to the introduction of limiters [van Leer, 1974, Sweby, 1984].

7.2.3.1 High-Resolution Schemes

We consider two-step numerical schemes which have the following general form in 1D meshes

$$u_i^{n+1} = \sum_j b_j u_{i+j}^n \tag{7.45}$$

The range of j is called the support of the schemes and is separated between ju upwind points and jd downwind points so that the total number of support points is $M = ju+jd+1$. We recall the first order upwind scheme and second order Lax and Wendroff [1960] schemes for the linear convection equation in terms of the CFL number $\sigma = a\Delta t/\Delta x$ which must be less than or equal to 1 for stability:

$$\text{FOU} \qquad u_i^{n+1} = u_i^n - \sigma(u_i^n - u_{i-1}^n) \tag{7.46}$$

$$\text{LW} \qquad u_i^{n+1} = u_i^n - \frac{\sigma}{2}(u_{i+1}^n - u_{i-1}^n) + \frac{\sigma^2}{2}(u_{i+1}^n + u_{i-1}^n - 2u_i^n) \tag{7.47}$$

General conditions to obtain a scheme of given order of accuracy p can be obtained and the reader is referred to [Hirsch, 2007, Chap. 8] for the proof:

$$\sum_j b_j j^m = (-\sigma)^m \;\; \forall\, m \in [\![0, p]\!] \tag{7.48}$$

7.2.3.2 Monotonicity conditions

A scheme is said to be monotone or total variation diminushing (TVD) if the new solution value at time $(n+1)$, u_i^{n+1}, does not reach values outside the range covered by u_{i+j}^n, that is

$$u_{\min}^n \leq u_{i+j}^n \leq u_{\max}^n \implies u_{\min}^n \leq u_i^{n+1} \leq u_{\max}^n \tag{7.49}$$

A sufficient condition for monotocity is that $b_j \geq 0$ for all j in the support of the scheme [Hirsch, 2007, Chap. 8]. An equivalent way of looking at monotonicity is to see it as the requirement that a local minimum cannot decrease and a local maximum cannot increase when going from one timestep to the other. Rewriting the two-step numerical scheme Eq. (7.45) with $\sum_j b_j = 1$:

$$u_i^{n+1} = u_i^n + \sum_j b_j (u_{i+j}^n - u_i^n) \tag{7.50}$$

If $b_j \geq 0$ is satisfied then $u_i^{n+1} \geq u_i^n$ for a local minimum and $u_i^{n+1} \leq u_i^n$ for a local maximum, *i.e.* the solution stays within the previous solution bounds.

7.2.3.3 Godunov's theorem

Theorem 7.1 (Godunov and Bohachevsky [1959])**.** *All linear monotone schemes for the convection equation are necessarily first order accurate.*

Proof. The proof is only made for the two-step explicit schemes. Following Eq. (7.48), a second order accurate scheme must satisfy

$$\sum_j b_j = 1 \quad \sum_j j b_j = -\sigma \quad \sum_j j^2 b_j = \sigma^2 \tag{7.51}$$

We consider a first order monotone scheme. Monotonicity implies that $\forall j,\ b_j \geq 0$. Invoking Schwartz's inequality yields

$$\sigma^2 = \left(\sum_j j b_j \right)^2 = \left(\sum_j j \sqrt{b_j} \sqrt{b_j} \right)^2 \leq \sum_j j^2 b_j \tag{7.52}$$

Finally since the two vectors $(j\sqrt{b_j})$ and $(\sqrt{b_j})$ are non colinear (except for $b_j = 0$ which is not possible) then the inequality is strict and $\sigma^2 < \sum_j j^2 b_j$. The last equality for second order accuracy can not be satisfied for a first order monotone scheme. □

7.2.3.4 High-resolution schemes and the concept of limiters

The Godunov theorem tells us that TVD/monotonicity conditions are not possible with linear schemes with order higher than one. For monotone high-order schemes the concept of limiters needs to be introduced van Leer [1974], Sweby [1984]. Limiters are non-linear functions that control the ratio between successive gradients. The following steps can be applied to any high-order scheme to build a limiter:

1. Select a first-order monotone scheme (usually the upwind scheme) as reference. Express the high order scheme as the monotone scheme plus additional terms
2. Multiply the additional terms by a limiting function $\Psi(r_i)$, expressed as a function of ratios of successive gradients.
3. Express the monotonicity conditions to derive conditions on the limiters

Let us apply this methodology on the second-order Lax and Wendroff [1960] scheme Eq. (7.47). (1) Rewriting the scheme in terms of monotone terms and non-monotone terms gives

$$u_i^{n+1} = u_i^n - \underbrace{\sigma(u_i^n - u_{i-1}^n)}_{\text{monotone terms}} - \underbrace{\frac{\sigma}{2}(1-\sigma)(u_{i+1}^n - u_i^n) - \frac{\sigma}{2}(\sigma-1)(u_i^n - u_{i-1}^n)}_{\text{non-monotone terms}} \tag{7.53}$$

Limiters monitor the ratio of successive gradients $r_i = (u_i - u_{i-1})/(u_{i+1} - u_i)$ and so non-monotone terms are multiplied by limiting functions $\Psi(r_i)$. After some rearrangements the following expression is obtained:

$$u_{i+1}^n = u_i^n - \sigma\left\{1 + \frac{1-\sigma}{2}\left[\frac{\Psi(r_i)}{r_i} - \Psi(R_{i-1})\right]\right\}(u_i^n - u_{i-1}^n) \tag{7.54}$$

Requiring the terms in the brackets to be positive the limiting function must satisfy the following inequalities:

$$0 \leq \Psi(r) \leq \min(2r, 2) \tag{7.55}$$

For a linear solution $r = 1$, *i.e.* the ratio of gradients is constant, and the second order accurate schemes should be satisfied exactly hence $\Psi(1) = 1$. Moreover, though theoretically acceptable, solutions outside $\Psi = r$ and $\Psi = 1$ lead to schemes which are overcompressive (turning sine waves into square wave forms). Various limiters can then be defined but two have been chosen for implementation in AVIP:

- the van Leer [1974] limiter:
$$\Psi(r) = \frac{r + |r|}{1 + r} \tag{7.56}$$
- the Sweby [1984] limiter which takes a parameter β between 1 and 2:
$$\Psi(r) = \max(0, \min(\beta r, 1), \min(r, \beta)) \tag{7.57}$$

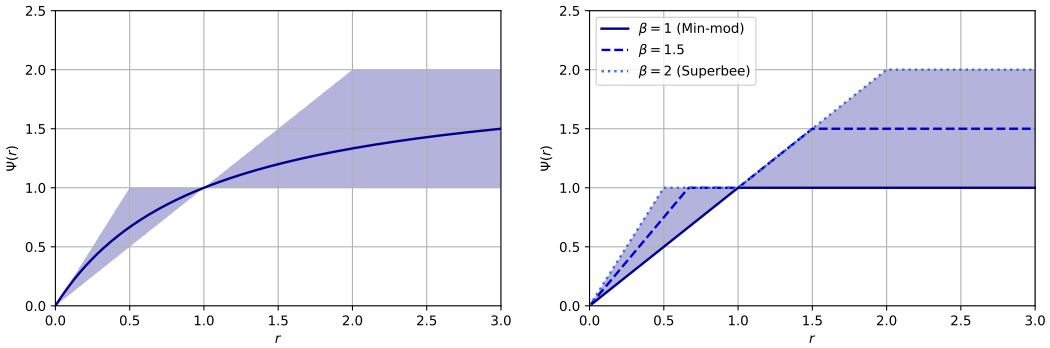
The two limiters are plotted in Sweby diagrams Fig. 7.2. The van Leer [1974] limiter is smooth and in the middle of the allowed space for limiters and can be considered as a good average option. The Sweby [1984] limiter, with its β parameter, continuously goes from the lower part of the allowed values for $\beta = 0$ to the upper part of the allowed values for $\beta = 1$. This allows flexibility on the amount of limiting that is applied: higher values are more accurate but also less stable. Note that the $\beta = 1$ and $\beta = 2$-Sweby limiters are also called Min-mod and Superbee limiters [Hirsch, 2007, Chap. 8].

7.2.4 Implementation in AVIP

The model equation that needs to be solved in the case of drift-diffusion is simply the advection-diffusion equation

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{F} = 0 \tag{7.58}$$

where $\mathbf{F} = \mathbf{V}u - D\nabla u$. This equation is averaged over the dual volume V_i.



(a) Van Leer limiter (b) Sweby limiters

Figure 7.2: Limiters in Sweby diagrams. They shaded area corresponds to the allowed space for limiters.

$$\frac{\mathrm{d}u_i}{\mathrm{d}t} + \frac{1}{V_i}\int_{V_i} \nabla \cdot \mathbf{F}\,\mathrm{d}V = 0 \tag{7.59}$$

We integrate this equation in a vertex-centered formulation looping eventually on the edges of the dual volume. To preserve data structures close to AVBP [Lamarque, 2007, Chap. 4] the following discretization is used

$$\int_{V_i} \nabla \cdot \mathbf{F}\,\mathrm{d}V = \sum_{\tau \in E(i)} \sum_{f \in \tau \cap \partial V_i} \int_f \mathbf{F} \cdot \mathbf{n}\,\mathrm{d}S \tag{7.60}$$

The first sum is a loop on each neighboring cell of a given node and then we sum on the faces of the intersection of the neighboring cell τ and the nodal surface ∂V_i. In practice in AVIP the residual is computed *at each edge* of the cells as shown in Fig. 7.3 in 2D for triangular and quadrangular cells where three and four edges are present for each element type, respectively. The number of edges differs from the number of vertices in 3D: tetrahedra elements have six edges whereas hexahedral elements have twelve edges.

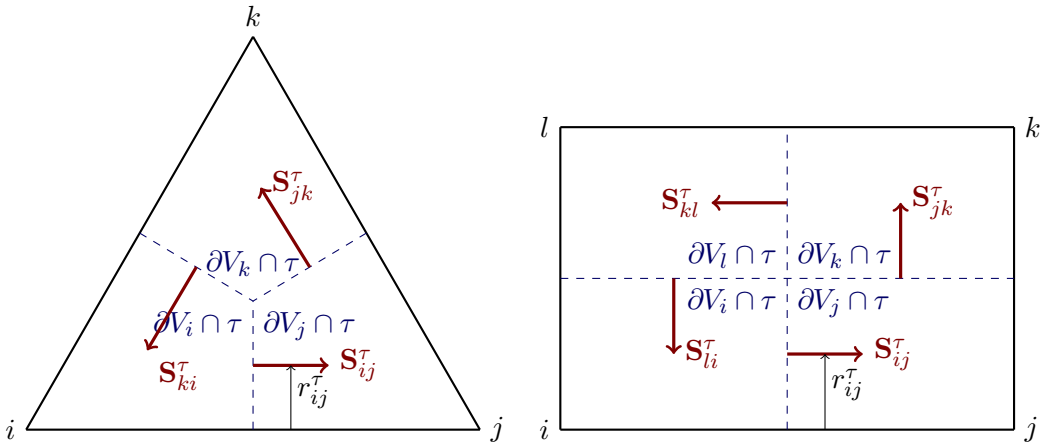
Cylindrical frames with axisymmetric conditions are also considered in this work. The flux formulation has to be slightly modified as the differential volume is now $\mathrm{d}V = r\mathrm{d}r\mathrm{d}\theta\mathrm{d}x$ where (x, r, θ) are the cylindrical coordinates. The integration on the azimuthal angle can be carried out since $\partial/\partial\theta = 0$ and therefore the averaged equation over the dual volume V_i yields

$$\frac{\mathrm{d}u_i}{\mathrm{d}t} + \frac{1}{r_i A_i}\int_{A_i} \nabla \cdot (r\mathbf{F})\,\mathrm{d}A = 0 \tag{7.61}$$

where $V_i = 2\pi r_i A_i$ results from Theorem. 5.3. Splitting this nodal surface flux along its edges yield the axisymmetric equivalent of Eq. (7.60) where the edges radiuses, shown for the edge ij in Figs. 7.3a and 7.3b, need to be included in the flux computation:

$$\int_{A_i} \nabla \cdot (r\mathbf{F})\,\mathrm{d}A = \sum_{\tau \in E(i)} \sum_{f \in \tau \cap \partial A_i} \int_f \mathbf{F} \cdot r\mathbf{n}\,\mathrm{d}l \tag{7.62}$$

The flux on the nodal surface associated to edge ij in Fig. 7.3 is finally simplified by



(a) Triangular cell (b) Quadrangular cell

Figure 7.3: Edges definitions in 2D cells. There are three edges ij, jk and ki in triangles and four edges ij, jk, kl and li in quadrangles.

invoking Theorem. 5.2:

$$\int_f \mathbf{F} \cdot r\mathbf{n}\,\mathrm{d}l = \mathbf{F}_{ij} r^\tau_{ij} \mathbf{S}^\tau_{ij} \tag{7.63}$$

where the radius r^τ_{ij} is shown in Fig. 7.3 for both elements and considering that the edge ij is an axis edge.

Three TVD schemes, whose formulation in 1D have been detailed in Sections. 7.2.2 and 7.2.3, will be presented to integrate the advection-diffusion equation in the finite volume vertex-centered formulation of AVIP: the Scharfetter Gummel (SG) scheme Scharfetter and Gummel [1969], its improved version (ISG) Kulikovsky [1995], a simple upwind scheme [Hirsch, 2007, Chap. 7] and the limited Lax-Wendroff (LLW) scheme Lax and Wendroff [1960], Sweby [1984]. The upwind and limited LW schemes are supplemented by a central differencing diffusive flux integration. These schemes has been originally developed for cartesian meshes and adaptations of these schemes on unstructured meshes with non-topologically dual elements has been carried out in AVIP and is presented.

7.2.4.1 Scharfetter Gummel scheme

Starting from (7.60), we can rewrite the flux term as

$$\int_f \mathbf{F} \cdot \mathbf{n}\,\mathrm{d}S = \mathbf{F}_{ij} \cdot \mathbf{S}_{ij} \tag{7.64}$$

Splitting the nodal surface normal into a tangential and perpendicular part $\mathbf{S}_{ij} = \mathbf{S}^{\parallel}_{ij} + \mathbf{S}^{\perp}_{ij}$ with respect to the edge direction $\hat{\mathbf{ij}} = \mathbf{ij}/ij$

$$\int_f \mathbf{F} \cdot \mathbf{n}\,\mathrm{d}S = \mathbf{F}_{ij} \cdot \mathbf{S}^{\parallel}_{ij} + \mathbf{F}_{ij} \cdot \mathbf{S}^{\perp}_{ij} \tag{7.65}$$

where a definition of these normals for non-topologically dual meshes is shown in Fig. 7.4.

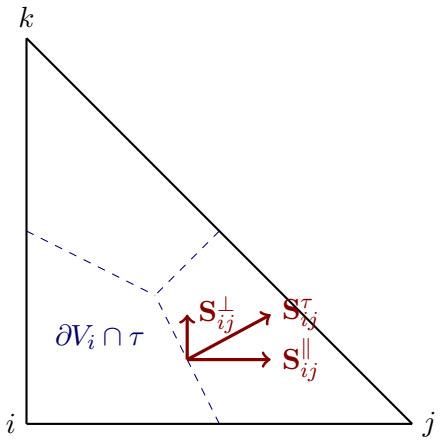


Figure 7.4: Definitions of tangential and perpendicular node surface normals in a non topologically dual triangular cell.

The Scharfetter Gummel scheme Kulikovsky [1995] can be applied on the tangential part of the flux by projecting the flux along the tangential normal:

$$\mathbf{F}_{ij} \cdot \mathbf{S}_{ij}^{\parallel} = u\mathbf{V}_{ij} \cdot \mathbf{S}_{ij}^{\parallel} - D\nabla u \cdot \mathbf{S}_{ij}^{\parallel} \tag{7.66}$$

The SG scheme gives tangential flux projection but does not give any information about the *normal* flux projection:

$$\mathbf{F}_{ij} \cdot \mathbf{S}_{ij}^{\parallel} = \mathbf{V}_{ij} \cdot \mathbf{S}_{ij}^{\parallel} \frac{u_j - e^{\alpha} u_i}{1 - e^{\alpha}} \tag{7.67}$$

where

$$\alpha = \frac{h_{ij}\mathbf{V}_{ij} \cdot \hat{\mathbf{ij}}}{D_{ij}} \tag{7.68}$$

On meshes where the perpendicular component is non-zero another scheme must be applied on top of the SG scheme. A centered-scheme has been chosen for this normal part:

$$\mathbf{F}_{ij} \cdot \mathbf{S}_{ij}^{\perp} = \mathbf{V}_{ij} \frac{u_i + u_j}{2} \cdot \mathbf{S}_{ij}^{\perp} - D_{ij} \frac{\nabla u_i + \nabla u_j}{2} \cdot \mathbf{S}_{ij}^{\perp} \tag{7.69}$$

The improved Scharfetter Gummel scheme (ISG) virtual node reconstruction is carried on the edge exactly as explained in the 1D case since locally the tangential flux reconstruction reduces to a 1D problem in Eq. (7.66).

7.2.4.2 Upwind scheme

A simple upwind scheme has been implemented for comparison purposes and its corresponding flux is given by

$$\int_f \mathbf{F} \cdot \mathbf{n} \, \mathrm{d}S = \max(0, \mathbf{S}_{ij} \cdot \mathbf{V}_{ij}) u_i + \min(0, \mathbf{S}_{ij} \cdot \mathbf{V}_{ij}) u_j \tag{7.70}$$

7.2.4.3 Limited Lax-Wendroff scheme

We are only interested in the convective part of the scheme for now so $u_t + \nabla \cdot (\mathbf{V}u) = 0$ is considered. The starting point of the Lax-Wendroff scheme is a Taylor expansion in time of the variable u [Lamarque, 2007, Hirsch, 2007]

$$u^{n+1} = u^n + \Delta t u_t + \frac{\Delta t^2}{2} u_{tt} + \mathcal{O}(\Delta t^3) \tag{7.71}$$

Keeping only the first three terms and using the advection diffusion equation the Taylor expansion is rewritten as

$$u^{n+1} = u^n - \Delta t \nabla \cdot (\mathbf{V}u) + \frac{\Delta t^2}{2} \nabla \cdot (\mathbf{V} \nabla \cdot (\mathbf{V}u)) \tag{7.72}$$

Integrating on the nodal volume V_i following (7.60) yields

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{V_i} \sum_{\tau \in E(i)} \sum_{f \in \tau \cap \partial V_i} \left[\int_f u \mathbf{V} \cdot \mathbf{n} \, \mathrm{d}S - \frac{\Delta t}{2} \int_f \nabla \cdot (\mathbf{V}u) \mathbf{V} \cdot \mathbf{n} \, \mathrm{d}S \right] \tag{7.73}$$

Assuming a constant velocity on the edge, the Lax-Wendroff schemes finally writes

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{V_i} \sum_{\tau \in E(i)} \sum_{f \in \tau \cap \partial V_i} \left[u_{ij} - \frac{\Delta t}{2} \nabla u_{ij} \cdot \mathbf{V}_{ij} \right] \mathbf{V}_{ij} \cdot \mathbf{S}_{ij} \tag{7.74}$$

where ij denotes the mean of the values at the edge, *e.g.* $u_{ij} = (u_i + u_j)/2$.

The limited Lax-Wendroff scheme Eq. (7.54) is rewritten in a 1D finite volume formulation so that the value at the middle of the edge ij is defined by

$$u_{i+1/2} = u_i + \frac{1 - \sigma}{2} (u_{i+1} - u_i) \Psi(R_i) \tag{7.75}$$

where the limiting function is either the van Leer [1974] or Sweby [1984] limiter.

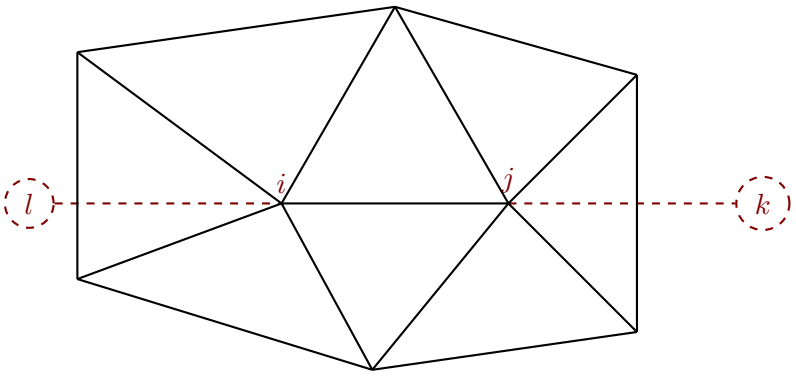


Figure 7.5: Reconstruction of neighboring nodes using upwind cells.

This cartesian limiting scheme needs to be adapted to unstructured meshes. In a cell of an unstructured mesh, there is no real upwind node so that only a virtual upwind node k

can be defined at an edge ij shown in Fig. 7.5. Only the upwind gradient $u_k - u_i$ is needed and is computed by leveraging the centered gradient of AVIP so that approximately we have

$$\nabla u_i \cdot (2\mathbf{ij}) = u_j - u_l \tag{7.76}$$

The ratio of gradients is computed as

$$R_{ij} = \frac{u_j - u_i - \nabla u_i \cdot (2\mathbf{ij})}{u_j - u_i} \tag{7.77}$$

In the end the limited Lax-Wendroff scheme in Finite Volume Vertex Centered formulation is given by

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{V_i} \sum_{\tau \in E(i)} \sum_{f \in \tau \cap \partial V_i} \left[u_i + \left(\frac{u_j - u_i}{2} - \frac{\Delta t}{2} \nabla u_{ij} \cdot \mathbf{V}_{ij} \right) \Psi(R_{ij}) \right] \mathbf{V}_{ij} \cdot \mathbf{S}_{ij} \tag{7.78}$$

7.2.4.4 Central difference diffusive flux integration

When using upwind or limited LW scheme for the advection term, a central difference scheme is applied on the diffusive flux $\mathbf{F}_D = -D\nabla u$ as

$$\int_f \mathbf{F}_D \cdot \mathbf{n} \, \mathrm{d}S = -D_{ij} \frac{\nabla u_i + \nabla u_j}{2} \cdot \mathbf{S}_{ij} \tag{7.79}$$

7.2.5 Boundary conditions in AVIP

The boundary fluxes for quandrangular elements are shown in Fig. 7.6 for a boundary node i. Two choices for the flux computation are possible

- The two fluxes are taken equal to the node flux

$$\mathbf{F}_i^{\tau_1} = \mathbf{F}_i^{\tau_2} = \mathbf{F}_i \tag{7.80}$$

- The fluxes are interpolated, which in two-dimensions reduce to

$$\mathbf{F}_i^{\tau_1} = 0.75\mathbf{F}_i + 0.25\mathbf{F}_j \tag{7.81}$$
$$\mathbf{F}_i^{\tau_2} = 0.75\mathbf{F}_i + 0.25\mathbf{F}_k \tag{7.82}$$

Two types of boundary conditions are implemented:

- Supersonic boundary conditions: $\mathbf{F} = \mathbf{F}_C + \mathbf{F}_D$
- Neumann boundary conditions: $\mathbf{F} = \mathbf{F}_C$ where $\nabla u \cdot \mathbf{n} = 0$ is thus implied

7.2.6 Time step computation

We use a cell-based approach to compute the timestep. Four time steps can be computed for each species at each cell τ:

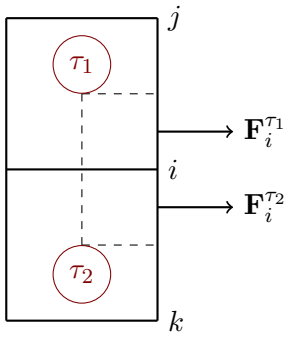


Figure 7.6: Fluxes at the boundary conditions.

- Convective time step with CFL constant

$$\Delta t_{\text{conv}} = \frac{\text{CFL}\Delta x_\tau}{||\mathbf{V}_\tau||} \tag{7.83}$$

- Diffusive time step with Fourier constant F

$$\Delta t_{\text{diff}} = \frac{\text{F}\Delta x_\tau^2}{D_\tau} \tag{7.84}$$

- Dielectric time step with contant A_D

$$\Delta t_{\text{diel}} = A_D \frac{\varepsilon_0}{\sum_i |q_i n_i \mu_i|} \tag{7.85}$$

- Chemical time step with constant A_τ

$$\Delta t_{\text{chem}} = A_\tau \max_k S_{\text{chem}}^k \tag{7.86}$$

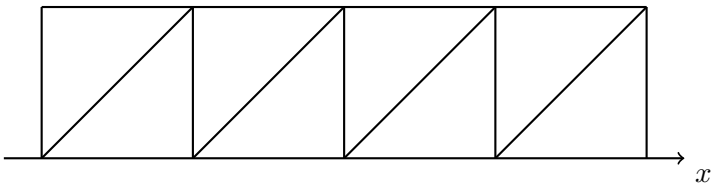
The minimum over all cells is taken as the reference timestep. The convective and diffusive time steps are computed for each species modeled with drift-diffusion equations and the resulting time step is the minimum of all the considered time steps.

7.3 Validation cases

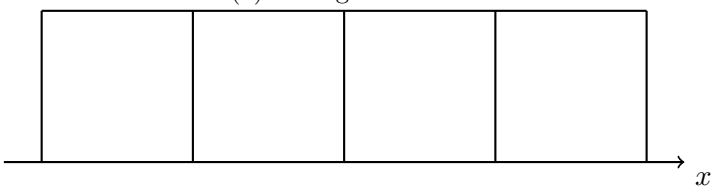
The two TVD schemes developed in this chapter, LLW and ISG, are now tested on three cases: one-dimensional linear advection, two-dimensional linear advection and a one-dimensional shock. The Sweby limiter with its parameter $\beta \in [1, 2]$ is chosen in combination with LLW: it is flexible as it introduces more accuracy at the price of robustness when increasing β. From now on, LLW-β refers to the limited Lax-Wendroff scheme with β-Sweby limiter and ISG-ε refers to the improved Scharfetter-Gummel scheme with ε parameter.

7.3.1 One dimensional advection

Scalar advection in a 1D domain that goes from -1.0 to 1.0. Quadrangular and triangular cells shown in Fig. 7.7 are used. Note that AVIP (and AVBP as well) do not have dedicated routines for one-dimensional simulations so that to simulate one-dimensional cases duplicated nodes in the y direction are always necessary. Four profiles are selected inspired from [Hirsch, 2007, Chap. 7]: a gaussian, a step and two packet waves. Periodic boundary conditions are applied at the left and right boundaries of the domain whereas a simple flux closure is applied on the top and bottom parts. Simulations are run at CFL = 0.3 and CFL = 0.7 where the same time step is applied for both triangular and quadrangular meshes at a given CFL within the time step computation of AVIP.



(a) Triangular elements



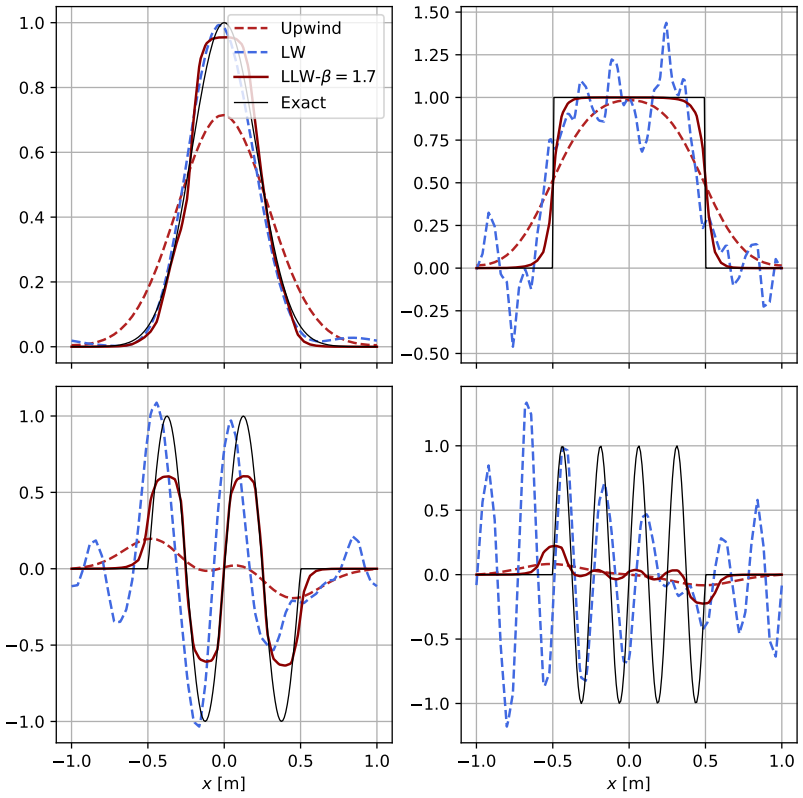
(b) Quadrangular elements

Figure 7.7: 1D domain with quadrangular and triangular meshes used for the one-dimensional advection test case.

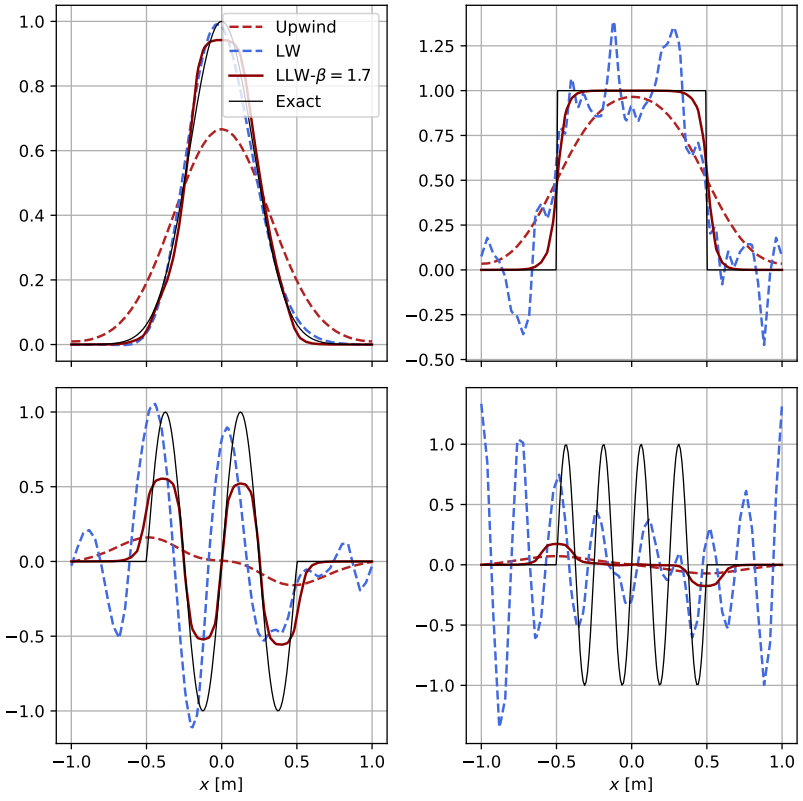
Results for the four profiles in coarse triangular and quadrangular meshes of 51×2 nodes for the Upwind, LW and LLW-β = 1.7 schemes are shown in Fig. 7.8 at CFL = 0.3 after one full round of advection *i.e.* when the profiles are back to their initial position. The diffusive character of the upwind scheme is clearly emphasized at high spatial frequencies when transporting the two packet waves. The amplitude of the packet-wave are so damped at this resolution that the initial lobes are barely visible.

The Lax-Wendroff scheme corrects this amplitude damping but introduces undesired wiggles due to its highly dispersive character so that only the gaussian profile is correctly convected due to the absence of discontinuities. The step and packet-waves, on the other hand, produce high amplitude wiggles after one round. Introduction of limiters allows to dissipate those wiggles by monitoring the gradient variations. The LLW-β = 1.7 scheme maintains high accuracy for the step profile while still preserving a reasonable amplitude for the two-wave packet in both triangular and quadrangular elements. Due to the coarse mesh (only 6 points per period), the four-wave packet is still substantially damped.

The influence of the limiter on the solution accuracy is shown in Fig. 7.9 where the value of the β parameter has been varied from 1 to 2 in quadrangular and triangular elements. The gain in accuracy can be clearly observed as the peak amplitude gets closer to the exact solution when β is increased from 1 to 2 in all cases.

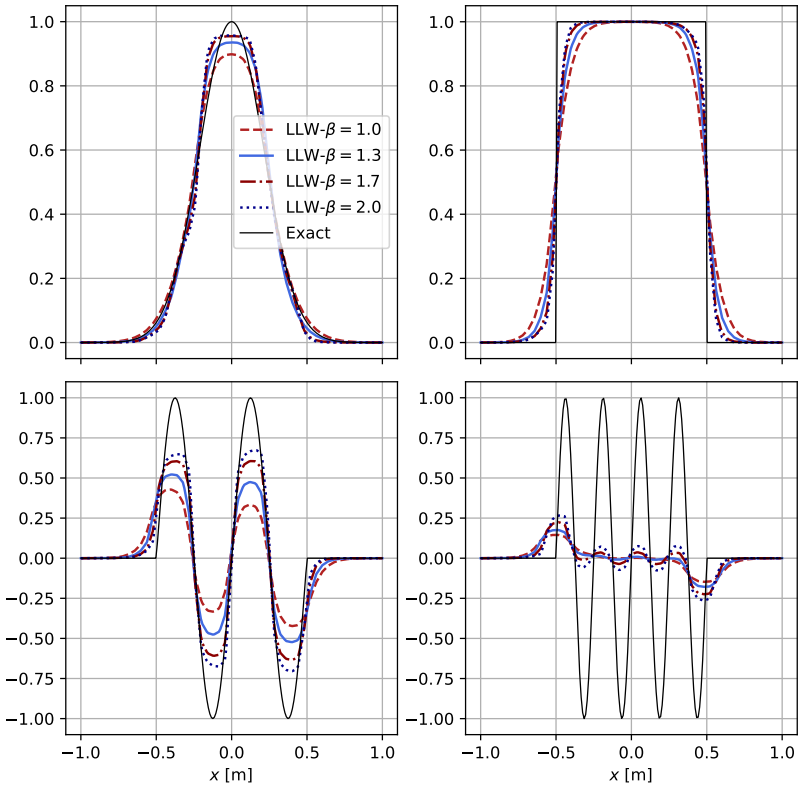


(a) Triangular elements

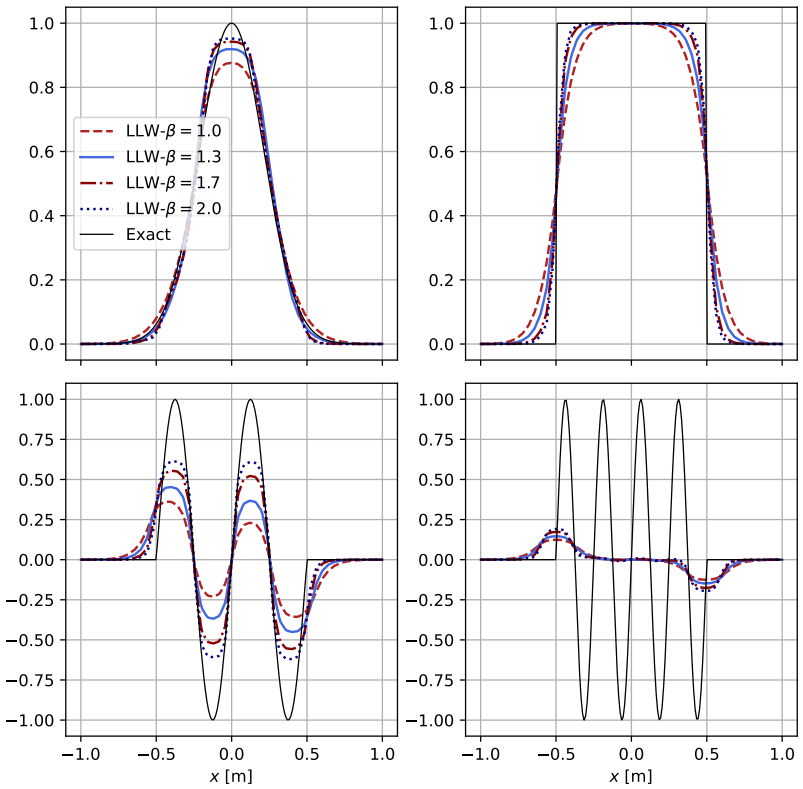


(b) Quadrangular elements

Figure 7.8: One-dimensional advection of the scalar variable u after one full round for four different inital conditions (gaussian, step, 2-packetwave, 4-packetwave) at 51×2 resolution and CFL $= 0.3$ using Upwind, LW and LLW schemes.



(a) Triangular elements



(b) Quadrangular elements

Figure 7.9: One-dimensional advection of the scalar variable u after one full round for four different inital conditions (gaussian, step, 2-packetwave, 4-packetwave) at 51×2 resolution and CFL $= 0.3$ using LLW-β with $\beta = 1$, 1.3, 1.7 and 2.0.

Increasing the CFL to 0.7 leads to the results shown in Fig. 7.10 where there is overall less diffusion than at CFL = 0.3. The Sweby limiter starts to produce oscillations for $\beta = 1.7$ and $\beta = 2.0$ for the step and packet-wave profiles advection. It can be noted that using $\beta = 2.0$ the maximum amplitude of the packet-waves reaches value above the exact solution. On top of those wiggles, the shape of the solution is disorted for high-β and the limiter becomes *over-compressive*.

From this one-dimensional study, the benefits of limiters are shown to make wiggles disappear from the original LW scheme. The value of the limiting parameter β is critical and needs to be adapted for each case: the higher the value of β the less diffusive the scheme is but less robust it is as well. A value between [1.0-1.6] in all cases seem to be reasonable, high values of β are discarded for their over-compressive property.

7.3.2 Two dimensional advection

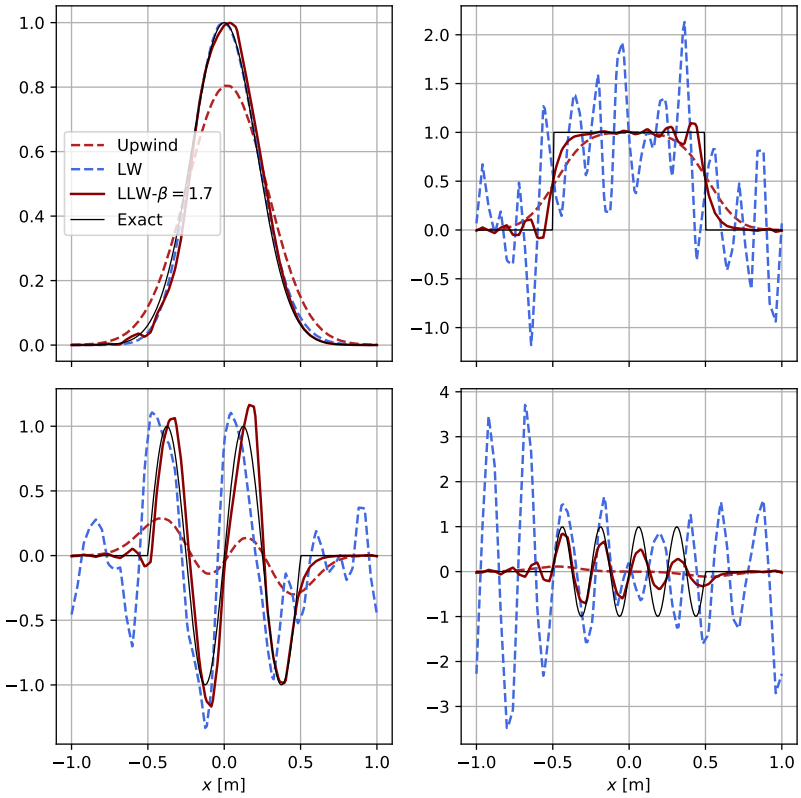
The previous one-dimensional cases are extended to two dimensions by considering square domains with quadrangular and triangular elements shown in Fig. 7.11. The initial condition of the gaussian, step and packet-waves in two-dimensions is illustrated in Fig. 7.11. Note that both the scalar variable u and the lengths can be considered non-dimensional.

Results for upwind, LW and LLW-$\beta = 1.3$ schemes are shown in Figs. 7.12, 7.13, 7.14 for the quadrangular mesh of Fig. 7.11b at 101×101 resolution. Similar conclusions apply for triangular meshes (not shown). The same colorbars for each case are used for comparison and grey areas correspond to values outside the colorbar. The same behaviors and conclusions from the one-dimensional study apply in two dimensions: the upwind scheme dissipates too much which is shown especially for the packet-wave. The Lax-Wendroff schemes produces oscillations that are particularly strong in the 2D step test case. These oscillations, as the 1D case, are also observed at the tail and head of the packet-waves. These oscillations are corrected by the limited Lax-Wendroff scheme shown in Fig. 7.14 where the step is correctly transported and magnitudes of the profiles are all below the initial values, showing the TVD property of the scheme in multi-dimensions. As for the one-dimensional case, the amplitude of the high-frequency packet-wave is still substantially damped.

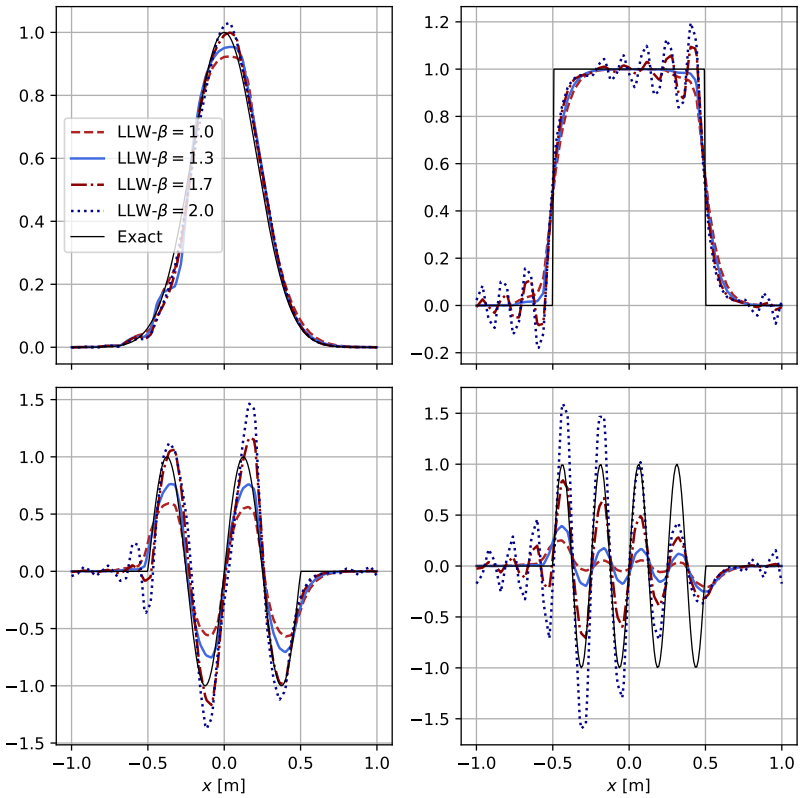
Order of the schemes The order of the schemes for the Gaussian and step profiles in quadrangular meshes is computed and results are shown in Fig. 7.15. One full round of the fully periodic domain has been simulated to get the L^2 error of the scalar variable. The limited Lax-Wendroff scheme outperforms the classical Upwind scheme in both cases and the effect of the limiter value β is more pronounced for the step profile as it allows to retrieve discontinuities better.

7.3.3 One-dimensional stiff advection

The Scharfetter-Gummel (SG) numerical scheme on all the above cases behaves like the upwind scheme as shown by the asymptotic analysis. The benefit of virtual nodes interpolation can not be seen on those cases as the advection speed is constant. A case representative of plasma discharges is now studied to validate the SG and ISG-ε schemes implementation. This test case is taken from Kulikovsky [1995] where a drift-diffusion model in a 1D domain

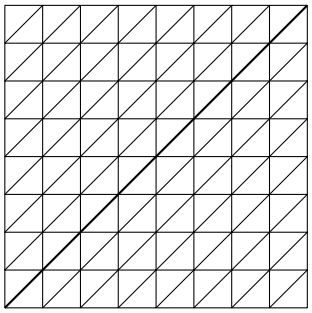


(a) Upwind, LW and LLW-$\beta = 1.7$

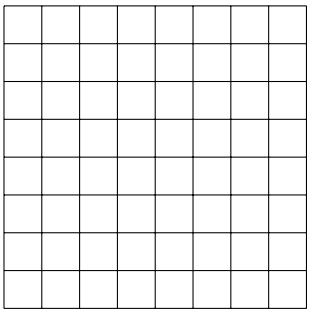


(b) LLW-$\beta = 1$, 1.3, 1.7 and 2.0

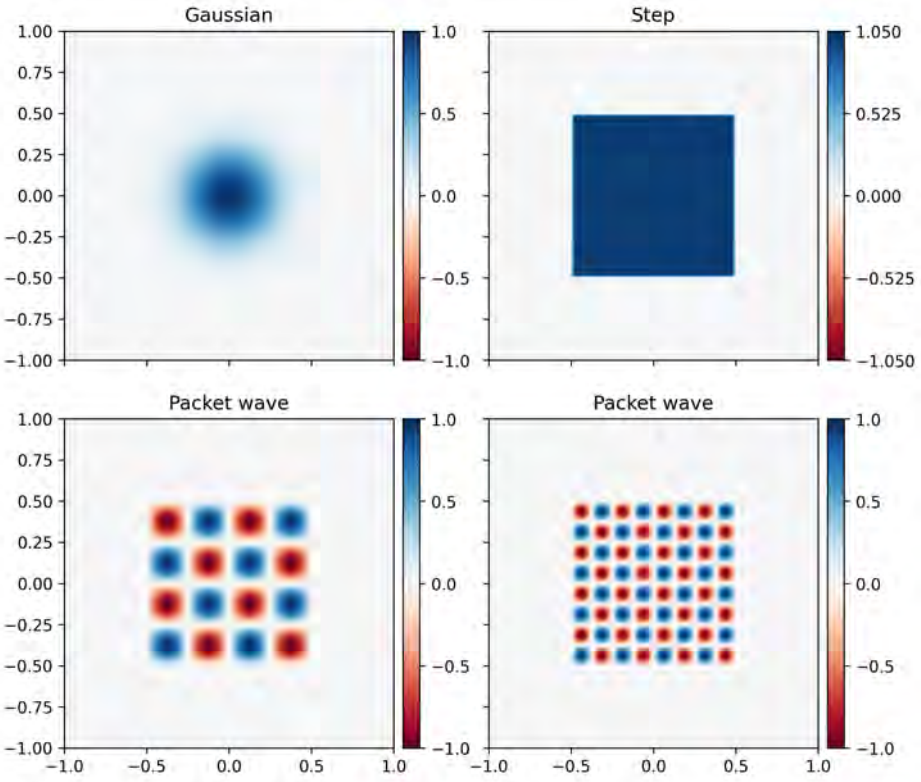
Figure 7.10: One-dimensional advection of the scalar variable u after one full round for four different inital conditions (gaussian, step, 2-packetwave, 4-packetwave) at CFL = 0.7 in quadrangular elements using various schemes.



(a) Triangular elements



(b) Quadrangular elements



(c) Initial profiles for gaussian, step and packet-waves

Figure 7.11: 2D domain $[-1, 1]^2$ used for 2D constant advection test case of the scalar variable u with (a) quadrangular and (b) triangular meshes using (c) four different profiles.

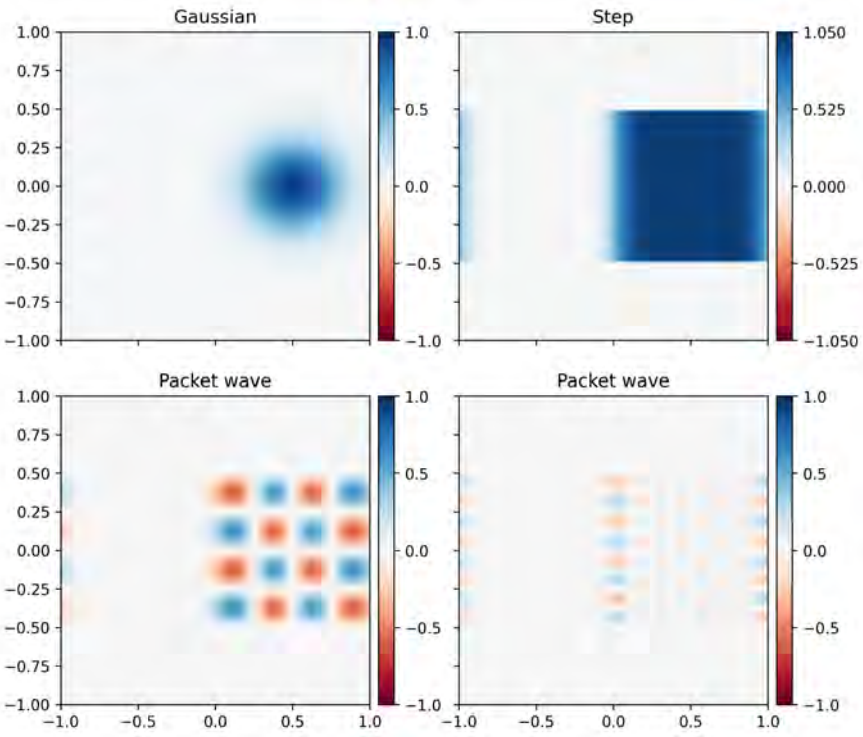


Figure 7.12: Two-dimensional advection test case results in 2D domain $[-1, 1]^2$ for scalar variable u for four profiles using upwind scheme at CFL = 0.3 in a 101×101 quadrangular mesh.

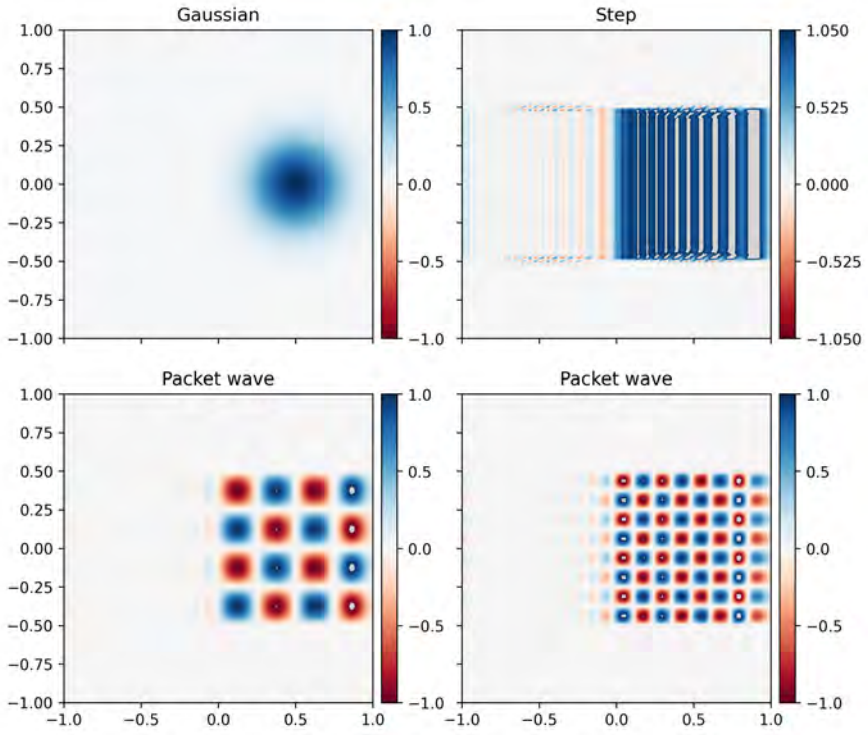


Figure 7.13: Two-dimensional advection test case results in 2D domain $[-1,1]^2$ for scalar variable u for four profiles using LW scheme at CFL $= 0.3$ in a 101×101 quadrangular mesh.

$[0,1]$ is considered with a linear electric field $E(x) = Ax$. Mobility and diffusion coefficients are equal to 1 so that the studied equation is

$$\frac{\partial n}{\partial t} - \frac{\partial Axn}{\partial x} - \frac{\partial^2 n}{\partial x^2} = 0 \tag{7.87}$$

The coefficient A is set to 10^4 so that advection is dominant and the effective equation can be considered without the diffusion term. The analytical solution is given by the method of characteristics to yield

$$n(x,t) = n_0(xe^{At})e^{At} \tag{7.88}$$

where n_0 is the initial solution profile. The initial profile is advected to the left and compressed so that its standard deviation decreases. In plasma simulations low and high values of densities, sometimes several orders of magnitude apart, need to be well transported. Hence the following profile with amplitude n_2 and background n_1 is chosen:

$$n_0(x) = n_1 + \frac{1}{2}\left[1 + \tanh\left(\frac{x - x_0}{\sigma}\right)\right] n_2 \tag{7.89}$$

where $n_1 = 10^2$ and $n_2 = 10^{12}$ so that 10 orders of magnitude separate the maximum and minimum values of the profile. This is a typical shape representative of a plasma discharge front. Results for SG, LLW-$\beta = 1.0$ and ISG-$\varepsilon = 0.1$ are shown in Fig. 7.16 after 200 iterations at CFL $= 0.4$ in a 101×2 resolution mesh illustrated in Fig. 7.7. Both schemes

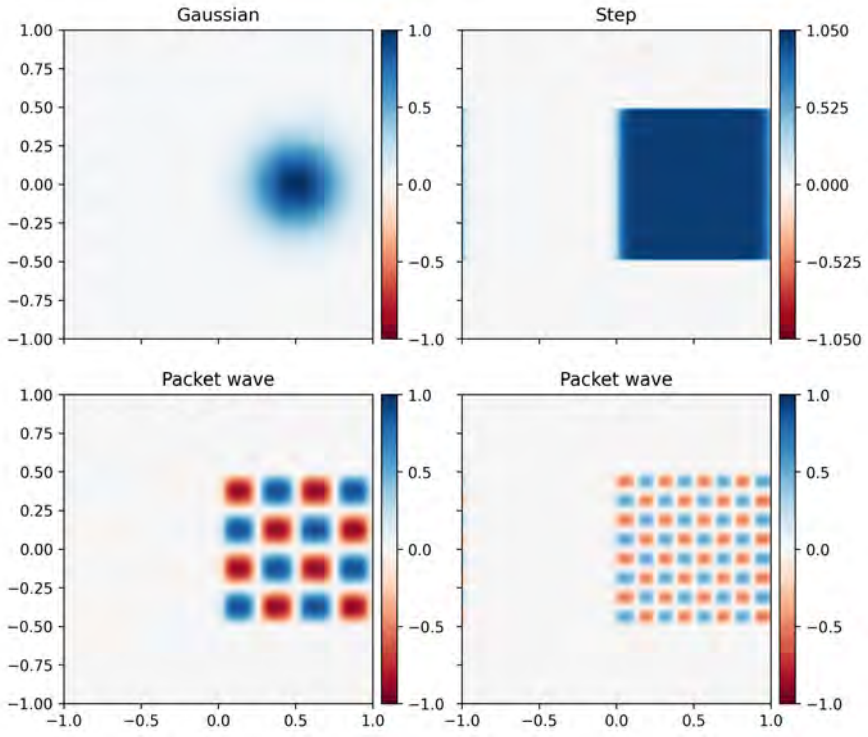
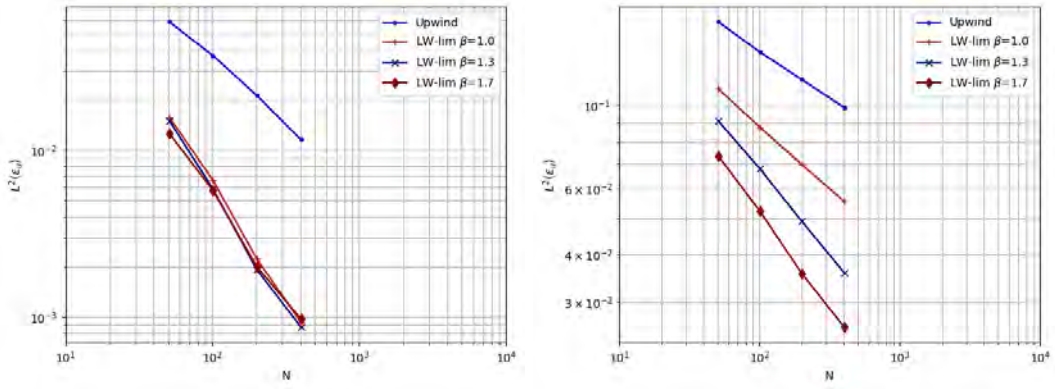


Figure 7.14: Two-dimensional advection test case results in 2D domain $[-1, 1]^2$ for scalar variable u for four profiles using LLW-$\beta = 1.3$ scheme at CFL $= 0.3$ in a 101×101 quadrangular mesh.



(a) Gaussian (b) Step

Figure 7.15: Convergence in 2D quadrangular mesh in a periodic square domain for the two-dimensional linear advection test case.

LLW-β and ISG-ε allow to retrieve better the discontinuity than the SG scheme which reduces to an upwind scheme for this advection dominated problem.

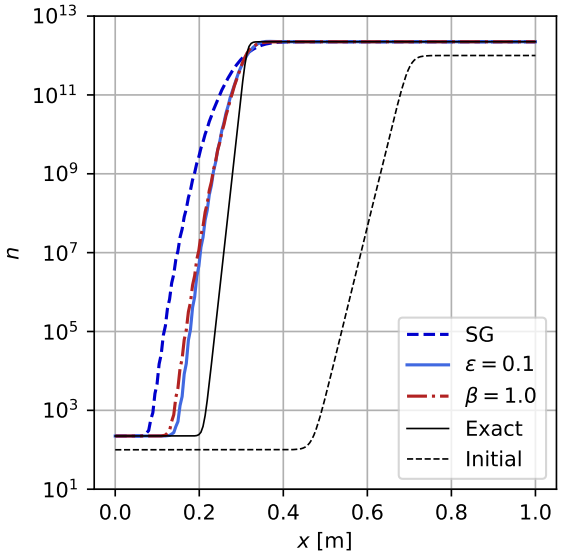


Figure 7.16: One-dimensional stiff advection test case using SG, ISG-$\varepsilon = 0.1$ and LLW-$\beta = 1.0$ schemes after 200 iterations at CFL $= 0.4$ in a 101×2 resolution mesh.

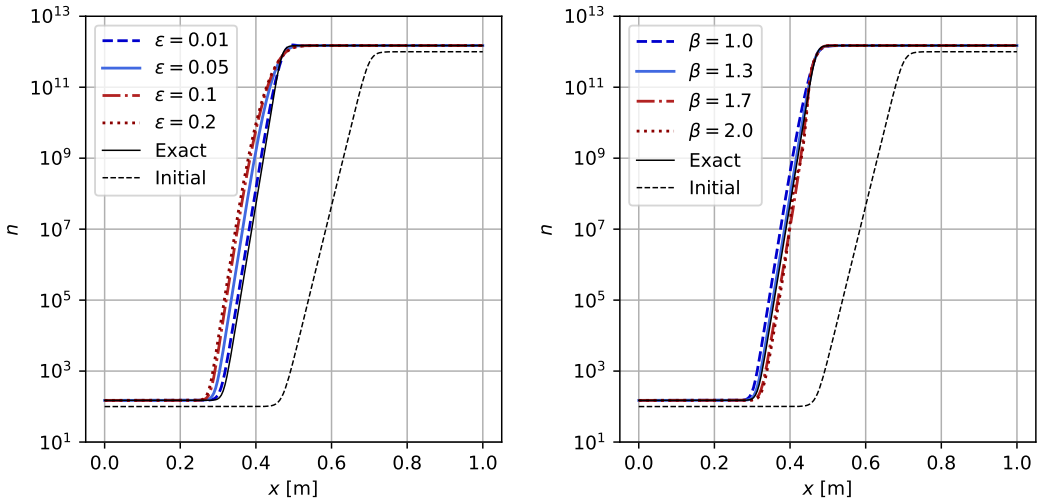
Both LLW-β and ISG-ε schemes have parameters that allow to gain higher order and better capture discontinuities: $1 \leq \beta \leq 2$ for limited Lax-Wendroff and $0 \leq \varepsilon \leq 1$ for improved SG. For Lax-Wendroff the higher the value of β the better the scheme can capture discontinuities whereas for improved SG the lower the value of ε the better. The influence of those parameters on the hyperbolic tangent profile is shown in Fig. 7.17a and 7.17b, respectively. Results are shown at CFL $= 0.4$ after 200 iterations on a finer grid than previously with 201 nodes in the x direction. Both schemes tend to the exact solution as their parameters are closer to their limits ($\beta = 2$ and $\varepsilon = 0$): we can however observe that the improved SG scheme tends monotonically from left to right whereas the limited Lax-Wendroff is a bit left then a bit right of the profile. These values are theoretical and in practice oscillations appear as the ones showed for the linear advection case when the parameters β and ε are too close to their limit with an added over-compressive effect for LLW-β.

7.3.4 Summary of the schemes

A summary of the schemes is given in Tab. 7.1 with the recommanded range of parameters to try at first for each case based on the results of the various validation cases.

Scheme	Parameter range	Recommanded range
LLW-β	[1, 2]	[1, 1.6]
ISG-ε	[0, 1]	[0.01, 0.1]

Table 7.1: Summary of the TVD schemes with recommanded values of parameters.



(a) Improved SG

(b) Limited Lax-Wendroff

Figure 7.17: Influence of interface capturing parameters ε and β for improved SG and limited LW schemes for the one-dimensional stiff advection test case.

Chapter 8

Euler and Navier-Stokes equations

Contents

In this chapter, the development of robust numerical schemes for the Euler equations is detailed. The choice of Riemann solvers Toro [2009] to properly describe interfaces and shocks has been made. The work in this chapter builds upon previous work at CERFACS [Joncquieres, 2019, Carmona, 2021] and tries to improve and address their limitations, especially concerning the MUSCL procedure.

In a first part, reminders on the properties of the Euler equations are given adapted from [Toro, 2009, Chap. 3]. The HLLC solver [Toro, 2009, Chap. 10] and HLLC MUSCL-Hancock solver [Toro, 2009, Chap. 14] are then presented along with their implementations in AVIP. The time integration of the HLLC-MUSCL solver is also discussed. We finally test the implemented schemes on a variety of test cases, measuring its practical order of accuracy whenever possible. The robustness of the solvers are shown in one and multi-dimensional shocks as well as their competitive accuracy on non-stiff canonical cases.

Note that the developments presented in this work have also been reported in the current version of AVBP.

8.1 General properties of the Euler equations

Mathematical properties of the Euler equations are recalled in this section. Conservative and primitive formulations are first laid out in one-dimension along with their eigenstructure and eigenvectors. We then present the exact resolution of the 1D Riemann problem which is the basis of all Riemann solvers.

8.1.1 One-dimensional Euler equations

8.1.1.1 Conservative formulation

Let us take the 1D Euler equations in conservative formulation

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0 \tag{8.1}$$

with

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix} \qquad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (\rho E + p)u \end{bmatrix} \tag{8.2}$$

where ρ is the density, u the velocity, p the pressure with the total energy $\rho E = \rho u^2/2 + p/(\gamma - 1)$ for ideal gases. We define the flux jacobian $\mathbf{A}(\mathbf{U})$ so that

$$\mathbf{A}(\mathbf{U}) = \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{\gamma-3}{2}u^2 & (3-\gamma)u & (\gamma-1) \\ -\gamma E u + (\gamma-1)u^3 & \gamma E - \frac{3}{2}(\gamma-1)u^2 & \gamma u \end{bmatrix} \tag{8.3}$$

The flux jacobian is diagonalizable and has three eigenvalues $\lambda_1 = u-a$, $\lambda_2 = u$, $\lambda_3 = u+a$ where a is the sound speed defined as $a = \sqrt{\gamma p/\rho}$. One right eigenvector matrix is

$$\mathbf{K} = \begin{bmatrix} 1 & 1 & 1 \\ u-a & u & u+a \\ H - ua & \frac{1}{2}u^2 & H + ua \end{bmatrix} \tag{8.4}$$

where $H = E + \frac{p}{\rho}$ is the total enthalpy.

8.1.1.2 Primitive formulation

Similar relations are obtained for the primitive variables $\mathbf{W} = [\rho\ u\ p]^T$:

$$\mathbf{W}_t + \mathbf{A}(\mathbf{W})\mathbf{W}_x = 0 \tag{8.5}$$

where

$$\mathbf{A} = \begin{bmatrix} u & \rho & 0 \\ 0 & u & 1/\rho \\ 0 & \rho a^2 & u \end{bmatrix} \tag{8.6}$$

The three eigenvalues are identical to the conservative formulation and one right eigenvector matrix is

$$\mathbf{K} = \begin{bmatrix} 1 & 1 & 1 \\ -a/\rho & 0 & a/\rho \\ a^2 & 0 & a^2 \end{bmatrix} \tag{8.7}$$

One left eigenvector matrix (which is not in this case the inverse of the right eigenvector matrix above) is

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & -\frac{1}{\rho a} \\ 1 & 0 & -1/a^2 \\ 0 & 1 & \frac{1}{\rho a} \end{bmatrix} \tag{8.8}$$

so that along the λ_i-characteristic $\mathbf{L}^{(i)} \cdot \mathrm{d}\mathbf{W} = 0$. Hence

$$\begin{cases} \mathrm{d}p - \rho a \mathrm{d}u = 0 & \text{along} \quad \mathrm{d}x/\mathrm{d}t = u - a \\ \mathrm{d}p - a^2 \mathrm{d}\rho = 0 & \text{along} \quad \mathrm{d}x/\mathrm{d}t = u \\ \mathrm{d}p + \rho a \mathrm{d}u = 0 & \text{along} \quad \mathrm{d}x/\mathrm{d}t = u + a. \end{cases} \tag{8.9}$$

Another set of equation for primitive variables $\mathbf{W} = [\rho \; u \; s]^T$ can be derived where

$$\mathbf{A} = \begin{bmatrix} u & \rho & 0 \\ a^2/\rho & u & \frac{1}{\rho}\frac{\partial p}{\partial s} \\ 0 & 0 & u \end{bmatrix} \tag{8.10}$$

and for which one right eigenvector matrix is

$$\mathbf{K} = \begin{bmatrix} 1 & -\frac{\partial p}{\partial s} & 1 \\ -a/\rho & 0 & a/\rho \\ 0 & a^2 & 0 \end{bmatrix} \tag{8.11}$$

8.1.2 Characteristic fields

Definition 8.1 (Linearly degenerate fields)**.** *A field associated to an eigenvalue λ_i is said to be linearly degenerate if*

$$\forall\, \mathbf{U} \in \mathbb{R}^m \;\; \nabla\lambda_i(\mathbf{U}) \cdot \mathbf{K}^{(i)} = 0 \tag{8.12}$$

Definition 8.2 (Genuinely non-linear fields)**.** *A field associated to an eigenvalue λ_i is said to be genuinely non-linear if*

$$\forall\, \mathbf{U} \in \mathbb{R}^m \;\; \nabla\lambda_i(\mathbf{U}) \cdot \mathbf{K}^{(i)} \neq 0 \tag{8.13}$$

8.1.2.1 Rankine-Hugoniot conditions

Given a hyperbolic system of conservation laws

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0 \tag{8.14}$$

and a discontinuous wave solution of speed S_i associated with the λ_i characteristic field, the Rankine-Hugoniot (RH) conditions state that

$$\Delta \mathbf{F} = S_i \Delta \mathbf{U} \tag{8.15}$$

8.1.2.2 Generalized Riemann Invariants

We consider a general quasi-linear hyperbolic system

$$\mathbf{W}_t + \mathbf{A}(\mathbf{W})\mathbf{W}_x = 0 \tag{8.16}$$

with $\mathbf{W} = [w_1, w_2, \ldots, w_m]^T$ and its i-characteristic with eigenvalue λ_i and corresponding right eigenvector $\mathbf{K}^{(i)}$.

Generalized Riemann Invariants (GRI) are relations that hold true, for certain waves, across the wave structure and lead to the following $(m-1)$ ordinary differential equations

$$\frac{\mathrm{d}w_1}{k_1^{(i)}} = \frac{\mathrm{d}w_2}{k_2^{(i)}} = \ldots = \frac{\mathrm{d}w_m}{k_m^{(i)}} \tag{8.17}$$

8.1.3 Elementary-wave solutions of the Riemann problem

We consider the 1D Riemann problem for the Euler equations

$$\begin{cases} \mathbf{U}_t + \mathbf{F}(\mathbf{U})x = 0 \\ \mathbf{U}(\mathbf{x}, 0) = \begin{cases} \mathbf{U}_L & \text{if} \quad x > 0 \\ \mathbf{U}_R & \text{if} \quad x < 0 \end{cases} \end{cases} \tag{8.18}$$

with $\mathbf{F}$ and $\mathbf{U}$ given by (8.2). This is the simplest non-trivial solution of the Euler equations across a discontinuity. The Riemann problem for the 1D Euler equations admits three kind of waves:

- The λ_1 and λ_3 are genuinely non-linear and are either shocks or rarefaction waves
- The λ_2 wave is always a contact discontinuity

8.1.3.1 Contact discontinuities

Across a contact disconstinuity, states are connected through a single jump discontinuity in a *linearly degenerate field* and the RH relations and the GRI are valid. Using the GRI across $\lambda_2 = u$ wave for the conservative formulation

$$\frac{\mathrm{d}(\rho E)}{\frac{1}{2}u^2} = \frac{\mathrm{d}(\rho u)}{u} = \mathrm{d}\rho \tag{8.19}$$

so that $\mathrm{d}u = 0$ and $\mathrm{d}p = 0$. Hence

$$u = \text{constant} \quad \text{and} \quad p = \text{constant} \tag{8.20}$$

across a contact discontinuity. Only the density and density-dependent physical values can change across this wave.

8.1.3.2 Rarefaction waves

Across a rarefaction wave, the two states are connected through a smooth transition in a *genuinely non-linear field* and the GRI relations apply. We use the primitive-entropy formulation and make use of the GRI across the wave:

$$\frac{\mathrm{d}\rho}{\rho} = \mp\frac{\mathrm{d}u}{a/\rho} = \frac{\mathrm{d}s}{0} \tag{8.21}$$

$$\Longrightarrow \mathrm{d}u \pm \frac{a}{\rho}\mathrm{d}\rho = 0 \quad \text{and} \quad s = \mathrm{cst} \tag{8.22}$$

$$\Longrightarrow u \pm \frac{2a}{\gamma - 1} = \mathrm{cst} \quad \text{and} \quad s = \mathrm{cst} \quad \text{across} \quad \lambda = u \mp a \tag{8.23}$$

8.1.3.3 Shocks

In a shock wave the two states are connected through a single jump discontinuity in a *genuinely non-linear field* and the Rankine-Hugoniot relations apply. In the case of 1D shocks, two configurations are possible: in the λ_1 wave or the λ_3 wave. The region downstream or upstream will be denoted the Star region. The Rankine Hugoniot relations tell us that

$$\mathbf{\Delta F} = S_K \mathbf{\Delta U}. \tag{8.24}$$

We change the reference frame for one that is moving at velocity S_K (where $K = L, R$ corresponding to respectively the λ_1 or λ_3 wave) so that $\mathbf{\Delta F} = 0$ and thus

$$\rho_* \hat{u}_* = \rho_K \hat{u}_K \tag{8.25}$$

$$\rho_* \hat{u}_*^2 + p_* = \rho_K \hat{u}_K^2 + p_K \tag{8.26}$$

$$(\rho_* \hat{E}_* + p_*)\hat{u}_* = (\rho_K \hat{E}_K + p_K)\hat{u}_K \tag{8.27}$$

Different relations can then be derived between the different physical values and the main ones are given below:

$$\frac{\rho_*}{\rho_K} = \frac{\frac{p_*}{p_K} + \frac{\gamma-1}{\gamma+1}}{\frac{\gamma-1}{\gamma+1}\frac{p_*}{p_K} + 1} \tag{8.28}$$

$$\frac{\rho_*}{\rho_K} = \frac{(\gamma+1)(M_K - M_S)^2}{(\gamma-1)(M_K - M_S)^2 + 2} \tag{8.29}$$

$$\frac{p_*}{p_K} = \frac{2\gamma(M_K - M_S)^2 - (\gamma - 1)}{(\gamma+1)} \tag{8.30}$$

$$M_K - M_S = \pm\sqrt{\frac{\gamma+1}{2\gamma}\frac{p_*}{p_K} + \frac{\gamma-1}{2\gamma}} \tag{8.31}$$

$$u_* = S_K\left(1 - \frac{\rho_K}{\rho_*}\right) + u_K\rho_K/\rho_* \tag{8.32}$$

where $M_K = u_K/a_K$ and $M_S = S_K/a_K$ are the Mach numbers of the K state and of the shock.

8.1.4 Exact solution of the 1D Riemann problem

The λ_2 wave is a contact wave so that pressure and velocity are continuous across that wave. Only the density can admit a discontinuity across this wave so that the sought variables are p_*, u_*, ρ_{*L} and ρ_{*R}. We use the primitive variables $\mathbf{W}$ to specify the 4 states of this problem which are depicted in Fig. 8.1.

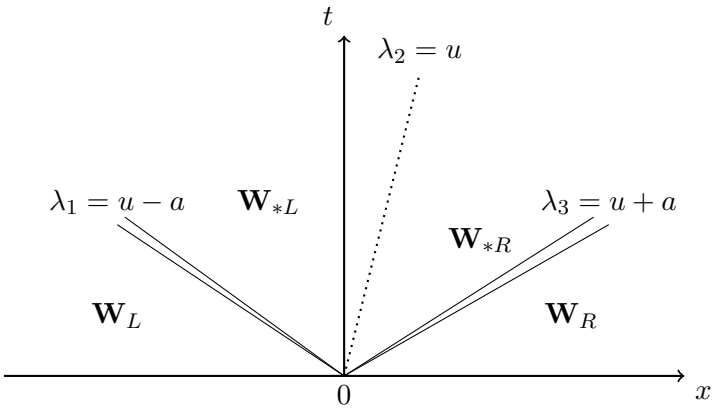


Figure 8.1: Structure of the Riemann problem for 1D Euler equations.

The central physical quantity of the Riemann problem is the pressure in the star region: it indicates whether the first and third interfaces are shocks or rarefactions waves. The solution for pressure p_* is given by the root of the following algebraic equation [Toro, 2009, Chap. 4]

$$f(p, \mathbf{W}_L, \mathbf{W}_R) \equiv f_L(p, \mathbf{W}_L) + f_R(p, \mathbf{W}_R) + \Delta u = 0 \tag{8.33}$$

where $\Delta u = u_R - u_L$ and the function f_K with $K = L, R$ is

$$f_K(p, \mathbf{W}_K) = \begin{cases} (p - p_K)\left[\frac{A_K}{p+B_K}\right]^{1/2} & \text{for} \quad p > p_K \quad \text{(shock)} \\ \frac{2a_K}{\gamma-1}\left[\left(\frac{p}{p_K}\right)^{\frac{\gamma-1}{2\gamma}} - 1\right] & \text{for} \quad p \leq p_K \quad \text{(rarefaction)} \end{cases} \tag{8.34}$$

where

$$A_K = \frac{2}{(\gamma+1)\rho_K} \quad B_K = \frac{\gamma-1}{\gamma+1} p_K \tag{8.35}$$

The velocity can be deduced from these functions as

$$u_* = \frac{1}{2}(u_L + u_R) + \frac{1}{2}(f_R(p_*) - f_L(p_*)) \tag{8.36}$$

Finally the densities are given by either the Rankine-Hugoniot relations for a shock $p_K > p_*$ or the isentropic equations for rarefaction waves since s is constant

$$\rho_{*K} = \begin{cases} \rho_K \frac{\frac{p_*}{p_K} + \frac{\gamma-1}{\gamma+1}}{\frac{\gamma-1}{\gamma+1}\frac{p_*}{p_K}+1} & \text{for} \quad p_* > p_K \\ \rho_K \left(\frac{p_*}{p_K}\right)^{1/\gamma} & \text{for} \quad p_* > p_K \end{cases} \tag{8.37}$$

8.1.5 Multi-dimensional Euler equations

The main properties of the multi-dimensional Euler equations are now given with emphasis on the rotational invariance which is really relevant in the Finite Volume Method. The integral form of the Euler equations, useful for FVM, is

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_V \mathbf{U}\,\mathrm{d}V + \int_{\partial V} \mathbf{F}\cdot\mathbf{n}\,\mathrm{d}S = 0 \tag{8.38}$$

The rotational invariance property simplifies the expression of $\mathbf{F}\cdot\mathbf{n}$ and is presented in two and three dimensions.

8.1.6 2D

The conservative variables vector $\mathbf{U}$ and fluxes $\mathbf{F}$ for the two-dimensional Euler equations are

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} \rho u & \rho v \\ \rho u^2 + p & \rho uv \\ \rho vu & \rho v^2 + p \\ \rho wu & \rho wv \\ (\rho E + p)u & (\rho E + p)v \end{bmatrix} \tag{8.39}$$

The flux jacobian of $\mathbf{F}$ is also diagonalizable with the same spectra as the one-dimensional case. However the eigenvalue u is degenerate so that $\lambda_1 = u - a$, $\lambda_2 = \lambda_3 = u$ and $\lambda_4 = u + a$. Across the λ_2 and λ_3 the pressure and normal velocity are constant so that density and tangential velocity can be discontinuous.

We define the components of the total flux $\mathbf{F} = (\mathbf{F}_1, \mathbf{F}_2)$ and the two angles of a 2D unitary vector $\mathbf{n} = (\cos(\theta), \sin(\theta))$ so that the rotational invariance property holds

$$\mathbf{F}\cdot\mathbf{n} = \cos(\theta)\mathbf{F}_1 + \sin(\theta)\mathbf{F}_2 = \mathbf{T}^{-1}\mathbf{F}_1(\mathbf{T}\mathbf{U}) \tag{8.40}$$

where $\mathbf{T}$ is the rotation matrix

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8.41}$$

8.1.7 3D

The conservative variables vector $\mathbf{U}$ and fluxes $\mathbf{F}$ for the three-dimensional Euler equations are

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} \rho u & \rho v & \rho w \\ \rho u^2 + p & \rho uv & \rho uw \\ \rho vu & \rho v^2 + p & \rho vw \\ \rho wu & \rho wv & \rho w^2 + p \\ (\rho E + p)u & (\rho E + p)v & (\rho E + p)w \end{bmatrix} \tag{8.42}$$

The flux jacobian of $\mathbf{F}$ also has the same spectrum as the 1D and 2D flux jacobians where the dimension of the eigensubspace u is now 3 so that across λ_2, λ_3 and λ_4 the pressure and normal velocity are constant while the density and tangential velocities are discontinuous.

The rotational invariance property also holds with a somewhat more complex rotation matrix involving two angles

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \cos\theta_y \cos\theta_z & \cos\theta_y \sin\theta_z & \sin\theta_y & 0 \\ 0 & -\sin\theta_z & \cos\theta_z & 0 & 0 \\ 0 & -\sin\theta_y \cos\theta_z & -\sin\theta_y \sin\theta_z & \cos\theta_y & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{8.43}$$

The inverse of the above matrix is given by

$$\mathbf{T}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \cos\theta_y \cos\theta_z & -\sin\theta_z & -\sin\theta_y \cos\theta_z & 0 \\ 0 & \cos\theta_y \sin\theta_z & \cos\theta_z & -\sin\theta_y \sin\theta_z & 0 \\ 0 & \sin\theta_y & 0 & \cos\theta_y & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{8.44}$$

8.2 Integrating the Euler equations

Various cell-vertex numerical schemes are already present in AVBP Lamarque [2007]. All schemes are however high-order, centered methods that do not possess the Total Variation Diminishing property. Artificial viscosity is applied to capture strong gradients and reduce wiggles.

To fill this gap, Riemann solvers have been developed based on Toro [2009] to capture shocks and strong discontinuities that are present in both plasma Euler equations and also in the mixture equations since streamer discharges can create strong shocks.

8.2.1 The HLLC MUSCL-Hancock solver

8.2.1.1 The HLLC solver

The HLL (Harten Lax and van Leer) and HLLC (where C stands for contact) solvers solve the 1D Riemann problem in an approximate manner. We place ourselves in a 1D mesh along the x direction where i is the mesh index. In a finite volume formulation the flux at the middle of the edge $\mathbf{F}_{i+1/2}$ is sought and the Riemann problem is

$$\begin{cases} \mathbf{U}_t + \mathbf{F}(\mathbf{U})x = 0 \\ \mathbf{U}(\mathbf{x}, 0) = \begin{cases} \mathbf{U}_i & \text{if} \quad x > 0 \\ \mathbf{U}_{i+1} & \text{if} \quad x < 0 \end{cases} \end{cases} \tag{8.45}$$

where $\mathbf{F}_{i+1/2} = \mathbf{F}(\mathbf{U}_{i+1/2}(0))$. The HLL solver solves the Riemann problem using only two waves. The solution $\mathbf{U}^{\text{hll}}$ between the two waves is constructed using integral relations and the reader is referred to [Toro, 2009, Chap. 10] for more details. The HLLC solver builds on top of the HLL solver by introducing a contact wave of speed S_* as shown in Fig. 8.2.

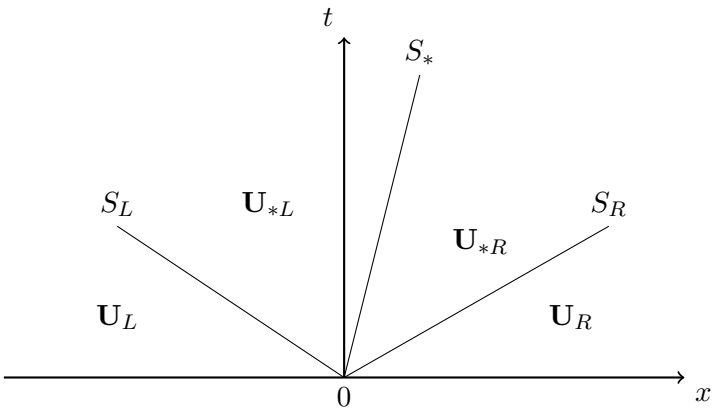


Figure 8.2: Structure of the solution for HLLC solver.

Two new states $\mathbf{U}_{*L}$ and $\mathbf{U}_{*R}$ are introduced which are related to the HLL solution by

$$\frac{S_* - S_L}{S_R - S_L}\mathbf{U}_{*L} + \frac{S_R - S_*}{S_R - S_L}\mathbf{U}_{*R} = \mathbf{U}^{\text{hll}} \tag{8.46}$$

From Fig. 8.2, the solution of the 1D Riemann problem is

$$\mathbf{U}(x,t) = \begin{cases} \mathbf{U}_L & x/t \leq S_L \\ \mathbf{U}_{*L} & S_L \leq x/t \leq S_* \\ \mathbf{U}_{*R} & S_* \leq x/t \leq S_R \\ \mathbf{U}_R & x/t \geq S_R \end{cases} \tag{8.47}$$

The fluxes inside the star regions are not directly evaluated at the states but are computed through Rankine-Hugoniot relations so that:

$$\mathbf{F}^{\text{hllc}}_{i+1/2} = \begin{cases} \mathbf{F}_L & 0 \leq S_L \\ \mathbf{F}_{*L} & S_L \leq 0 \leq S_* \\ \mathbf{F}_{*R} & S_* \leq 0 \leq S_R \\ \mathbf{F}_R & 0 \geq S_R \end{cases} \tag{8.48}$$

where $\mathbf{F}_{*K} = \mathbf{F}_K + S_K(\mathbf{U}_{*K} - \mathbf{U}_K)$. We finally need to retrieve star region variables. From the exact solution of the Riemann problem the pressure and normal velocity are continuous

across the contact wave while the density changes across all waves. From the Rankine-Hugoniot relations and enforcing equality of pressures the star region speed is found:

$$S_* = \frac{p_R - p_L + \rho_L u_L(S_L - u_L) - \rho_R u_R(S_R - u_R)}{\rho_L(S_L - u_L) - \rho_R(S_R - u_R)} \tag{8.49}$$

The star region conserved variables then reads

$$\mathbf{U}_{*K} = \rho_K \left(\frac{S_K - u_K}{S_K - S_*}\right) \begin{bmatrix} 1 \\ S_* \\ v_K \\ w_K \\ E_K + (S_* - u_K)\left[S_* + \frac{p_K}{\rho_K(S_K - u_K)}\right] \end{bmatrix} \tag{8.50}$$

Another formulation of the HLLC solver can be retrieved by first noticing that

$$\mathbf{F}(\mathbf{U}) = \mathbf{u}\mathbf{U} + \mathbf{p}\mathbf{D} \quad \text{where} \quad \mathbf{D} = [0\,1\,0\,0\,u]^T \tag{8.51}$$

From the Rankine-Hugoniot relation the following equality holds

$$\mathbf{U}_{*K} = \frac{S_K \mathbf{U}_K - \mathbf{F}_K + p_{*K}\mathbf{D}_*}{S_K - S_*} \tag{8.52}$$

To estimate the wave speeds we use the pressure based estimates from [Toro, 2009, Chap. 10.5.2] which are close to the exact resolution of the 1D Riemann problem recalled in Section. 8.1.4:

$$S_L = u_L - a_L q_L \quad \text{and} \quad S_R = u_R + a_R q_R \tag{8.53}$$

where

$$q_K = \begin{cases} 1 & \text{if} \quad p_* \le p_K \\ \left[1 + \frac{\gamma+1}{2\gamma}\left(\frac{p_*}{p_R} - 1\right)\right]^{1/2} & \text{if} \quad p_* > p_K \end{cases} \tag{8.54}$$

$$p_* = \frac{1}{2}(p_L + p_R) - \frac{1}{2}(u_R - u_L)\bar{\rho}\bar{a} \quad \text{wih} \quad \bar{\rho} = \frac{1}{2}(\rho_L + \rho_R) \quad \bar{a} = \frac{1}{2}(a_L + a_R) \tag{8.55}$$

8.2.1.2 The MUSCL-Hancock procedure

HLLC is a robust scheme however it is also highly diffusive and a high-order reconstruction is needed for practical use. Several reconstruction methods for Riemann solvers can be found in [Toro, 2009, Chap. 14], most notably the Weight Average Flux (WAF) reconstruction and the MUSCL-Hancock reconstruction. The former proves to be impractical for implementation in AVIP and so only the latter will be detailed. The MUSCL-Hancock reconstruction is decomposed in three steps: data reconstruction, evolution and finally Riemann problem. This procedure can be applied on other Riemann solvers but in AVIP the base Riemann solver is HLLC. The first step is to assume a piece-wise linear function in each cell:

$$\mathbf{U}_i(x) = \mathbf{U}_i^n + \frac{(x - x_i)}{\Delta x}\Delta_i \tag{8.56}$$

where Δ_i is a suitable chosen slope vector. At the boundaries of the cell the values are

$$\mathbf{U}_i^L = \mathbf{U}_i^n - \frac{1}{2}\Delta_i \quad \mathbf{U}_i^R = \mathbf{U}_i^n + \frac{1}{2}\Delta_i \tag{8.57}$$

We advance the solution by a timestep $\Delta t/2$ in the evolution phase:

$$\overline{\mathbf{U}_i^K} = \mathbf{U}_i^K + \frac{1}{2}\frac{\Delta t}{\Delta x}\Big[\mathbf{F}(\mathbf{U}_i^L) - \mathbf{F}(\mathbf{U}_i^R)\Big] \tag{8.58}$$

Finally the Riemann problem is solved with $\mathbf{U}_L \equiv \overline{\mathbf{U}_i^R}$ and $\mathbf{U}_R \equiv \overline{\mathbf{U}_{i+1}^L}$ to yield the intercell flux. The slope vector is given by

$$\Delta_i = \frac{1}{2}(1+\omega)\Delta_{i-1/2} + \frac{1}{2}(1-\omega)\Delta_{i+1/2} \tag{8.59}$$

where

$$\Delta_{i-1/2} \equiv \mathbf{U}_i^n - \mathbf{U}_{i-1}^n \quad \Delta_{i+1/2} \equiv \mathbf{U}_{i+1}^n - \mathbf{U}_i^n \tag{8.60}$$

and ω is an interpolation parameter that governs the weight of left and right gradient in the total nodal gradient. The two best parameters for this are $\omega = -1/2$ leading to a third order accurate interpolation at the cell-face value while $\omega = -1/3$ leads to a third order accurate scheme [Hirsch, 2007, Chap. 8.4.1]. A limited version of the slope can be retrieved by setting

$$\overline{\Delta}_i = \begin{cases} \max(0, \min(\beta\Delta_{i-1/2}, \Delta_{i+1/2}), \min(\Delta_{i-1/2}, \beta\Delta_{i+1/2})) & \Delta_{i+1/2} > 0 \\ \min(0, \max(\beta\Delta_{i-1/2}, \Delta_{i+1/2}), \max(\Delta_{i-1/2}, \beta\Delta_{i+1/2})) & \Delta_{i+1/2} < 0 \end{cases} \tag{8.61}$$

where $\beta = 1$ corresponds to the `minmod` limiter while $\beta = 2$ reproduces the `superbee` limiter [Toro, 2009, Chap. 14.4.3].

8.2.2 Implementation in AVIP

The implementation of the HLLC and MUSCL-Hancock procedure in AVIP are now detailed where the generalization of the previous relations to multidimensional and multicomponent flows in an unstructured mesh is given. As with the scalar transport equations, we solve these equations in a Finite Volume Vertex-Centered fashion.

8.2.2.1 The HLLC solver

We start from Eq. (8.38) by setting $V = V_i$, *i.e.* we integrate on the nodal i volume of the mesh. Splitting of the fluxes across the relevant faces yields the following discretization

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{U}_i + \frac{1}{V_i}\sum_{\tau \in E(i)}\sum_{f \in \partial V_i \cap \tau}\int_f \mathbf{F}\cdot\mathbf{n}\,\mathrm{d}S = 0 \tag{8.62}$$

This discretization is exactly the same as the one detailed in Eq. (7.60) and Fig. 7.3 of the previous chapter for scalar transport variables. Two ways of computing the face integral are possible: either by splitting across each dimension and solving each Riemann problem or

by rotating the flux into the local basis of the edge [Toro, 2009, Chap. 16]. The latter has been implemented in AVIP as it is less computationally expensive. Hence the face integral is computed in three steps:

1. Projection of the solution in the local basis $\mathbf{TU}$
2. Solve the Riemann problem in the local basis $\mathbf{F}_1 = \mathbf{F}_1(\mathbf{TU})$
3. Project back into the original basis $\mathbf{T}^{-1}\mathbf{F}_1(\mathbf{TU})$

In three dimensions and multi-component flows (where each species k is described by its mass fractions Y_k) the conservative variables, fluxes are now given by:

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \\ \rho Y_1 \\ \vdots \\ \rho Y_N \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ \rho w u \\ (\rho E + p)u \\ \rho Y_1 u \\ \vdots \\ \rho Y_N u \end{bmatrix} \tag{8.63}$$

The resulting star region state for multicomponent flows is now as follows:

$$\mathbf{U}_{*K} = \rho_K \left(\frac{S_K - u_K}{S_K - S_*} \right) \begin{bmatrix} 1 \\ S_* \\ v_K \\ w_K \\ E_K + (S_* - u_K)\left[S_* + \frac{p_K}{\rho_K (S_K - u_K)} \right] \\ (\rho Y_1)_K \\ \vdots \\ (\rho Y_N)_K \end{bmatrix} \tag{8.64}$$

8.2.2.2 The MUSCL procedure implementation in AVIP

The equivalents of $\Delta_{i-1/2}$ and $\Delta_{i+1/2}$ Eq. (8.61) in unstructured meshes are needed for MUSCL implementation. They correspond respectively to Δ_{li} and Δ_{ij} for i and Δ_{ij} and Δ_{jk} for j which are depicted in Figs. 8.3 and 8.4:

$$\Delta_{li} = \mathbf{U}_i - \mathbf{U}_l \quad \Delta_{ij} = \mathbf{U}_j - \mathbf{U}_i \quad \Delta_{jk} = \mathbf{U}_k - \mathbf{U}_j \tag{8.65}$$

Inside the domain of computation the gradients computed in AVIP are centered gradients so that the following reconstruction can be made:

$$\mathbf{U}_j - \mathbf{U}_l = (\nabla \mathbf{U})_i \cdot (2\mathbf{ij}) \tag{8.66}$$

In the case of regular meshes such as Figs. 8.3 and 8.4 the reconstructed nodes actually exist (l and k) but in the case of irregular meshes in Fig. 8.5 *virtual* upwind nodes are retrieved

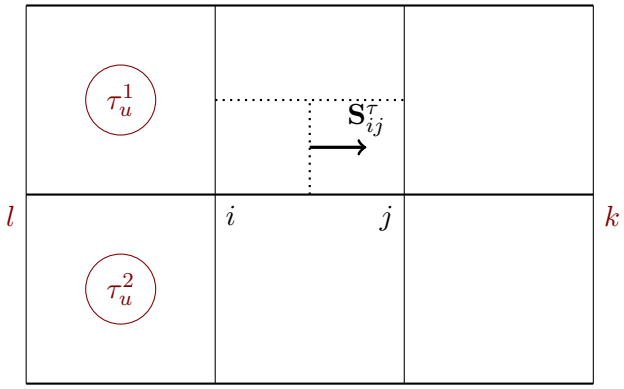


Figure 8.3: Reconstruction of neighboring nodes inside AVIP in quadrangular cells.

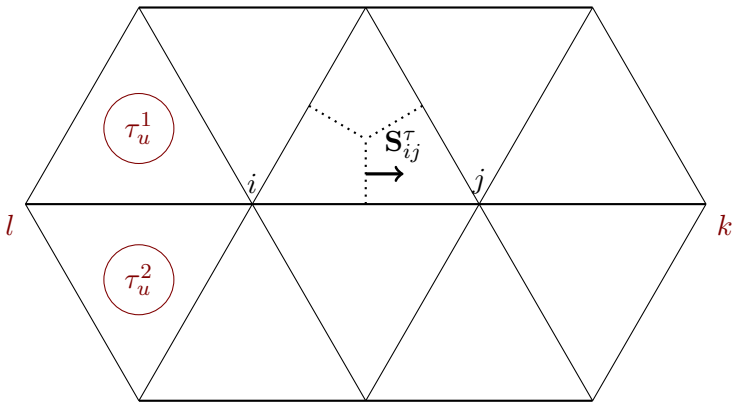


Figure 8.4: Reconstruction of neighboring nodes inside AVIP in triangular cells.

(l and k in dashed circles) which are consistent with the structured cartesian formulation of limiters: l and k are at a distance ij of i and j, respectively.

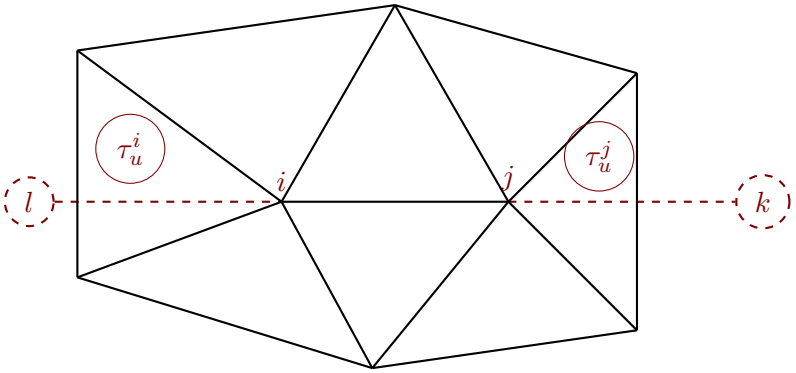


Figure 8.5: Reconstruction of neighboring nodes using upwind cells.

This reconstruction of upwind nodes is different from the ones adopted in Alauzet and Loseille [2010] and Joncquieres [2019] where upwind cells relative to the edge and node considered are sought which are shown in Fig. 8.5. For the i node in edge ij, this cell is defined as the one whose center is the most aligned with the edge ij. In the case of irregular meshes whose edges are not aligned this cell is well-defined (τ_u^i and τ_u^j in Fig. 8.5) however in the case of aligned edges such as those shown in Figs. 8.3 and 8.4 two equivalent upwind cells can be defined τ_u^1 and τ_u^2 for the i node. For rather regular elements a mean of the two upwind cells would be better than just picking one as this could lead otherwise to odd-even decoupling. This bad behavior in the case of regular elements vanishes for the virtual node reconstruction considered in Eq. (8.66). Moreover finding the associated upwind cells for each edge is computationally expensive and communications are needed across partitions whereas this implementation is local to the cell.

Effect of multicomponent mixture For multicomponent gas mixture equations the MUSCL-Hancock procedure needs to be slightly modified compared to what is presented in [Toro, 2009, Chap. 14]. In the original HLLC solver, primitive variables are sought from conservative variables which for the pressure reads:

$$p = (\gamma - 1)\rho e_s \tag{8.67}$$

where e_s is the internal or sensible energy. For multicomponent mixtures with temperature dependent c_v this relation does not hold anymore. In AVIP the pressure is computed from the temperature which is deduced from an iterative procedure on the sensible energy e_s so that

$$T = e_s^{-1} e_s(T) \implies p = \rho r T \tag{8.68}$$

Hence to circumvent the impossibility to use Eq. (8.67) the pressure is interpolated along with the conserved variables at the beginning of the MUSCL reconstruction:

$$\mathbf{U}_i^L = \mathbf{U}_i^n - \frac{1}{2}\bar{\Delta}_i \quad \mathbf{U}_i^R = \mathbf{U}_i^n + \frac{1}{2}\bar{\Delta}_i \tag{8.69}$$

$$p_i^L = p_i^n - \frac{1}{2}\bar{\Delta}p_i \quad p_i^R = p_i^n + \frac{1}{2}\bar{\Delta}p_i \tag{8.70}$$

This is an approximation compared to the original scheme: since the pressure is a non-linear function of the conserved variables a linear variation of the conservative variables does not result in a linear variation of the pressure.

8.2.2.3 Boundary conditions

Flux closure is performed exactly as for the scalar transport equations detailed in Sec. 7.2.5. Hence an equivalent of supersonic outlet is applied at the end of the numerical scheme yielding a result close to the cell-vertex schemes. Characteristic boundary conditions [Poinsot and Lelef, 1992] implemented in AVBP by Moureau et al. [2005] are then available where a detailed account can be found in [Lamarque, 2007, Chap. 4.12.2]. Values predicted by the numerical scheme is corrected by a wave decomposition of the Euler equations which is close to Riemann solvers.

A more consistent approach would be to include the boundary conditions directly in the numerical scheme by solving Riemann problems at the boundaries with the HLLC MUSCL-Hancock solver. Virtual states depending on the type of boundary are constructed in this case and Riemann problems are solved at each boundary interface. A comparison between NSCBC and Riemann solver boundary conditions could be an interesting future work.

8.2.2.4 Time integration of the HLLC MUSCL-Hancock solver

The Riemann solvers are not spatio-temporal schemes like the classical LW and TTGC schemes of AVBP. Hence a time integration on top of if must be chosen. Many strategies can be adopted depending on the stability criterions we want to achieve [Hirsch, 2007, Chap. 9]. Since we are dealing with unsteady flows the low-storage Runge-Kutta (RK) schemes for coupling with the HLLC MUSCL solver is chosen:

$$\begin{aligned}
&\mathbf{U}^{(1)} = \mathbf{U}^n \\
&\cdots \\
&\mathbf{U}^{(j)} = \mathbf{U}^n + \alpha_j \mathbf{R}^{(j-1)} \\
&\cdots \\
&\mathbf{U}^{(K)} = \mathbf{U}^n + \alpha_K \mathbf{R}^{(K-1)} \\
&\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t \sum_{k=1}^{K} \beta_k \mathbf{R}^{(k)}
\end{aligned}$$

where $\mathbf{R}$ is the residual, $\sum_k \beta_k = 1$ for consistency and each $\mathbf{U}^{(j)}$ is called a RK stage and K the number of stages. In AVIP the choice of $\beta_K = 1$, $\beta_j = 0 \quad j = 1 \ldots k-1$ is made where the RK1 to RK4 methods are defined by the coefficients given in Tab. 8.1.

Time integration method	α_2	α_3	α_4	β_K
RK1	0	0	0	1
RK2	1/2	0	0	1
RK3	1/3	1/2	0	1
RK4	1/4	1/3	1/2	1

Table 8.1: Runge-Kutta coefficients.

In [Toro, 2009, Chap. 14], a simple RK1 time integration scheme is always chosen in structured grids and seems to be stable for all the shock cases studied. The more stable but more costly RK2 and RK3 time integrations are also possible in AVIP and can be switched on whenever necessary.

8.2.3 Geometric source terms

As recalled in the first part, Navier-Stokes equations in axisymmetric conditions have the following form:

$$\frac{\partial r\mathbf{U}}{\partial t} + \boldsymbol{\nabla}_{2\mathrm{D}} \cdot (r\mathbf{F}^C_{rz} + r\mathbf{F}^D_{rz}) = \mathbf{S}^C_{rz} + \mathbf{S}^D_{rz} \tag{8.71}$$

where $\mathbf{F}^C_{rz}$ and $\mathbf{F}^D_{rz}$ are the classical convective and diffusive Navier-Stokes fluxes. Integration of these fluxes is done as for the scalar transport equations detailed earlier in Section. 7.2.4. However contrary to the scalar case, convective and diffusive geometric source terms are also present:

$$\mathbf{S}^C_{rz} = \begin{bmatrix} 0 \\ p \\ 0 \\ 0 \end{bmatrix} \qquad \mathbf{S}^D_{rz} = \begin{bmatrix} 0 \\ -2\eta\frac{u_r}{r} - \lambda\left(\frac{u_r}{r} + \frac{\partial u_r}{\partial r} + \frac{\partial u_z}{\partial z}\right) \\ 0 \\ 0 \end{bmatrix} \tag{8.72}$$

These geometric source terms are added after the transport residuals and radial velocity is corrected to zero after application of these geometric source terms to ensure $u_r = 0$ at the axis.

8.3 Validation cases

The HLLC MUSCL scheme is now tested on a variety of flows. The canonical 1D shock tube and 2D convective vortex are simulated where the order of convergence is assessed for both cases. Multi-dimensional shocks are then considered with explosions and two-dimensional Riemann problems taken from Deng et al. [2019]. Next Navier-Stokes equations are considered with the von Karman street test case to evaluate the capability of the scheme to not only propagate but also create vortices. Finally a one-dimensional flame is simulated using the HLLC MUSCL scheme and its accuracy compared to classical AVBP schemes is highlighted.

8.3.1 One-dimensional shock tubes

Figure 8.6: One dimensional shock configuration.

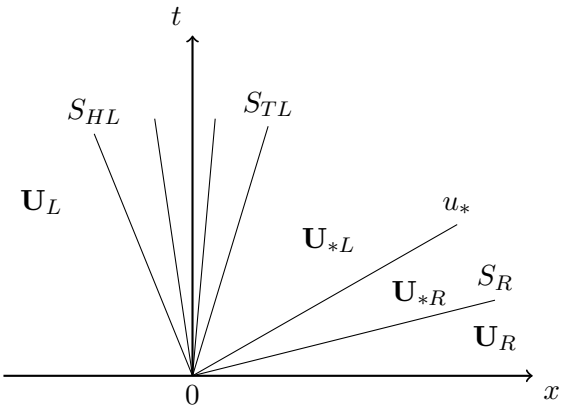


Figure 8.7: Structure of the Riemann problem for Case 1.

Five shock test cases presented in [Toro, 2009, Chap. 10] have been used to validate the TVD property of the schemes. Two states $\mathbf{U}_L$ and $\mathbf{U}_R$ are defined for each test case in a one-dimensional geometry as shown in Fig. 8.6. The initial conditions and solutions of the shocks in terms of star region variables are presented in Table 8.2. A more qualitative description of each case in terms of shocks and rarefaction waves (RW) is as follows:

- Case 1 is a modified version of the famous Sod problem [Toro, 2009, Chap. 10.8] where there is a left rarefaction wave, a right travelling contact wave and a right shock where a sketch of the wave-structure of the case is shown in Fig. 8.7.

  Left RW: S_{HL} = -0.433 m/s - S_{TL} = 0.300 m/s. Right Shock: S_R = 2.153 m/s

- Case 2 consists of two rarefaction waves going opposite directions and a stationnary contact wave. This test case is ideal for weakly ionized plasmas as it allows to evaluate the capability of the schemes to capture low-density flows.

  Left RW: S_{HL} = -2.748 m/s - S_{TL} = -0.348 m/s. Right RW: S_{TR} = 0.348 m/s - S_{HR} = 2.748 m/s

- Case 3 is the stiffest one where two shocks going opposite ways with a right travelling contact wave can be observed.

  Left RW: S_{HL} = -37.417 m/s - S_{TL} = -13.900 m/s. Right Shock: S_R = 23.518 m/s

- Case 4 consist of a slow right travelling contact wave. The purpose of this case is to assess the performance of the schemes to capture contacts, shear waves and vortices.

  Left RW: S_{HL} = -0.900 m/s - S_{TL} = -0.900 m/s. Right RW: S_{TR} = 1.283 m/s - S_{HR} = 1.283 m/s

1	ρ	u	p	a
W_L:	1.000	0.750	1.000	1.183
W_L^*:	0.580	1.361	0.466	1.061
W_R^*:	0.340	1.361	0.466	1.386
W_R:	0.125	0.000	0.100	1.058

2	ρ	u	p	a
W_L:	1.000	-2.000	0.400	0.748
W_L^*:	0.022	0.000	0.002	0.348
W_R^*:	0.022	0.000	0.002	0.348
W_R:	1.000	2.000	0.400	0.748

3	ρ	u	p	a
W_L:	1.000	0.000	1000.000	37.417
W_L^*:	0.575	19.597	460.894	33.497
W_R^*:	5.999	19.597	460.894	10.371
W_R:	1.000	0.000	0.010	0.118

4	ρ	u	p	a
W_L:	1.400	0.100	1.000	1.000
W_L^*:	1.400	0.100	1.000	1.000
W_R^*:	1.000	0.100	1.000	1.183
W_R:	1.000	0.100	1.000	1.183

Table 8.2: Summary of the four one-dimensional shock tube test cases.

Three schemes are tested on those cases for quadrangular and triangular meshes in 1D: HLLC, HLLC MUSCL-Hancock RK1 and LW. Note that changing the time integration to RK2 or RK3 for the 1D shock tube does not change the results. The triangular mesh consists of quadrangular elements cut in half along the diagonal. The three schemes are used without artificial viscosity and a summary of the passing or failure of each scheme is presented in Tab. 8.3.

Results for the four test cases are shown in Figs. 8.8-8.11 for a 101 nodes-resolution in the x direction. The LW produces strong oscillations in Case 1 and 4 as shown in Figs. 8.8 and 8.11. It is moreover not able to capture the vacuum and the two strong shocks of cases 2 and 3 respectively. HLLC and HLLC schemes yield stable simulations for all cases. The MUSCL reconstruction proves to be effective as a higher accuracy is clearly observed in Figs. 8.8-8.11 and a value of $\beta = 1.0$ is used for all those cases.

The influence of the Sweby [1984] parameter β for the limiting procedure is shown in Fig. 8.12 for a resolution of 51 nodes in the x direction allowing to discriminate the schemes better. The overcompressive effect of the superbee limiter $\beta = 2.0$ is clearly seen as overshoots on all variables can be observed. The value of β should be set depending on each case starting from $\beta = 1.0$ and a low value across strong shocks should be set.

Order of the schemes The order of the schemes on the first shock tube (Case 1) is computed in order to get a more quantitative assessment on the previous observations. Results are shown in Fig. 8.13 at CFL = 0.7 for HLLC, HLLC-MUSCL with two different values of β and LW for the density ρ and the pressure p. The robustness of the Riemann solvers is clearly seen compared to the centered LW scheme classicaly used in AVBP. LW (red di-

	HLLC	MUSCL	LW
Case 1	OK	OK	OK
Case 2	OK	OK	Crash
Case 3	OK	OK	Crash
Case 4	OK	OK	OK

Table 8.3: Results of the HLLC, HLLC MUSCL and LW schemes on the four shock tube cases.

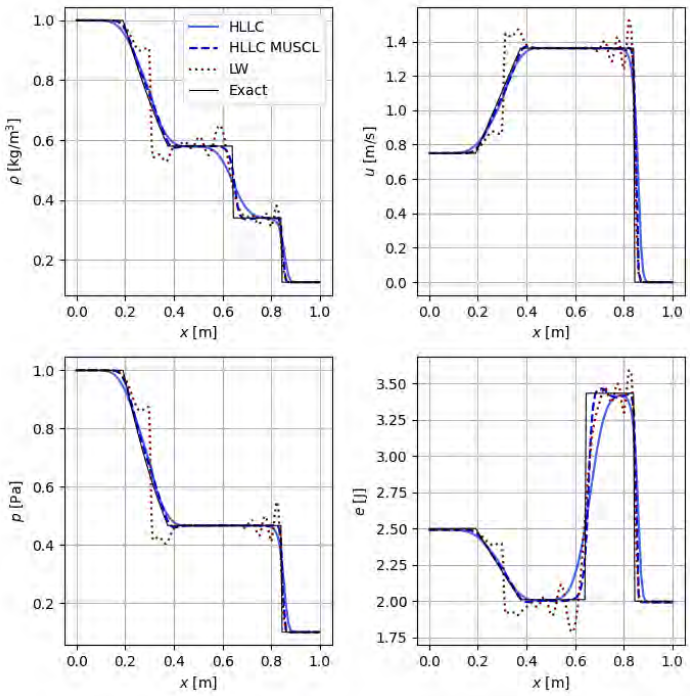


Figure 8.8: Shock tube case 1 with the HLLC, HLLC MUSCL RK1 and LW schemes for a quadrangular mesh.

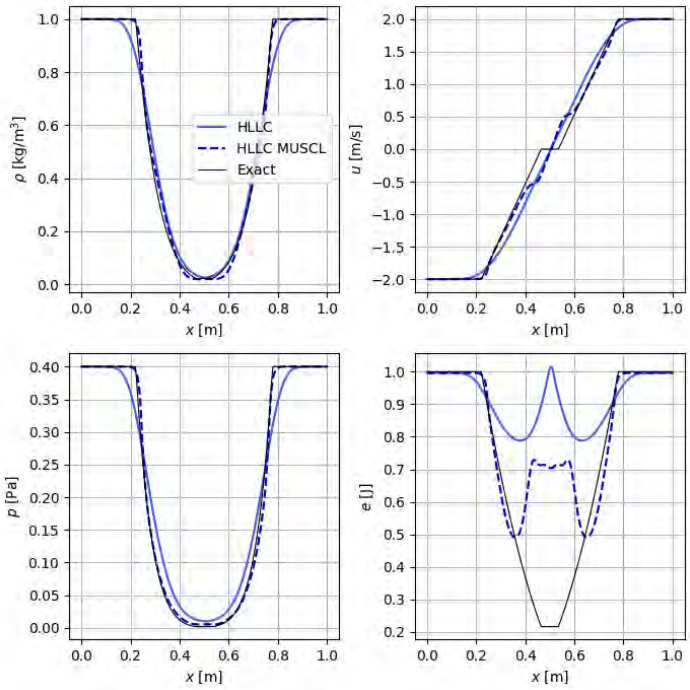


Figure 8.9: Shock tube case 2 with the HLLC, HLLC MUSCL RK1 and LW schemes for a quadrangular mesh.

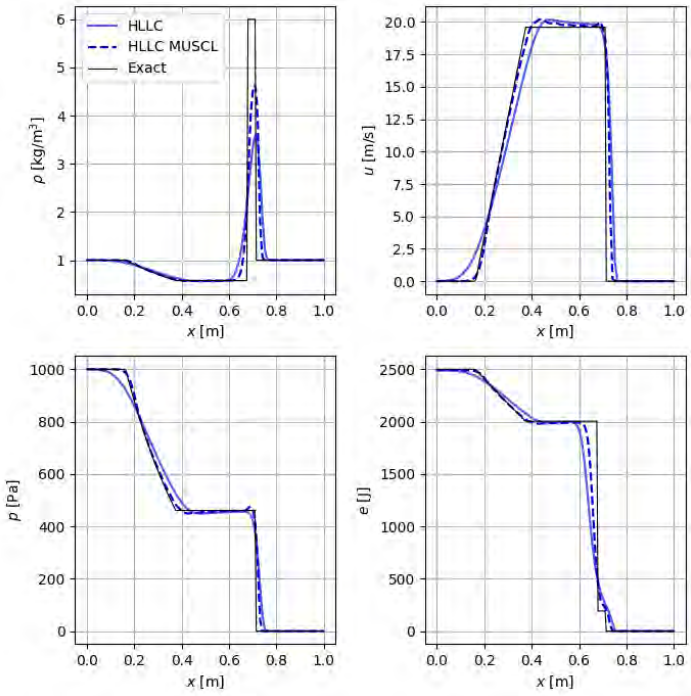


Figure 8.10: Shock tube case 3 with the HLLC, HLLC MUSCL RK1 and LW schemes for a quadrangular mesh.

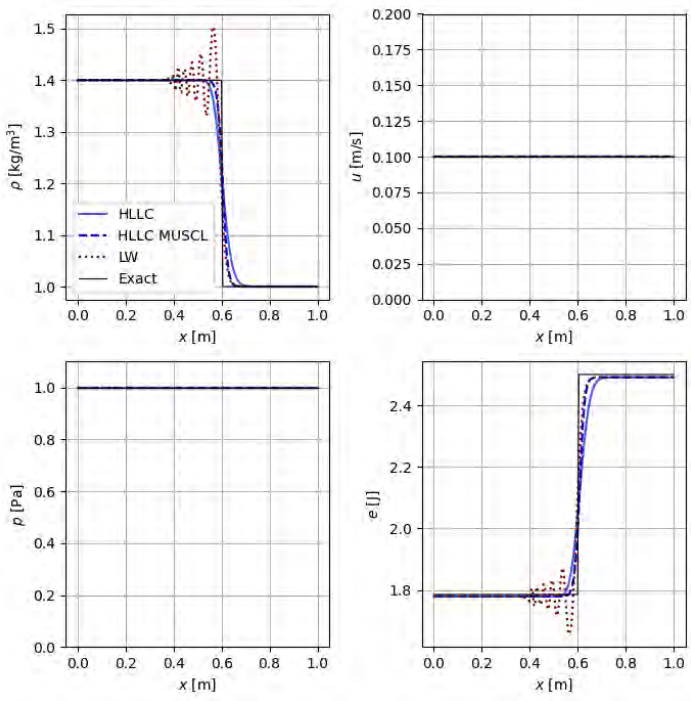


Figure 8.11: Case 4 with the three schemes for quadrangular mesh.

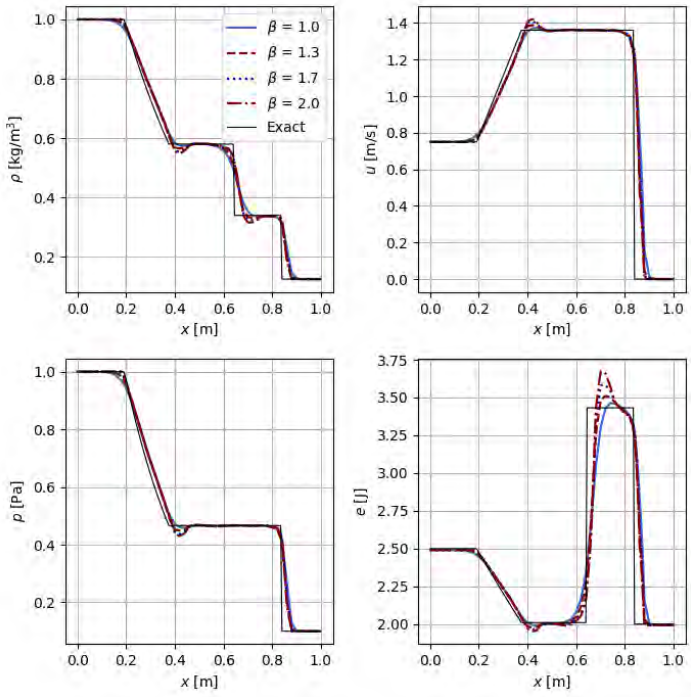


Figure 8.12: Case 1 with different β-Sweby limiter values for HLLC MUSCL RK1 scheme.

amond) does not converge to the exact solution as the number of nodes per direction N_h increases and is seen to stagnate from 201 nodes for both density and pressure. HLLC and HLLC-MUSCL on the other hand manage to converge to the exact solution. The order is similar for both schemes but the absolute error of HLLC-MUSCL is lower than HLLC. As expected, the higher the value of β the more accurate the scheme. Note that the density convergence (slope around 0.6) is slower compared to the pressure convergence (slope around 1.0) and discrepancies between Riemann solvers are more pronounced for the density.

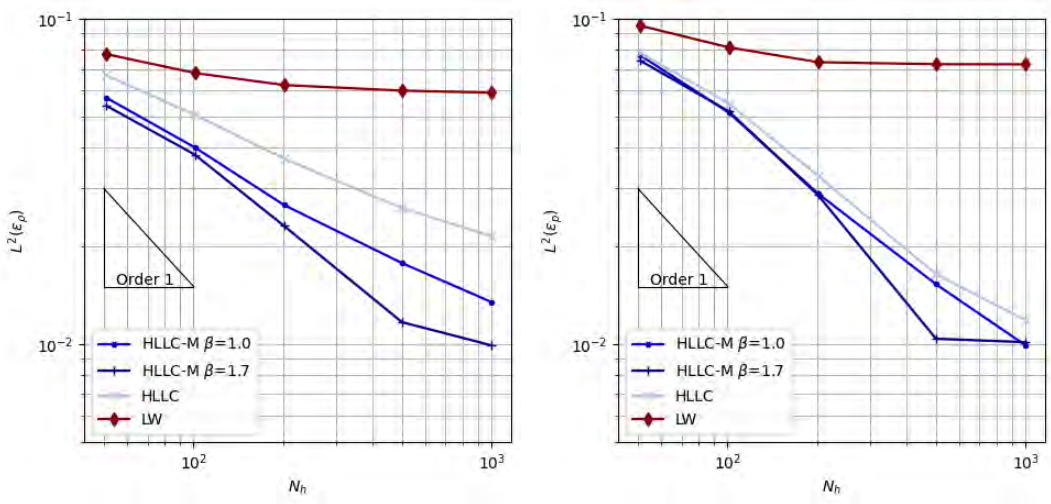
8.3.2 Convection of a 2D vortex

We consider a 2D-isentropic vortex that follows the Euler equations. Since it is an isentropic flow only the mass conservation and the momentum equations are necessary. The variables are decomposed into a mean flow and a perturbation flow convected at the mean flow velocity:

$$\mathbf{v}(\mathbf{x}, t) = \mathbf{v}_\infty + \mathbf{v}'(\mathbf{x} - \mathbf{x}_0 - \mathbf{v}_\infty t) \tag{8.73}$$

$$T = T_\infty + T'(\mathbf{x} - \mathbf{x}_0 - \mathbf{v}_\infty t) \tag{8.74}$$

Injecting the decompositions into the Euler equations the mass and momentum equations yield



(a) Density order (b) Pressure order

Figure 8.13: Order of the schemes on the shock tube Case 1.

$$\nabla \cdot (\rho \mathbf{v}') = 0 \tag{8.75}$$

$$\nabla \mathbf{v}' \cdot \mathbf{v}' + \frac{\gamma R}{\gamma - 1} \nabla T' = 0 \tag{8.76}$$

Setting $\mathbf{r} = \mathbf{x} - \mathbf{x}_0 - \mathbf{v}_\infty t$ a gaussian profile is chosen for the speed and the temperature is also retrieved so that

$$v'_r = 0 \quad v'_\theta = f(r) = \frac{\Gamma r}{R_c^2} \exp\left(-\frac{r^2}{2R_c^2}\right) \tag{8.77}$$

$$T'(r) = -\frac{\Gamma^2}{2C_p^m R_c^2} \exp\left(-\frac{r^2}{R_c^2}\right) \tag{8.78}$$

The maximum speed of the vortex is reached at $r = R_c$

$$v'_{\max} = \frac{\Gamma}{R_c \sqrt{e}} \tag{8.79}$$

Finally the pressure and density can be deduced from the temperature using isentropic relations:

$$\frac{\rho}{\rho_\infty} = \left(\frac{T}{T_\infty}\right)^{\frac{1}{\gamma-1}} \quad \frac{p}{p_\infty} = \left(\frac{T}{T_\infty}\right)^{\frac{\gamma}{\gamma-1}} \tag{8.80}$$

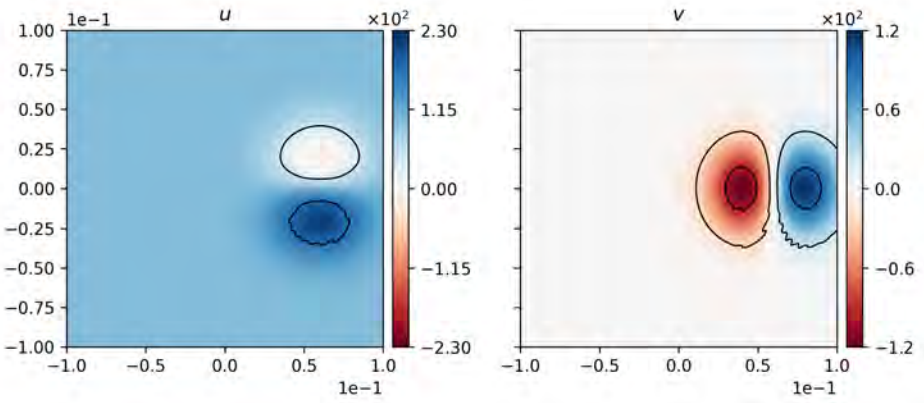
We choose a square domain centered at $(x_0, y_0) = (0, 0)$ with $x_{\max} = 0.1$ m with periodic boundary conditions. Two different vortices are studied whose parameters are presented in

	p_0	U_0	T_0	R_c	Γ	$v'_{\max}$
Weak vortex	1.01325 bar	100 m/s	300 K	2 cm	1 m^2/s	30.3 m/s
Strong vortex	1.01325 bar	100 m/s	300 K	2 cm	4 m^2/s	121.3 m/s

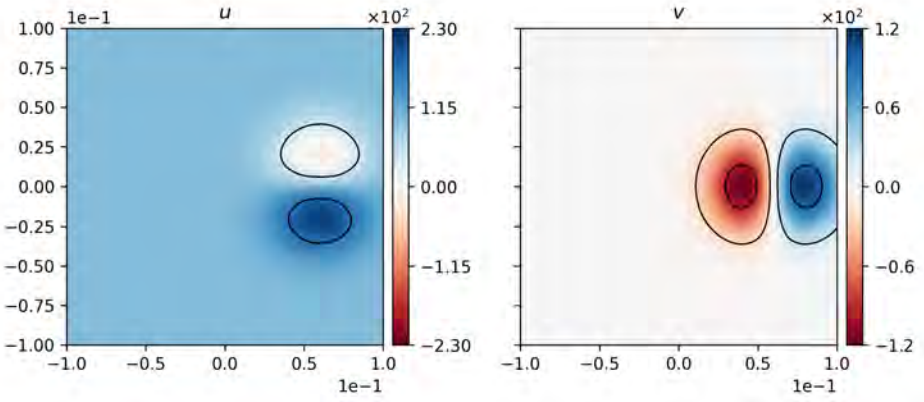
Table 8.4: Parameters of the two vortices.

Tab. 8.4. The weak and strong vortices are differentiated by the maximum speed of the vortices which is above the background speed U_0 for the strong vortex and below for the weak vortex.

The time integration of HLLC-MUSCL has proven to be an important issue for the strong vortex. Using an Euler time integration (RK1), the apparition of spurious oscillations coming from the limiting procedure can be seen in Fig. 8.14a for RK1 at CFL=0.7 for the speeds u and v. With the same CFL these oscillations dissipate using an RK2 time integration (Fig. 8.14b). No significant gain of stability has been found by switching to RK3 and the default time integration used from now on is RK2.



(a) HLLC-MUSCL RK1

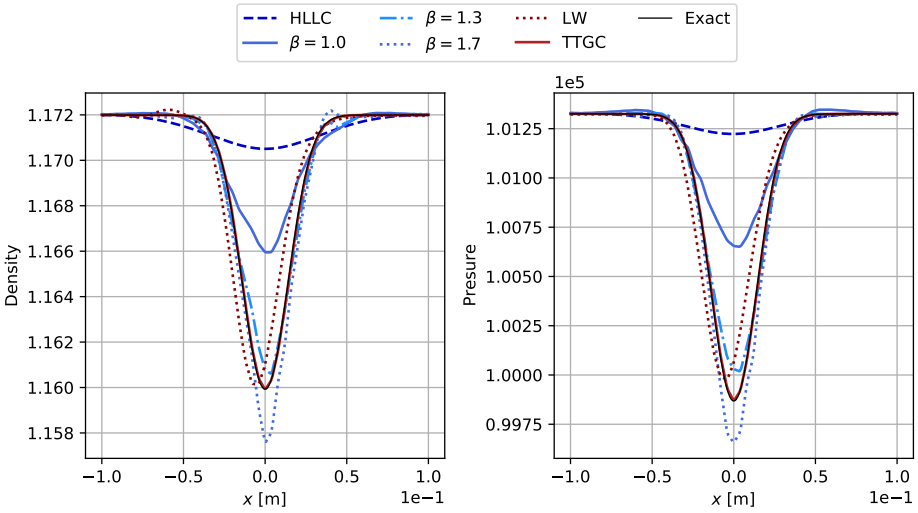


(b) HLLC-MUSCL RK2

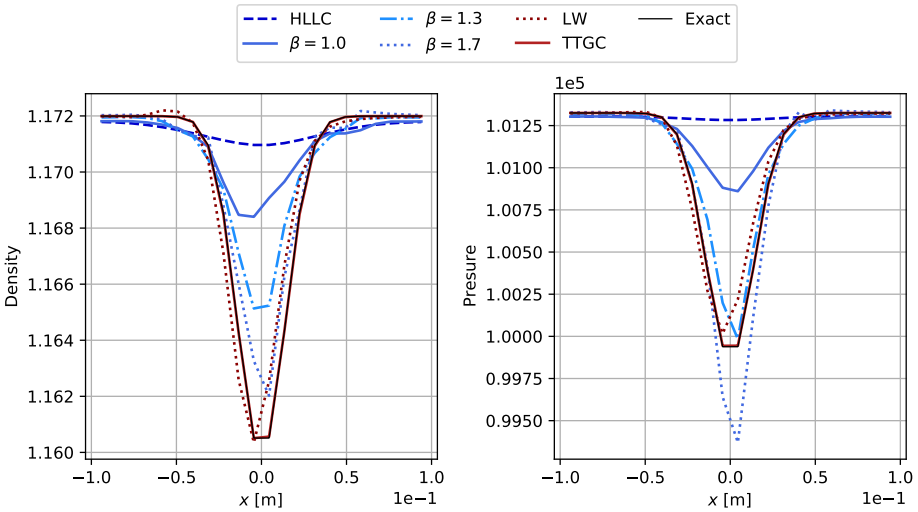
Figure 8.14: Speeds u and v [m/s] in the x and y axis respectively for strong convective vortex with HLLC MUSCL and different values of time integrations.

The different profiles of density and pressure after one full round are shown in Fig. 8.15

for CFL=0.7 in a coarse 51×51 mesh. In this case HLLC proves to be way too dissipative as the peak of density and pressure is almost invisible after one round. The improved accuracy of HLLC-MUSCL can be appreciated. However a too high value of β can lead to overshoot of pressure in coarse resolutions (blue-dotted line).



(a) Quadrangular elements



(b) Triangular elements

Figure 8.15: Comparison of the schemes after one full round of the domain with 51×51 resolution for the weak vortex.

Order of the schemes The order of the schemes on the weak convective vortex is computed in order to get a more quantitative assessment on the previous observations. Convergence is achieved by taking grids from 31×31 to 401×401. The error for pressure is shown in Fig. 8.16 at CFL = 0.7 for HLLC, HLLC-MUSCL RK2 with two different values of β and LW. Contrary to the slopes shown in Joncquieres [2019], Carmona [2021], the order

is taken after two full rounds of the domain and not after a very small displacement of the vortex. The small displacement is necessary to retrieve the first order of dissipative scheme as explained in Vanharen [2017]. It is not however really relevant for higher order schemes as these schemes are used for unsteady phenomena that need to be accurately transported over long period of times.

The overdissipative property of HLLC is clearly seen, the vortex is too damped so that the convergence is almost flat after two rounds. The HLLC-MUSCL has almost a second order accuracy after two rounds where the effect of limiters is less visible as the number of nodes is increased: there is not much to limit when the resolution is sufficient.

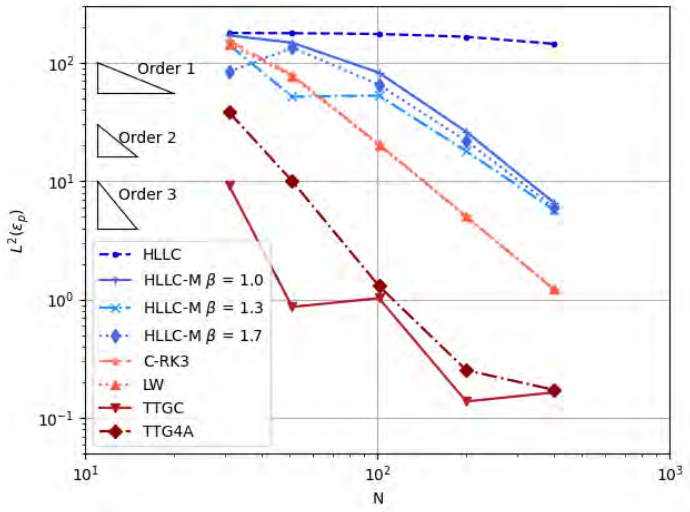


Figure 8.16: Order of the schemes on the weak vortex test case for pressure after two full rounds.

8.3.3 Multidimensional shocks

Multidimensional shocks are now studied in this part where two generalizations of the one-dimensional shock tube are studied: cylindrical and spherical explosions [Toro, 2009, Chap. 17] as well as two-dimensional Riemann problems Deng et al. [2019].

8.3.3.1 Explosions

The one-dimensional shock tube is extended as a circle in two dimensions and a sphere in three dimensions. An interior $\mathbf{U}_{\text{in}}$ and exterior $\mathbf{U}_{\text{out}}$ states are defined in this case as shown in Fig. 8.17. A smoothing function is applied so that there is no staircase profile along the interface. The geometry and initial conditions of [Toro, 2009, Chap. 17.1] are taken so that we simulate the two states

$$\mathbf{U}_{\text{in}} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{U}_{\text{out}} = \begin{bmatrix} 0.125 \\ 0 \\ 0 \\ 0.1 \end{bmatrix} \tag{8.81}$$

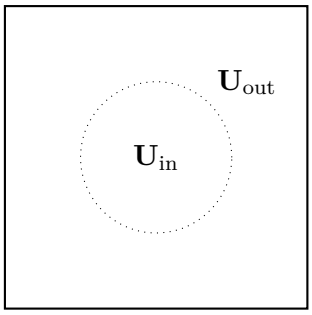
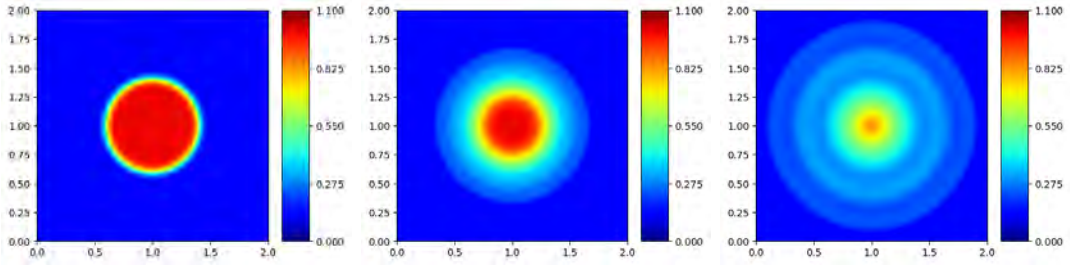


Figure 8.17: Two dimensional explosion problem.

in a $[0, 2] \times [0, 2]$ square with 101×101 resolution. The results are shown at different instants in Fig. 8.18 for a CFL of 0.5. An outgoing shock and inward rarefaction waves are present and well captured by the HLLC MUSCL RK2 scheme as shown in Fig. 8.18 for the density field.



(a) $t = 0$ s (b) $t = 0.15$ s (c) $t = 0.3$ s

Figure 8.18: Density [kg/m^3] for the two-dimensional explosion at different instants using HLLC MUSCL RK2.

This test case also allows to validate the pressure source term Eq. (8.72) in the axisymmetric formulation of the Euler equations as results should be consistent when comparing a 1D line in axisymmetric conditions with the 2D simulation. Comparisons of 2D simulations with HLLC-MUSCL RK2 at $\beta = 1.0$ and $\beta = 1.7$ with a 1D axisymmetric simulation with a much finer mesh of 1001 nodes along the y direction are shown in Fig. 8.19. Radial shocks have a tendency to create local minimas at the interfaces compared to the planar shocks presented in Section. 8.3.1. The 1D-axisymmetric simulation and 2D simulations are consistent and yield the same results and differences are due to coarser resolution for the 2D test case. As done in [Toro, 2009, Chap. 16], the consistency of the solutions allows to consider the fine-mesh 1D-axisymmetric solution to be the exact one validating the HLLC-MUSCL scheme for multi-dimensional cylindrical shocks.

8.3.3.2 2D Riemann problems

Another generalization of the 1D shock tube is to define four states in a square as shown in Fig. 8.20. Four different cases are considered here: the initial values for the first two cases are taken from Deng et al. [2019] and the third and fourth cases are variants of these. The geometry is a $[-0.5, -0.5] \times [0.5, 0.5]$ square with 602×602 resolution. All units are

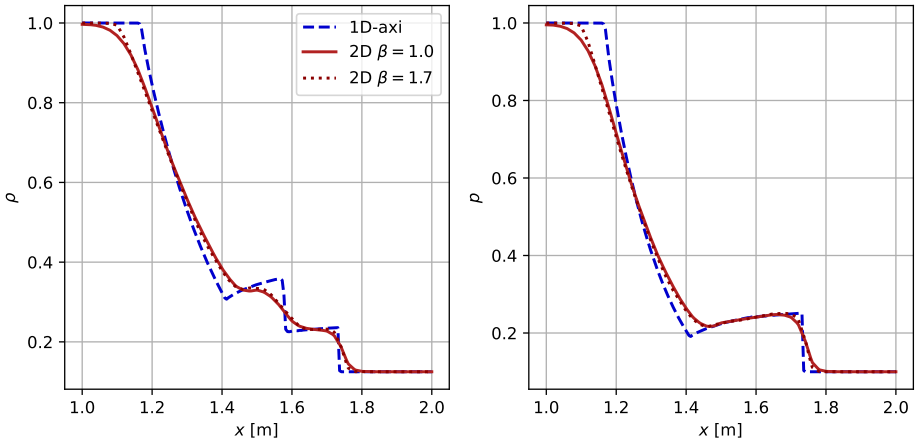


Figure 8.19: Comparison of 1D axisymmetric and 2D simulations for pressure and density profiles using HLLC MUSCL RK2 scheme.

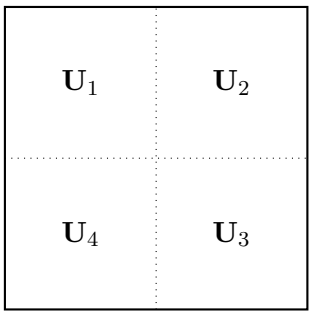
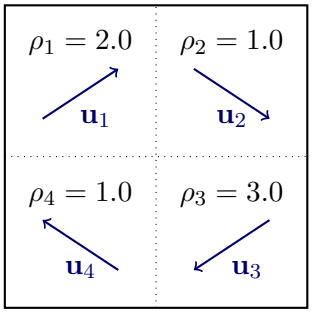
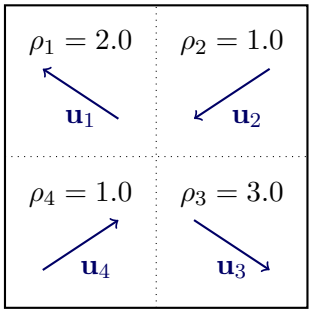


Figure 8.20: Two dimensional Riemann problem.

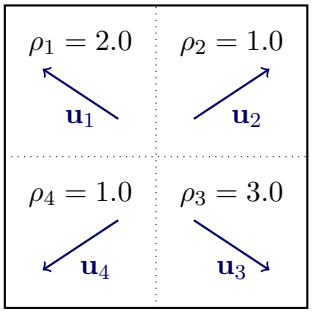
non dimensionalized here: pressure is always constant at 1.0 in all cases while the absolute values of the speeds on the x and y axis are 0.75 and 0.5 respectively. The different studied configurations are shown in Fig. 8.21 where the orientations of the speeds are given with arrows. The first case is a mixing interface where the four states turn clockwise. The second case consists of a shock in one diagonal and an expansion in the other diagonal. The third and fourth cases are respectively an all-in or all-out configuration in terms of speeds' directions.



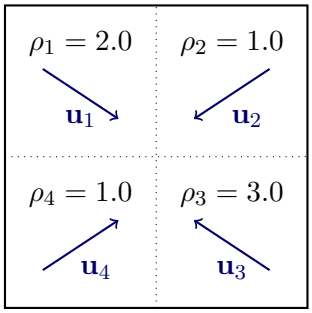
(a) Case 1 2D-Riemann problem.



(b) Case 2 2D-Riemann problem.



(c) Case 3 2D-Riemann problem.



(d) Case 4 2D-Riemann problem.

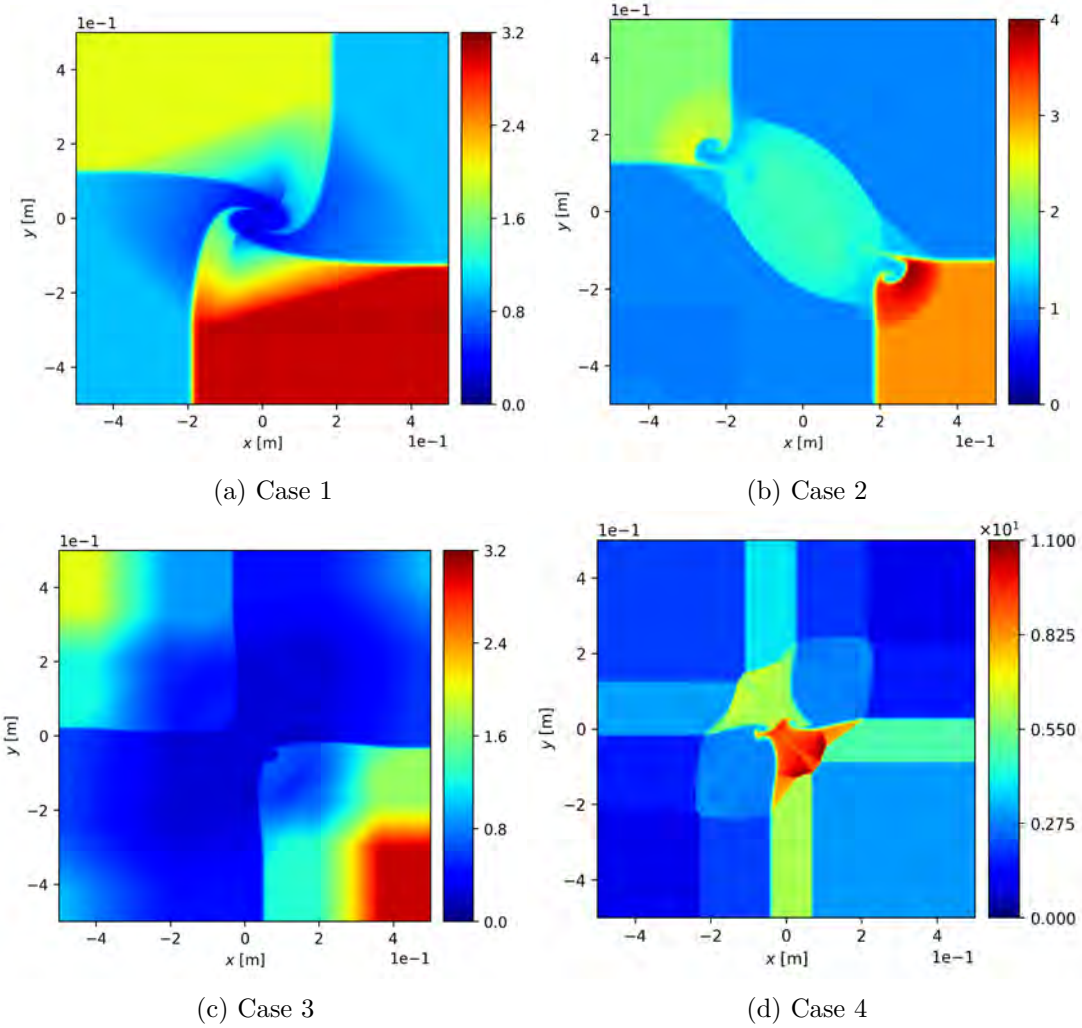
Figure 8.21: The four 2D Riemann problems considered.

Results of the four 2D Riemann cases using HLLC MUSCL RK2 are shown in Fig. 8.22. Interfaces and shocks are well captured by the scheme for all cases. The RK1 time integration produces wiggles similar to the ones observed for the convective vortex test case. This is especially true for the first case where vortices appear shown in Fig. 8.23.

8.3.4 Von Karman Street

The propagation of vortices and shocks in 2D has been validated in the previous section and we now turn to the vortices creation capability of the scheme. The Von Karman street test case is thus considered. The diameter of the cylinder is $d = 0.03$ m and the computational domain is $(L_x, L_y) = (60d, 40d)$ with the center of the cylinder at $(x_0, y_0) = (20d, 20d)$. The mesh has 60 000 nodes and 30 000 elements (only triangles) and a sketch of the computational domain with boundary conditions is shown in Fig. 8.24.

Results are shown at the onset of vortices for Re = 56 at $t = 0.02$ s in Fig. 8.25. LW manages to produces vortices in the wake of the cylinder but HLLC MUSCL-Hancock fails to do so as shown by Fig. 8.25 with a minmod limiter ($\beta = 1.0$). Increasing the Sweby parameter β allows to retrieve the vortices and results for $\beta = 1.3$ and $\beta = 1.7$ are shown



(a) Case 1

(b) Case 2

(c) Case 3

(d) Case 4

Figure 8.22: The four 2D Riemann solutions using HLLC-M RK2.

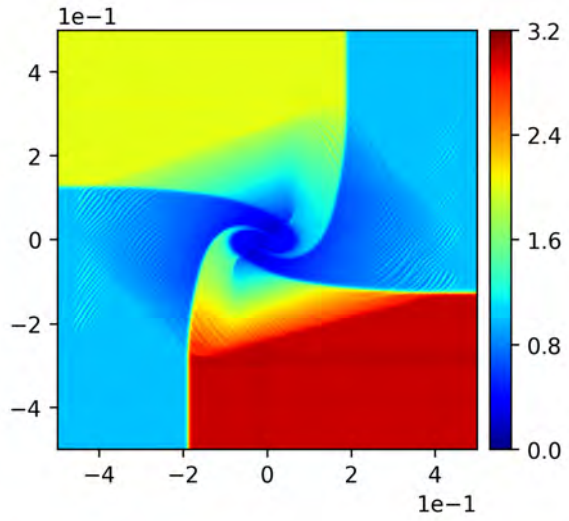


Figure 8.23: First 2D Riemann case using HLLC-M RK1.

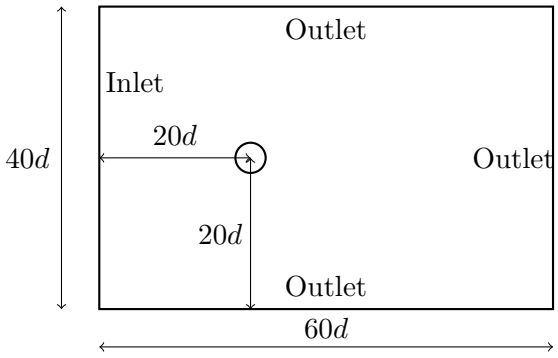
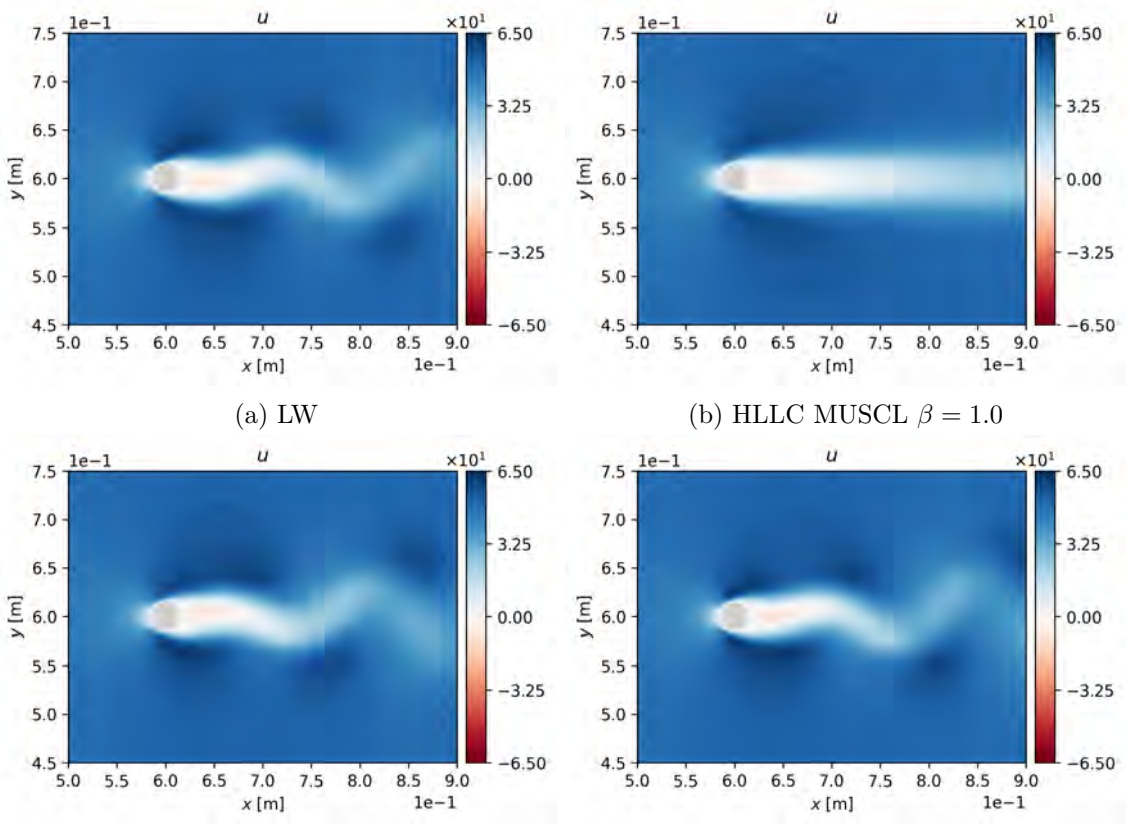


Figure 8.24: Sketch of the computational domain for the Von Karman street simulation.

in Figs. 8.25c and 8.25d where vortices at the same instant are created for those two values. HLLC is more diffusive than HLLC MUSCL and yields results similar to the HLLC MUSCL $\beta = 1.0$ scheme: no vortices appear.



(a) LW

(b) HLLC MUSCL $\beta = 1.0$

(c) HLLC MUSCL $\beta = 1.3$

(d) HLLC MUSCL $\beta = 1.7$

Figure 8.25: Axial speed of the Von Karmann street test case for Lax-Wendroff and HLLC MUSCL schemes at the same instant for Re = 56.

8.3.5 One-dimensional flames

The capability of the Riemann solvers to simulate flames is critical for plasma assisted combustion simulations. An atmospheric pressure one-dimensional flame at equivalence ratio $\phi = 0.8$ is simulated in AVIP using a converged solution from CANTERA [Goodwin et al., 2021]. A simple two-step chemical scheme with six species is used [Franzelli et al., 2012]:

$$CH_4 + 1.5\,O_2 \longrightarrow CO + 2\,H_2O \tag{8.82}$$

$$CO + 0.5\,O_2 \longrightarrow CO_2 \tag{8.83}$$

The 1D domain is 2 cm long and contains 500 quadrangular cells or 1000 triangular cells which have the same shape as Fig. 7.7. Boundary conditions and the geometry are summarized in Fig. 8.26: inlet and outlet NSCBC boundary conditions are applied at the

left and right parts of the domain while symmetry conditions are applied at the top and bottom. The inlet velocity of $0.2815\,\mathrm{m\,s^{-1}}$ is the laminar flame speed computed using the GRI-3.0 mechanism [Smith et al.].

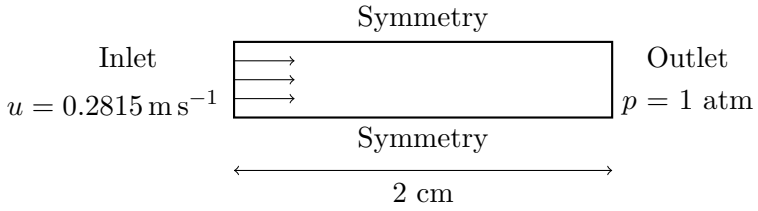


Figure 8.26: One dimensional flame configuration and boundary conditions.

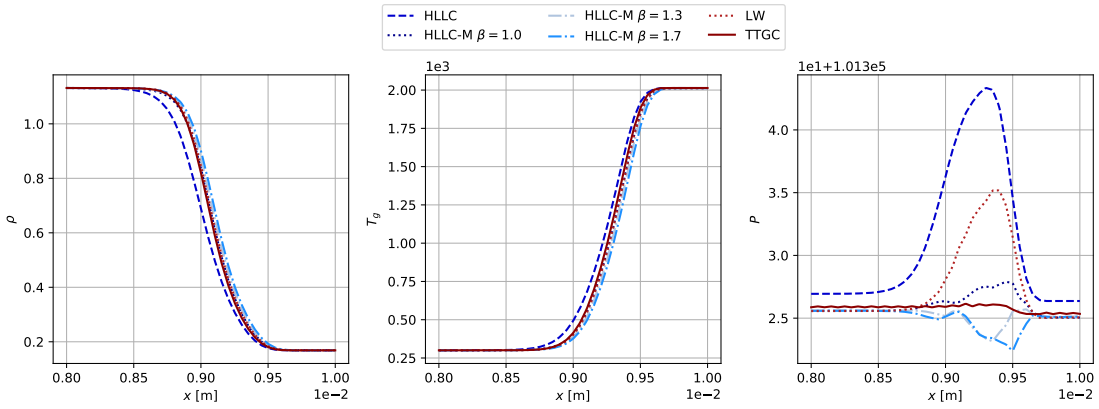
Simulations have been run for 10^{-1} s and results for HLLC and HLLC MUSCL are compared with classical schemes from AVBP that are embedded in AVIP: Lax-Wendroff and TTGC Lamarque [2007]. The density, pressure and temperature for the different schemes are shown in Fig. 8.27 for the quadrangular mesh at three different instants. A drift of the different schemes due to an error in the flame speed propagation is observed compared to the theoretical value of $28.15\,\mathrm{cm\,s^{-1}}$. TTGC converges to a value slightly above at $28.16\,\mathrm{cm\,s^{-1}}$ (0.03% error) after 0.08 s whereas LW underestimates the laminar flame speed at $27.93\,\mathrm{cm\,s^{-1}}$ (0.78% error). The diffusive scheme HLLC overestimates the flame speed by roughly 30% making it unrelevant for flame front propagation simulation. For HLLC MUSCL RK2, the Sweby parameter has a great influence: at $\beta = 1.0$, the flame front of HLLC MUSCL RK2 is between TTGC and LW which is satisfactory but as the parameter is increased the laminar flame speed is underestimated quite significantly so that a value of β below 1.3 should be prescribed for combustion applications.

Concerning the pressure, for this case the TTGC results are taken as reference since this scheme is the most accurate so that the right pressure jump is about 1 Pa. As expected, HLLC behaves poorly due to its diffusive nature and an overshoot of pressure of about 20 Pa is observed in the flame front while having a less steep flame front. The MUSCL procedure allows to retrieve a steeper flame front and a diminished pressure overshoot which is less than the one from LW for both quadrangular and triangular meshes. Increasing the value of the Sweby limiter β causes the pressure overshoot to become an undershoot but the amplitude remains similar (around 4 Pa).

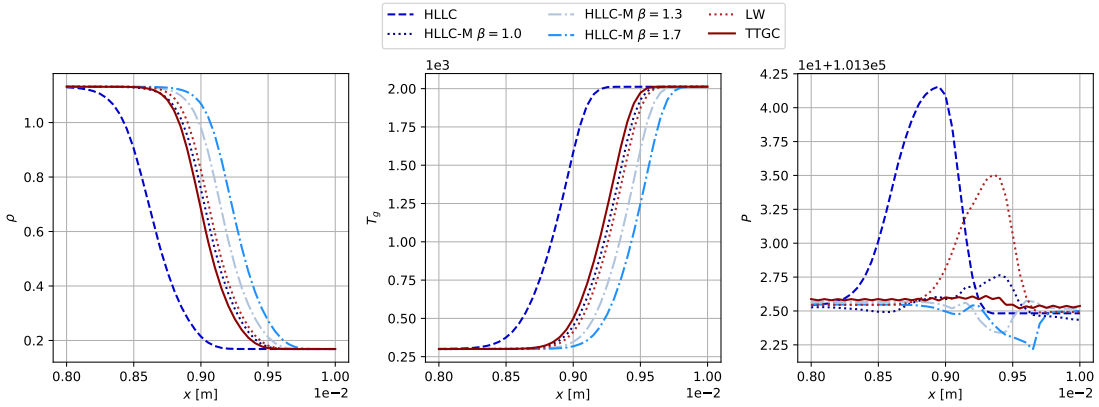
To conclude the Sweby parameter of the HLLC MUSCL RK2 scheme should be kept sufficiently low (below 1.3) to have proper flame propagation with reasonable flame speed and thickness.

8.3.6 Summary of the schemes and performance

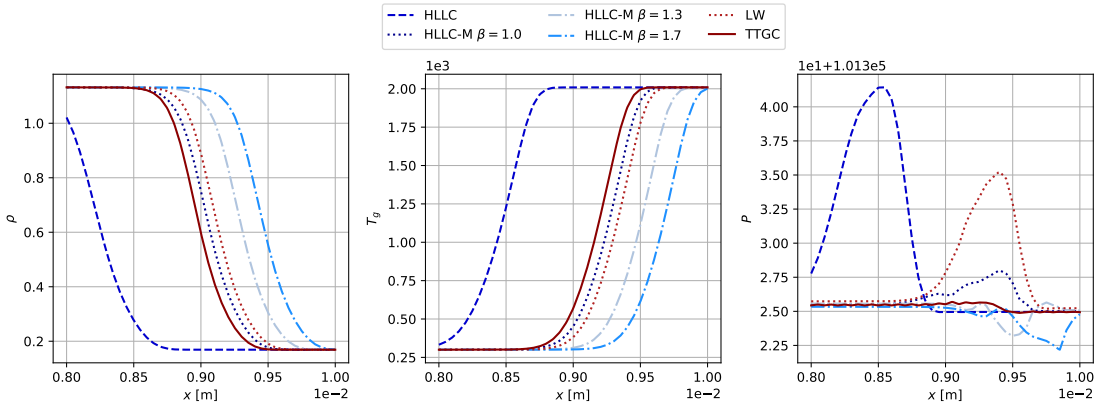
From the results of all the validation cases performed, the HLLC MUSCL RK2 scheme with Sweby parameter β between 1 and 1.3 is the best option for combustion simulation that include shocks. A summary of the non-dimensionalized cost of Riemann solvers compared to the most robust and fastest scheme of AVBP, LW, is given in Tab. 8.5. This performance test has been carried out with the strong Convective Vortex test case. HLLC is in between LW and TTGC in terms of cost. The MUSCL procedure adds another 45% cost with its



(a) $t = 4\,\text{ms}$



(b) $t = 32\,\text{ms}$



(c) $t = 64\,\text{ms}$

Figure 8.27: One-dimensional flames using Riemann solvers and classical AVBP schemes at $t = 4, 32$ and 64 ms.

LW	1
TTGC	2.3
HLLC	1.7
HLLC-MUSCL RK1	2.5
HLLC-MUSCL RK2	4.6

Table 8.5: Non-dimensionalized cost of the various Riemann solvers implemented.

gradient evaluation and limiting procedure. This cost is doubled for the HLLC-MUSCL RK2 since two time integration has been shown to be necessary for stability. Hence the scheme is rather costly compared to LW as it is 4.6 times slower.

Part III

Numerical Simulation of PAC

Chapter 9

Chemistry and Zero-Dimensional Reactors

9.1 Introduction to paper

The derivation and validation of a plasma assisted combustion chemistry is carried out in this paper accepted for publication in Combustion and Flame. We use the CANTERA code [Goodwin et al., 2021], with added features for plasma simulations, to conduct the validation of this detailed plasma assisted combustion chemistry against experiments. Most notably, the EEDF resolution is included in these plasma features. Three key properties are targeted: fast heating, radical production and vibrational slow heating. The reduction of this mechanism is then carried out to yield a smaller mechanism, suited for multi-dimensional simulations using AVIP.

Combustion and Flame 240 (2022) 111990

Contents lists available at ScienceDirect

Combustion and Flame

journal homepage: www.elsevier.com/locate/combustflame

Plasma assisted combustion of methane-air mixtures: Validation and reduction

L. Cheng[a,*], N. Barleon[a], B. Cuenot[a], O. Vermorel[a], A. Bourdon[b]

[a] CERFACS, 42 Avenue Gaspard Coriolis, Toulouse, Cedex 1 31057, France
[b] Laboratoire de Physique des Plasmas (LPP), CNRS, Sorbonne Université, Ecole Polytechnique, Institut Polytechnique de Paris, Palaiseau 91120, France

ARTICLE INFO

Article history:
Received 28 September 2021
Revised 7 January 2022
Accepted 8 January 2022

Keywords:
Plasma assisted combustion
Chemistry
Discharges
Reduction

ABSTRACT

For several years now plasma assisted combustion has been the subject of intense research due to stabilization effects a plasma can have on flames. Particularly, experiments have shown the promising impact of Nanosecond Repetitively Pulsed discharges on combustion while not exceeding an energy consumption of a few percent of the flame power. In this work, an incremental methodology with a step-by-step approach has been used to build a single plasma mechanism upon which combustion is added using the GRI 3.0 and Konnov v0.6. The methodology focuses on three key aspects of plasma assisted combustion: fast gas heating, slow gas heating and radical production. Selected experiments focusing on one or more of these aspects allow to validate the mechanism in large ranges of temperature (300-1500 K) and pressure (0.1-1 bar) in air, methane-air and argon diluted mixtures using glow and spark discharges. These experiments include a plasma assisted ignition case on which the ignition delay time is well captured by the mechanism. Slow gas heating has been modeled using a vibrational relaxation model validated against a detailed vibrational description. Discussions on ambiguous rates for critical reactions of excited nitrogen quenching are made in the light of their impact on the results on the chosen experiments. Finally, the resulting 100-species GRI 3.0-based and 264-species Konnov v0.6-based plasma mechanisms are reduced to make them suitable for multi-dimensional simulations. The DRGEP reduction method, based on plasma experiments and canonical combustion cases, is applied allowing to reduce the number of species by a factor larger than two. For the GRI-3.0 plasma mechanism, the reduced mechanism contains 47 species and 429 reactions. Hence significant performance is gained, opening the way to multi-dimensional simulations of plasma assisted combustion.

1. Introduction

In the context of lean combustion for reduced environmental impact, plasma-assisted combustion is currently investigated by many groups to address stability and ignition issues.

Among the various types of discharges, Nanosecond Repetitively Pulsed (NRP) discharges have been experimentally shown to have a significant impact on flame stabilization [1] and ignition [2,3]. NRP discharges produce non-thermal plasmas and active radical species which lead to increased efficiency [4,5].

Although their beneficial effect is well-known [6], the mechanisms at play in plasma-assisted combustion using NRP discharges are still not fully understood. In particular, the coupling between plasma physics and combustion thermochemistry is not yet clearly established. As this is difficult to address with experiments, numerical simulations may be used to analyze the flame-discharge interaction. Most of the studies including detailed kinetics are limited to 0D simulations [7–9]. One dimensional case [10,11] or two-dimensional simulations [12,13] of plasma-assisted methane combustion are usually limited to simplified chemistry. To avoid excessive computational cost of a detailed chemistry in a 3D configuration, a phenomenological model which mimic plasma discharge effects has been recently used in [14] allowing to model several hundred of discharges. A trade-off between detailed and phenomenological models is to use chemical reduction: by choosing suitable targets and cases, accuracy is preserved while keeping the computational cost reasonable. This method has already proven its worth in a combustion framework [15,16] and is considered as an interesting basis for the reduction of plasma-combustion kinetic mechanism [17].

Contrary to thermal discharges, NRP discharges provide a chemical effect on top of the thermal effect due to the non-equilibrium electrons that allow to efficiently produce excited states of atoms and molecules through electron-neutral collisions. In air, elec-

* Corresponding author.
E-mail address: cheng@cerfacs.fr (L. Cheng).

https://doi.org/10.1016/j.combustflame.2022.111990
0010-2180/

tronically excited states contribute to fast gas heating by spontaneous dissociation or quenching of electronically excited nitrogen molecules as identified by Popov in [18]. Vibrationally excited states also play a role in gas heating allowing to store energy into the vibrational modes of nitrogen molecule which is further released through vibration-translation (VT) and vibration-vibration (VV) relaxations [19].

Having consistent and detailed chemistry is thus a critical basis for a good description of plasma-assisted combustion. Plasma air mechanisms have been derived and validated in [7,20]. Despite good results from [20], a full description of the mechanism is lacking. Implementation of [7] does not give satisfactory results on one critical test case [21] because the mechanism was not validated at high temperature ($\simeq 1500K$) and atmospheric pressure. Complete plasma air sets that are concatenated from different sources can also be found in [22,23]. However, the validity of these sets of reactions is not verified for plasma-assisted combustion application. An important review of plasma-assisted combustion chemistry has been done in [24]. While the most important processes and their effects are highlighted, it is not straightforward to derive a complete kinetic mechanism relying solely on [24]. In [8], a detailed plasma-assisted combustion mechanism has been used to study ignition with pre-calculated electron impact reaction rates. Although computationally efficient, this strategy limits the conditions on which the detailed mechanism is valid. Indeed, important changes in initial conditions such as mixture composition can drastically change electron impact reaction rates, thus requiring their recalculations. The self-consistent computation of these rates during simulations through an Electron Energy Distribution Function (EEDF) solver such as [25] is thus more appropriate to derive a versatile mechanism.

The present work introduces new detailed plasma-assisted combustion mechanisms that are valid for a wide range of temperatures ([$300K - 1500K$]) and pressures ([0.1 bar - 1.0 bar]). Critical reaction rates, such as those of nitrogen excited states quenching, are not unambiguously established in the literature. These uncertainties lead to noticeable differences in two major features of plasma discharge effects: fast gas heating and radical production. In [26], the reaction rate for the quenching of N_2(B, C) with O_2 is said to be gas kinetics, meaning an underlying hard-sphere model for the cross-sections. On the other hand, experiments from [21] derived another rate for the quenching of N_2(C) with O_2 from measurements which highlights a temperature dependence that is not captured by a hard-sphere model. The quenching of N_2(B, C, a) with O introduced in [26] for discharges with high dissociation degree is questionable as it was not brought to light in any experiment. Fast gas heating is a major component of any plasma-assisted combustion mechanism: in this work, the straightforward approach of considering the enthalpy change of a reaction for gas heating is compared with the fast gas heating model developed in [18].

The iterative process of construction of such a mechanism is highlighted in Section 3: after having built a plasma air mechanism, plasma methane reactions are added. Combustion mechanisms are finally applied on top of it with a specific merging procedure. Systematic validation of mechanisms against several experiments [21,27–29], presented in Section 4, is performed after each upgrade in Section 5, thus validating different aspects of the chemistry: fast and slow gas heating as well as radical production. The fast gas heating model of [18] is tested and faced against the convential definition of heat release of reactions. Effects of uncertainties on the N_2^* quenching rates by O_2 and O are evaluated in terms of gas heating and radical production. The validation across a large range of temperature, composition and pressure prevents any overfitting of a mechanism to match a specific experiment.

As the mechanism is rather computationally expensive for dimensional simulations, a reduction step is mandatory. In [17], an extension of the DRGEP [15] reduction method was employed to reduce the mechanism of [30]. In this work, the original DRGEP method developed in ARCANE [16] is successfully employed to reduce the detailed mechanism in Section 6 without the need of any plasma specific procedure by choosing suitable reduction cases.

2. Modeling plasma assisted combustion

Zero-dimensional reactors have been chosen as first test cases to study plasma assisted combustion. The mass fractions Y_k and the gas temperature T_g are governed by:

$$m\frac{\mathrm{d}Y_k}{\mathrm{d}t} = V\dot{\omega}_k \tag{1}$$

$$mc_v\frac{\mathrm{d}T_g}{\mathrm{d}t} = -P\frac{\mathrm{d}V}{\mathrm{d}t} + V\dot{\omega}_T + \dot{Q} \tag{2}$$

where m is the mass of the mixture, t the time, $\dot{\omega}_k$ the mass reaction rate of species k, P the pressure, V the volume, c_v the mass heat capacity at constant volume, $\dot{\omega}_T$ the heat release rate and $\dot{Q}$ is a power source term explicited in Section 3.3.

In low temperature plasmas, the ionization degree, *i.e.* the proportion of charged species in the mixture, is rather low, reaching at most $10^{-4} - 10^{-3}$. However, thanks to the strong electric field in plasma discharges, electrons are much more energetic than other heavy species. The resulting electron-neutral collisions can efficiently produce reactive species such as radicals or excited states, driving the electron distribution far from a Maxwellian one. It is thus important to know the electron distribution function $f_e(\mathbf{r}, \mathbf{v}_e, t)$, with $\mathbf{r}$ the position and $\mathbf{v}_e$ the electron velocity, which obeys the Boltzmann equation:

$$\frac{\partial f_e}{\partial t} + \mathbf{v}_e \cdot \nabla f_e - \frac{e\mathbf{E}}{m_e} \cdot \nabla_v f_e = \left(\frac{\delta f_e}{\delta t}\right)_c \tag{3}$$

where $\mathbf{E}$ is the electric field and m_e the electron mass. ∇_v corresponds to gradient in velocity space. The collision term $(\delta f_e/\delta t)_c$ is computed from a complete set of cross sections containing mostly electron-neutral collisions.

Assuming a two-term expansion, the electron distribution function f_e can be written as the product of the EEDF F_0 and the electron density n_e [25]. The Boltzmann equation Eq. (3) is then recast in terms of F_0 and solved. It is then used to compute the reaction rate k_j associated to the cross-section σ_j and the effective electron temperature T_e by integrating over energy ε [25]:

$$k_j = \gamma \int_0^\infty \varepsilon\sigma_j F_0 \, d\varepsilon \tag{4}$$

$$T_e = \frac{2}{3}\int_0^\infty \varepsilon^{3/2} F_0 d\varepsilon \tag{5}$$

with $\gamma = \sqrt{\frac{2e}{m_e}}$ a constant and $\varepsilon = (v/\gamma)^2$ the electron energy in electron-volt. In typical discharges, T_e rises up to 5 eV.

As a consequence, in the mass reaction rate $\dot{\omega}_k$ and heat release rate $\dot{\omega}_T$ of Eqs. (1) and (2), two contributions may be distinguished: the set of EEDF reactions $\mathcal{B}$ (linked to an EEDF cross-section - superscript b) and the set of classical chemical reactions $\mathcal{C}$ (all the other ones - superscript c):

$$\dot{\omega}_k = \dot{\omega}_k^b + \dot{\omega}_k^c = \sum_{j\in\mathcal{B}} \dot{\omega}_{kj} + \sum_{j\in\mathcal{C}} \dot{\omega}_{kj} \tag{6}$$

$$\dot{\omega}_T = -\sum_{j\in\mathcal{B}} \dot{Q}_j(\Delta u_j - \varepsilon_{\mathrm{th}}^j) - \sum_{j\in\mathcal{C}} \dot{Q}_j \Delta u_j \tag{7}$$

where $\varepsilon_{\text{th}}^{j}$ is the energy lost during process j taken from cross-section datasets, $\Delta u_j = \sum_k \nu_{kj} u_k$ is the reaction energy change per mole with u_k the internal energy of species k. $\dot{Q}_j$ is the molar production rate of reaction j and is proportional to the reaction rate k_j. For the set of reactions $\mathcal{B}$, the reaction rate k_j is given by Eq. (4) while for the set of reactions $\mathcal{C}$ an extension of Arrhenius law including electron temperature T_e is employed:

$$k_j = A_j T_g^{\beta_j} \exp\left(-\frac{E_a}{RT_g}\right) T_e^{\beta'_j} \exp\left(-\frac{E'_a}{RT_e}\right) \tag{8}$$

In this work, the CANTERA code [31] has been used to solve Eqs. (1) and (2). An EEDF solver has been implemented in CANTERA, based on an open-source version of BOLSIG+ [25] named BOLOS, written by Alejandro Luque. Some features such as the energy grid adaptation of the open-source solver LOKI-B [32] have been introduced in our EEDF solver.

3. Chemical mechanisms

The plasma-assisted combustion mechanism is built in two steps. First, a chemical mechanism that describes the plasma dynamics is built based on the literature. Then, this mechanism is adapted to account for specific gas heating features of a plasma discharge. Finally, the plasma mechanism is merged with a combustion mechanism. In this work, two well-established schemes for methane-air combustion are tested: the GRI 3.0 [33] and the Konnov v0.6 [34].

3.1. Plasma mechanism

The cross sections for electron collisions with neutrals are taken from various datasets of the LXCat database (see Table 1) and associated reactions are included in the mechanism. Elastic cross-sections have been included for the (B, C, a) electronic states of N_2, assumed to be the same as for the ground state N_2. The electronically excited triplet states $N_2(B^3\Pi_g)$ - $N_2(B'^3\Sigma_u^-)$ - $N_2(W^3\Delta_u)$ and singlet states $N_2(a^1\Pi_g)$ - $N_2(a'^1\Sigma_u^-)$ - $N_2(w^1\Delta_u)$ are lumped into the $N_2(B^3\Pi_g)$ and $N_2(a^1\Pi_g)$ states respectively. To preserve the discharge energy transfer to the gas mixture, the difference of excitation energies is assumed to be converted into gas heating when produced by electron impact reactions. This procedure is illustrated in Reaction (9) where the $N_2(W^3\Delta_u)$ state is lumped into $N_2(B^3\Pi_g)$. Consequently, the difference of energies between these two states must be accounted for into the heat release of the reaction.

$$e^- + N_2 \longrightarrow e^- + N_2(W^3\Delta_u) \longrightarrow e^- + N_2(B^3\Pi_g) + 0.81\text{eV} \tag{9}$$

Electron impact reactions need to be supplemented with electron-ion and heavy species reactions to form a complete kinetic mechanism for plasma discharge. Two references ([22, Chap. 7–10], [18]) have been chosen to construct the plasma mechanism with which the combustion mechanism will be merged. As argon is often used as diluent in plasma assisted combustion studies [29], a quite simple plasma chemistry for Ar has been taken from [35] to handle mixture dilution with this noble gas. The detailed plasma mechanism finally contains 71 species and 680 reactions. A complete description of the plasma mechanism can be found in the supplementary materials S2.

3.2. Fast gas heating modeling

Usually the enthalpy change of a reaction is associated with heat release leading to an increase or decrease of the gas temperature. Table 2 illustrates the different behaviors of EEDF reactions (set $\mathcal{B}$) with two examples. In the first process of Table 2, the whole energy lost by the electron is transferred to the electronic state $N_2(A)$. Consequently, there is no heat release in this reaction. However, some electron impact processes can release heat. This is the case of the second process of Table 2 in which the electron looses more energy than necessary to dissociate O_2. This excess of energy is associated to gas heating in the kinetic modeling.

Table 1
Electron impact cross-sections used in this work.

Colliding partners	σ
N_2, $N_2(A^3\Sigma_u^+)$, $N_2(B^3\Pi_g)$, $N_2(C^3\Pi_u)$, $N_2(a^1\Pi_g)$, O_2, Ar	Phelps [41]
$O_2(a^1\Delta_g)$, $O_2(b_1\Sigma_g^+)$	Triniti [53]
CH_4, NO, N_2O	Hayashi [54]
$O(^3P,^1D,^1S)$, $N(^4S,^2D)$	IST-Lisbon [40]
O_3	Morgan [55]
$N_2(v)$	[40–42]

Table 2
Example of electron impact reactions.

#	Process	ε_{th} [eV]	Δu [eV]	$\Delta u - \varepsilon_{\text{th}}$ [eV]
1	$e^-(\varepsilon) + N_2 \longrightarrow e^-(\varepsilon - \varepsilon_{\text{th}}) + N_2(A)$	6.17	6.17	0
2	$e^-(\varepsilon) + O_2 \longrightarrow e^-(\varepsilon - \varepsilon_{\text{th}}) + 2O$	6.0	5.16	-0.84

For the other reactions (set $\mathcal{C}$), part of the products energy is stored into rotational and vibrational modes as suggested by Popov [18,36] and references therein. To account for this phenomenon, the heat release rate of some reactions is forced to the Fast Gas Heating (FGH, Model 1) values listed in Table 3 according to the work of [18,36]. In this table, FGH corresponds to the forced energy release per molecule (either observed experimentally or computed using the methodology of [18]) and HRR (Model 2) refers to the use of the conventional definition of Heat Release of Reaction using the formation enthalpy of the reactants and products. At the end, energy conservation is ensured through a vibrational energy residual ε_{vib} in the products following Eq. (10), discussed in more details in Section 3.3.

$$\varepsilon_{\text{vib}} = \text{HRR} - \text{FGH} \tag{10}$$

Among these reactions, it is well known that the dissociative quenching of electronically excited states N_2^* with O_2 (Reaction (11)) largely contributes to the fast gas heating and the production of atomic oxygen.

$$N_2^* + O_2 \longrightarrow N_2 + 2O \tag{11}$$

In particular, $N_2(B^3\Pi_g)$, $N_2(a^1\Pi_g)$ and $N_2(C^3\Pi_u)$ dissociate O_2 with a high reaction rate while $N_2(A^3\Sigma_u^+)$ dissociates O_2 at a rate two orders of magnitude lower at room temperature [37]. In [7], the rate of reaction (11) was taken constant for $N_2^* = N_2(B^3\Pi_g)$, $N_2(a^1\Pi_g)$, $N_2(C^3\Pi_u)$ at a value of $3 \times 10^{-10}\text{cm}^3\text{s}^{-1}$. Similar reaction rates are used in [37] for $N_2(B^3\Pi_g)$ and $N_2(C^3\Pi_u)$. While global agreement on the dissociative quenching rates for $N_2(A^3\Sigma_u^+)$, $N_2(B^3\Pi_g)$ and $N_2(C^3\Pi_u)$ is reached in the literature, the case of $N_2(a^1\Pi_g)$ is ambiguous. In most of the plasma-assisted combustion studies, the rate for $N_2(a^1\Pi_g)$ is one order of magnitude smaller than $N_2(B^3\Pi_g)$ and $N_2(C^3\Pi_u)$ states. For the $N_2(a^1\Pi_g)$ state, the rate coefficient used in [37] ($2.8 \times 10^{-11}\text{cm}^3\text{s}^{-1}$) was taken from [38] and corresponds to the quenching of the $N_2(a'^1\Sigma_u^-)$ state. In [39], the rate coefficient of the quenching of $N_2(a^1\Pi_g)$ with O_2 has been measured experimentally and was found high at $4.3 \times 10^{-10}\text{cm}^3\text{s}^{-1}$. The ratio of singlet excited states $N_2(a^1\Pi_g)$ production by electron impact reactions in air calculated with BOLSIG+ is depicted in Fig. 1, showing that $N_2(a^1\Pi_g)$ is dominant among $N_2(a'^1\Sigma_u^-, a^1\Pi_g, w^1\Delta_u)$ for electric fields higher than 100 Td with more than 50% of the over-

Table 3
Fast Gas Heating (FGH) and conventional Heat Release of Reaction (HRR) for reactions for which they differ.

Num	Reaction	Model 1		Model 2
		FGH [eV]	Ref	HRR [eV]
(A1)	e- + N_4+ ⟶ N_2(v) + N_2(C)	0	[56]	3.4
(A2)	N_2(A) + O_2 ⟶ N_2(v) + O_2(b1)	0	[18]	4.53
(A3)	N_2(A) + N_2(A) ⟶ N_2(v) + N_2(C)	0.4	[18]	1.3
(A4)	N_2(A) + N_2(A) ⟶ N_2(v) + N_2(B)	3.5	[18]	4.99
(A5)	N_2(B) + O2 ⟶ N_2(v) + 2 O	1.1	[18, Eq. 6]	2.19
(A6)	N_2(B) + N_2 ⟶ N_2(A) + N_2(v)	0	[56]	1.18
(A7)	N_2(a) + N_2 ⟶ N_2(B) + N_2(v)	0	[56]	1.05
(A8)	N_2(C) + O2 ⟶ N_2(v) + O + O	5.55	[18, Eq. 6]	5.87
(A9)	N_2(C) + O2 ⟶ N_2(v) + O + O(^{1}D)	3.59	[18, Eq. 6]	3.99
(A10)	N_2(C) + O2 ⟶ N_2(v) + O + O(^{1}S)	1.37	[18, Eq. 6]	1.68
(A11)	N_2(C) + N_2 ⟶ N_2(B) + N_2(v)	0	[18,56]	3.68
(A12)	O(^{1}D) + O2 ⟶ O + O2(v)	0	[18]	1.97
(A13)	O(^{1}D) + O2 ⟶ O + O2(a1, v)	0	[18]	0.99
(A14)	O(^{1}D) + O2 ⟶ O + O2(b1, v)	0	[18]	0.34
(A15)	O(^{1}D) + N_2 ⟶ O + N_2(v)	1.38	[18]	1.97

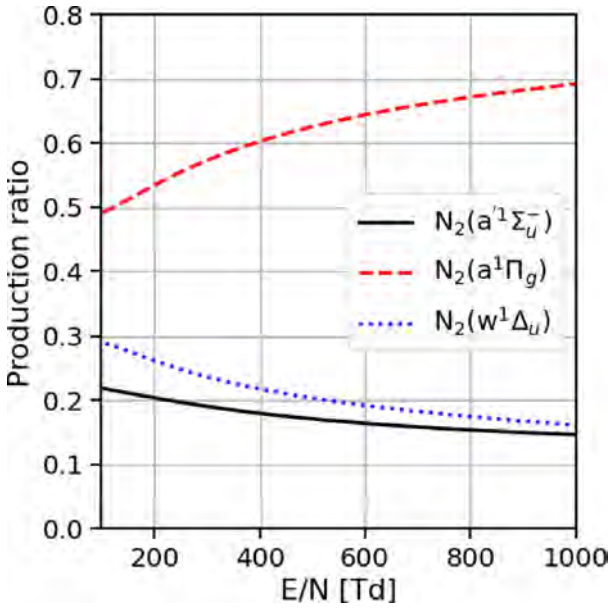


Fig. 1. Ratio of production rates of singlet excited states of N_2 using BOLSIG+ [25].

all production of singlet states. The state $N_2\left(a'^1\Sigma_u^-\right)$ only represents about 20% of the singlet states produced by the discharge.

The timescale of (11) for $N_2\left(a^1\Pi_g\right)$ is of the order of 3ns in air at $T = 1500K$ and atmospheric pressure assuming $X_{O_2} = 0.15$. According to [22], the $N_2\left(a^1\Pi_g\right) \rightarrow N_2\left(a'^1\Sigma_u^-\right)$ optical transition occurs on the millisecond timescale, which is much higher than the timescale of (11). The $N_2\left(a^1\Pi_g\right)$ collisional quenching timescale of reaction (12) is also higher by one order of magnitude (20ns) compared with the O_2 dissociative quenching timescale. Thus, during the very first nanoseconds following the pulse, it can be concluded that $N_2(a'^1\Sigma_u^-, a^1\Pi_g, w^1\Delta_u)$ is mainly composed of $N_2\left(a^1\Pi_g\right)$.

$$N_2\left(a^1\Pi_g\right)+M \longrightarrow N_2\left(a'^1\Sigma_u^-\right)+M, \qquad k = 9\times10^{-12}cm^3s^{-1} \quad (12)$$

Finally, three different models for gas heating are tested, and referred as follows. Model 1 considers the forced energy release of the reaction in Table 3, Model 2 refers to the use of the conventional HRR and Model 3 is based upon Model 1 with the rate from [39] for the dissociative quenching of O_2 by $N_2\left(a^1\Pi_g\right)$.

3.3. Slow gas heating modeling

The fast gas heating produced by a plasma discharge is followed by a slow gas heating corresponding to the relaxation of vibrational states. For the discharges considered in this work ($E/N \geq$ 10Td), an overwhelming majority of this energy is stored into vibrational modes of N_2 [1]. Thus, a detailed or simplified modeling of N_2(v) is included in the kinetic mechanism and a simplified model for O_2(v) is always used. In the detailed model VIB. 1, the electron vibrational excitation cross sections are taken from [40] and [41] for N_2(v=1-8) and from [42] for N_2(v=9-17). The main relaxation processes are vibrational-translational (VT) N_2-O and vibrational-vibrational (VV) N_2-N_2 reactions and as a first approximation only those processes have been accounted for in the detailed model:

$$N_2(v) + O \longleftrightarrow N_2(v-1) + O \quad (13)$$

$$N_2(v) + N_2(w-1) \longleftrightarrow N_2(v-1) + N_2(w) \quad (14)$$

The vibrational energy of level v, E_v, accounts for anharmonicity effects, using a Morse potential energy expression:

$$\frac{E_v}{hc} = \omega_e\left(v+\frac{1}{2}\right) - \omega_e x_e\left(v+\frac{1}{2}\right)^2 \quad (15)$$

where h is the Planck constant, c the speed of light in vaccum, ω_e the characteristic vibrational pulsation and $\omega_e x_e$ the anharmonicity. ω_e and $\omega_e x_e$ are given in units of wavelength and can be found in [22]. The base rate and scaling for both VV and VT relaxations are taken from [19]. The resulting mechanism contains 116 species (71 species of the plasma mechanism developed above plus 45 vibrational states N_2(v=1,45)) and 1916 reactions.

The modeling of the vibration kinetics of species k can be simplified by considering the mean non-equilibrium vibrational energy e^k_{vib}. The time evolution of e^k_{vib} and its interaction with the gas temperature are given by:

$$\frac{de^k_{\text{vib}}}{dt} = \dot{E}^k_{\text{vib}} - \dot{R}^k_{\text{VT}} \quad (16)$$

$$mc_v\frac{dT_g}{dt} = -P\frac{dV}{dt} + V\dot{\omega}_T + \sum_k \dot{R}^k_{\text{VT}} \quad (17)$$

In Eq. (16), $\dot{E}^k_{\text{vib}}$ corresponds to vibrational excitation of species k produced by the discharge. Reactions such as (A6) and (A7) do not produce any heating and instead the energy is deposited as

Table 4
Major plasma reactions of CH_4 containing mixture.

Reaction	k_f $[cm^3.s^{-1}]$	Ref.
Electron-impact reactions		
$e^- + CH_4 \longrightarrow e^- + CH_3 + H$	σ	[57]
$e^- + CH_4 \longrightarrow 2e^- + CH_4+$	σ	[57]
Electron-ion recombination		
$e^- + CH_4^+ \longrightarrow CH_3 + H$	$1.18 \times 10^{-8}(300/T_g)^{0.5}$	[58]
$e^- + CH_4^+ \longrightarrow CH_2 + 2H$	$2.42 \times 10^{-8}(300/T_g)^{0.5}$	[58]
$e^- + CH_4^+ \longrightarrow CH + H_2 + H$	$1.41 \times 10^{-8}(300/T_g)^{0.5}$	[58]
Quenching of excited states		
$N_2(A) + CH_4 \longrightarrow N_2 + CH_3 + H$	3.3×10^{-15}	[28]
$N_2(B) + CH_4 \longrightarrow N_2 + CH_3 + H$	3×10^{-10}	[28]
$N_2(C) + CH_4 \longrightarrow N_2 + CH_3 + H$	5×10^{-10}	[28]
$N_2(a) + CH_4 \longrightarrow N_2 + CH_3 + H$	3×10^{-10}	[28]
$O(^1D) + CH_4 \longrightarrow CH_3 + OH$	1.89×10^{-10}	[59]
$O(^1D) + CH_4 \longrightarrow CH_3O + H$	3.1×10^{-11}	[59]

molecular vibration in $\dot{E}^{N2}_{vib}$. The sink term $\dot{R}^k_{VT}$ corresponds to the vibrational-translational (VT) relaxation from species k:

$$\dot{R}^k_{VT} = \frac{e^k_{vib}}{\tau^k_{VT}} \qquad \tau^k_{VT} = \left(\sum_p 1/\tau^{pk}_{VT} \right) \tag{18}$$

where θ^k_v represents the characteristic vibrational temperature. Each relaxing species k has several colliding partners (denoted by p above). Note that the power source term $\dot{Q}$ of Eq. (2) has been expanded as $\dot{Q} = \sum_k \dot{R}^k_{VT}$.

In the simplified model VIB. 2, the mean vibrational energy of $k = N_2$ is considered with the colliding partners $p = O, N_2$ and the relaxation times τ^{pk}_{VT} are taken from [43].

In both models VIB. 1 and VIB. 2, the relaxation of the vibrational energy of O_2 is modeled using the simplified approach with the colliding partners $p = O, N_2, O_2$ and relaxation times taken from [1]. Note that O_2(a1, v) and O_2(b1, v) produced respectively by reactions (A14) and (A15) in Table 3 are modeled like O_2(v).

Using Model 1 (FGH) for reactions listed in Table 3 introduces reactions which produce vibrational species at an unknown distribution. It is thus impossible to include them in a detailed model of vibration and the corresponding vibrational energy residual is relaxed using Eq. (18), i.e., following model VIB. 2.

3.4. Combustion mechanisms and merging with plasma mechanism

To compute plasma-assisted combustion, methane is incorporated to the previous plasma air mechanism. First, plasma reactions involving methane are added to the mechanism. The most important ones are listed in Table 4 along with their references. Full description of the plasma methane kinetics can be found in the supplementary materials S2. Methane is able to create H radicals and is another possible partner for electronically excited nitrogen and oxygen.

Combustion reactions now need to be added to the mechanism. Two combustion mechanisms are considered here to illustrate the merging mechanisms with chemistries of increasing complexity: the GRI-3.0 [33] mechanism (53 species, 325 reactions) and the Konnov [34] mechanism (201 species, 2300 reactions). Two global quantities of these mechanisms, namely the ignition delay time and the flame speed S_L, are given in Fig. 2. Auto-ignition is computed for stoichiometric mixture by varying initial temperature and flame speed is computed with a fresh gas temperature and pressure of 300 K and 1 atmosphere by varying the equivalence ratio. Regarding auto-ignition times in Fig. 2a, both mechanisms are in agreement when dealing with standard conditions. However, the addition of a small amount of atomic oxygen $\Delta Y_O = 1 \times 10^{-3}$ to mimic a plasma chemical effect reveals a noticeable difference between the two mechanisms. For both mechanisms, the addition of atomic oxygen reduces the ignition delay time but this effect is much stronger for the GRI-3.0 mechanism. Concerning flame speeds shown in Fig. 2b, a difference of about 10% is observed between the two mechanisms in the lean-to-stoichiometric region for standard conditions. The addition of atomic oxygen leads to a slightly higher flame speed in the stochiometric and stochiometric-rich regions for the GRI-3.0 and Konnov mechanisms respectively. Overall these preliminary results show that the two mechanisms behave similarly in standard conditions but their response to plasma conditions can be quite different. The procedure for merging the plasma and combustion mechanisms can now be detailed.

As the methane-air plasma mechanism was developed independently from the combustion mechanisms, some reactions are redundant. For the GRI 3.0 approximately 40 reactions already exist in the plasma mechanism while for the Konnov mechanism this number is close to 100. While some of these reactions use the same rate constants, most do not and a choice must be made.

3.4.1. GRI 3.0

In the case of GRI 3.0 mechanism, choosing the rate constants from one or the other mechanism has negligible effects on the plasma cases studies in this paper (see Section 4). Thus to be safe the rates of GRI 3.0 are kept as they were optimized for combustion applications.

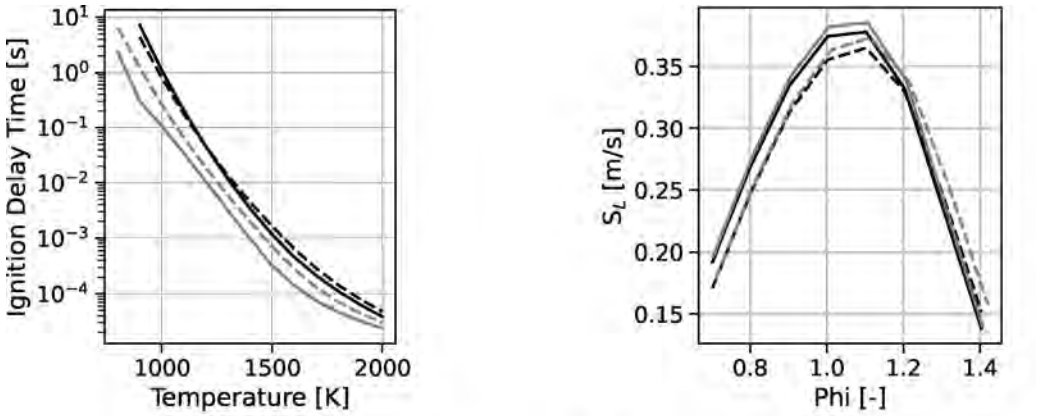
3.4.2. Konnov v0.6

As for the GRI 3.0, the rates of reactions involving atoms and molecules in their ground states can be taken from either mechanism without significant change in the cases of Section 4. However the Konnov mechanism [34] contains some excited species as detailed in [44]: $O(^1D)$, $O_2(\Delta)$ and $OH(\Sigma)$ denoted “OX”, “O2X” and "OHX" respectively. The reactions involving these species are much more important than those with ground state species as they contain part of the discharge energy and need to be treated carefully. Some reaction rates have been modified from the original Konnov mechanism. Due to the merging of the singlet states of O_2 in [34], the quenching of $O(^1D)$ by O_2 is described in the original Konnov mechanism by:

$$O(^1D) + O_2 \longrightarrow O + O_2(a1) \quad k = 2.64 \times 10^{-11} \exp(139/RT_g) \tag{19}$$

As our chemistry considers two distinct species O_2(a1) and O_2(b1), the rate coefficients of $O(^1D)$ quenching by O_2 are described more precisely as in [22]:

$$O(^1D) + O_2 \longrightarrow O + O_2 \qquad k = 6.4 \times 10^{-12} \exp(133/RT_g) cm^3 s^{-1} \tag{20}$$



(a) Auto-ignition at $\phi = 1$ (b) Flame speed at $T_0 = 300\,\mathrm{K}$, $P_0 = 1$ atm

Fig. 2. Comparison of the GRI-3.0 (solid lines) and Konnov (dashed lines) at normal composition (black) and with small atomic oxygen addition $\Delta Y_O = 1 \times 10^{-3}$ (gray) to mimic plasma discharge.

Table 5
Modified reactions in the Konnov mechanism according to our plasma mechanism. FGH and vibrational excitation are given in term of percentage of the energy change of reaction.

Reaction	FGH [%]	Vibration [%]	k_f $[cm^3.s^{-1}]$
$O(^1D) + O_2 \longrightarrow O + O_2(v)$	0	100	$6.4 \times 10^{-12} \exp(67/T_g)$
$O(^1D) + O_2 \longrightarrow O + O_2(a1, v)$	0	100	10^{-12}
$O(^1D) + O_2 \longrightarrow O + O_2(b1, v)$	0	100	$2.6 \times 10^{-11} \exp(67/T_g)$
$O(^1D) + N_2 \longrightarrow O + N_2(v)$	30	70	7.6×10^{-11}

$$O(^1D) + O_2 \longrightarrow O + O_2(a1) \qquad k = 1 \times 10^{-12} \mathrm{cm^3 s^{-1}} \tag{21}$$

$$O(^1D) + O_2 \longrightarrow O + O_2(b1) \quad k = 2.6 \times 10^{-11} \exp(133/RT_g) \mathrm{cm^3 s^{-1}} \tag{22}$$

The reaction rate used for reaction (19) actually corresponds to that of reaction (22). The underlying assumption is that the O_2(b1) formed in reaction (22) is directly quenched to form O_2(a1). This assumption is relaxed in the merging procedure as reaction (19) is replaced by reactions (20), (21) and (22).

Modifications have also been made to ensure correct repartition of the discharge energy: the heat release rate of some reactions was split into a fast gas heating and a slow gas heating following Eq. (10) based on [18]. Hence the three quenching reactions above are supposed to produce only vibrational energy. Likewise the quenching of reaction (23) is described as in [18] with 30% of energy going into FGH and 70% into vibration. A summary of the modifications made on energy repartition and reaction rates for the merging with the Konnov mechanism is summarized in Table 5.

$$O(^1D) + N_2 \longrightarrow O + N_2(v) \tag{23}$$

4. Test cases

4.1. Cases overview

Four experiments of NRP discharges taken from the literature are used to validate the models and chemistries described in Section 3. Their major characteristics are summarized in Table 6. These discharges correspond to typical electrode configurations depicted in Fig. 3 and cover either the glow [45] or spark [46] regimes of NRP discharges for a wide range of gas pressure and temperature ($0.1 - 1bar$ and $300 - 1500K$ respectively). For the plane to plane electrode configurations [28,35], experiments highlighted quasi-uniform discharges, thus validating the use of 0D reactors. For the pin-pin configurations [21,27], the validity of a 0D approach is limited to the first instants. Indeed, according to the acoustic timescale $\tau_a = r_d/a$ where r_d is the radius discharge and a the speed of sound, the discharge remains isochoric only until 300 ns and 2 µs in the conditions of [21] and [27] respectively. As both isochoric and isobaric reactor formulations are available in CANTERA, for each experiment the formulation yielding the best results compared with experiments has been kept. These choices are in agreement with previous simulations for cases B [28], C [7,27] and D [35]. Concerning case A for which no 0D simulation is available in the literature, the isochoric formulation is chosen because of the short timescale involved compared to the acoustic timescale τ_a.

The mixture compositions are either air or air-CH_4 with argon dilution in the plasma assisted ignition case of [35]. With this combination of cases, a large amount of experimental data is available such as time resolved gas temperature and species concentration or ignition times. In Section 5, these data will be used to test and validate the fast and slow gas heating, the radical production and finally the coupling of the discharge with conventional combustion kinetics. Each experiment is described in more details in the following.

4.2. Case A

This experimental setup is presented in [21]. NRP discharges are applied in air in a 4mm pin-to-pin configuration. The discharges are 10 ns long with an applied voltage of 5.7kV between the electrodes at a frequency of 10 kHz. The gas temperature was estimated through measurements of the N_2(B,C) rotational temperature. However, more recent work [47] shows that the rotational distribution of the excited states depends on various kinetic processes including electronic excitation, collisional quenching and rotational excitation. A detailed model was developed to account for these processes and enabled to infer the actual gas temperature from the measured rotational distributions of the N_2(B) and N_2(C)

Table 6
Main characteristics of selected experiments.

Name	Case A	Case B	Case C	Case D
Reference	Rusterholtz et al. [21]	Uddi et al. [28]	Montello et al. [27]	Aleksandrov et al. [35]
T_g [K]	1500–2500	300	300	1330
P	1 atm	60 Torr	100 Torr	0.45 atm
Geometry	Pin-Pin	Plane	Pin-Pin	Plane
Reactor	Isochoric	Isochoric	Isobaric	Isobaric
Mixture	Air	Air-CH_4	Air	Ar:N_2:O_2:CH_4
Data	T_g, O, N_2(B,C)	O	T_g, N_2(v)	Ignition time

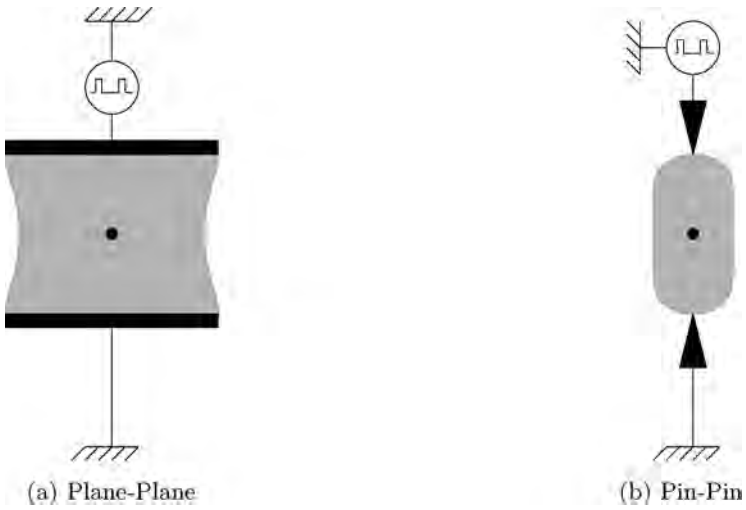


Fig. 3. Sketch of electrode configurations. The gray area corresponds to the discharge region. The point indicates the location modelled by the 0D reactor.

states. After multiple pulses the temperature of the gas just before a pulse is 1500 K with a temperature rise of about 900 K in 20 ns. Optical measurements allow access to electron, nitrogen electronic states (N_2(B), N_2(C)) and atomic oxygen densities. At the end of the discharge around 50% of oxygen is dissociated and the energy of the discharge is $E_d = 670 \pm 50\,\mu$J with 35% of this energy used for dissociation of O_2 and 21% for gas heating.

For the CANTERA 0D simulations, the reduced electric field is taken from experimental measurements [21] while the electron density at the center of the discharge is taken from a 1D simulation [20]. Initially, the mixture composition is N_2:O_2:O:77.4:18.6:4 accounting for O_2 dissociation from previous discharges.

4.3. Case B

This experiment is presented in [27]. NRP discharges are studied in plane-to-plane and pin-to-pin geometries and the focus is made on vibration populations of nitrogen. The discharges are either applied in burst mode (40 to 150 pulses) in the plane-to-plane configuration or in mono-pulse mode in the pin-to-pin configuration. It is found that in these conditions up to 50% of the discharge energy is stored into nitrogen vibrational modes. The initial temperature and pressure are $T_0 = 300K$ and $p_0 = 100Torr$ respectively. The discharges are approximately 100 ns long. In this work, only the pin-to-pin configuration has been simulated. Smooth electron density and reduced electric field profiles from [27] are imposed in the 0D simulations.

4.4. Case C

The experiment described in [28] contains two planar electrodes in a rectangular geometry that produce NRP discharges in pure air and CH_4-air mixtures at 300 K and 60 Torr. The pulses are 25 ns long and can be applied in single mode or in burst mode, in which case from 2 to 100 pulses are applied at 100 kHz. Measurements of atomic oxygen after the discharges are given as well as simulation results.

The simulations of [28] have been made with a reduced electric field and an electron density of Gaussian shape with a 20 ns FWHM (Full Width at Half Maximum). The peak reduced electric field is given at 247 Td and the peak electron density is chosen to satisfy the peak atomic oxygen mole fraction in air. These parameters are kept constant for the discharge in CH_4-air. The same procedure has been applied in the CANTERA simulations.

4.5. Case D

A shock tube setup to study ignition using NRP discharges is presented in [29]. A shock wave is initiated using a broken diaphragm and passes between two electrodes after reflection where the discharge is applied.

The simulated case, taken from [35], is characterized by an initial temperature $T_0 = 1330K$, an initial pressure $P_0 = 0.45bar$ and an initial composition Ar:N_2:O_2:CH_4-80:15:4:1. The electric field is deduced from the measurements of the electric potential at two different locations. As the discharge is modelled in a 0D framework it is necessary to impose the electron density profile during the pulses which is deduced from the discharge energy density given in [35].

5. Validation of the mechanisms

0D simulations performed with the plasma-assisted combustion mechanism of Section 3 are now compared with the experimental results for the four configurations described in Section 4. Each experiment allows to validate one or more aspects of the chemistry: gas heating, radical production and ignition time.

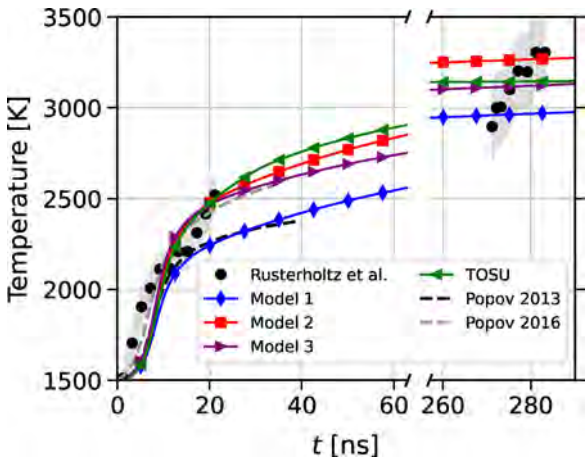


Fig. 4. Gas temperature time evolution for Case A [21] using the different fast gas heating models of Section 3 and TOSU mechanism against experimental measurements (point) [21] and reference simulations (dashed: Popov 2013 [20], Popov 2016 [37]). The filled areas correspond to the experimental uncertainties of [21].

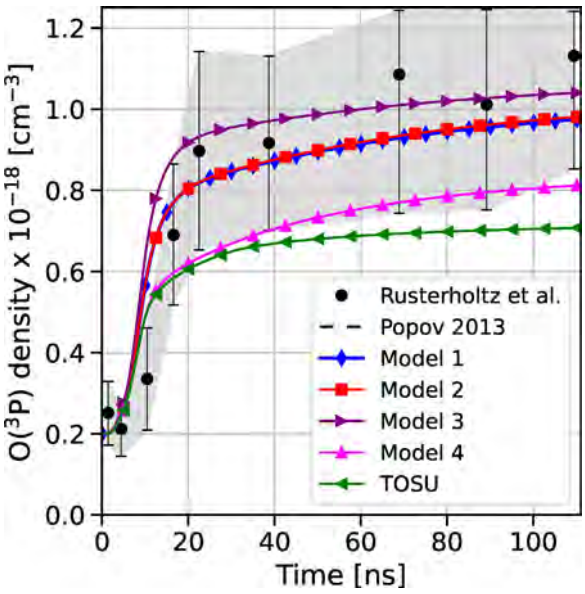


Fig. 5. Time evolution of atomic oxygen density for Case A [21] using different fast gas heating model against experimental measurements (point) [21] and reference simulations (dashed line) [20]. The filled areas correspond to the experimental uncertainties.

5.1. Fast gas heating

The NRP discharges of Case A [21] produce fast and intense gas heating during the first tens of nanoseconds. The temperature rise is used to assess the quality of the three models of fast gas heating presented in Section 3.2. In addition to these models, simulation has been performed using the kinetic mechanism of The Ohio State University (TOSU) [7,30]. Simulation and experimental results are shown in Fig. 4, along with previous simulation results of Popov [20,37]. In this figure, only the GRI 3.0 results are shown for readability since similar conclusions apply to the Konnov mechanism.

In Fig. 4, while Models 2 and 3 are close, a 300 K temperature difference is observed with Model 1 after 60 ns which corresponds to $\approx 30\%$ of the gas temperature increase. Both Models 2 and 3 are in good agreement with the measurements, hence other measured quantities must be used to differentiate them. The TOSU mechanism yields a gas temperature increase similar to models 2 and 3. In the work of Popov 2016 [37], the model developed in [20] has been updated to account for quenching reactions of electronically excited nitrogen by atomic oxygen. Despite gas temperature prediction improvements, a loss of accuracy on atomic oxygen production induced by this quenching is shown in the next section.

Additional information was obtained in the experiment thanks to a secondary reflected pulse at $t = 270$ns as shown in Fig. 4. This pulse was strong enough to efficiently stimulate the C states of N_2 allowing an estimation of its rotational temperature. At the very beginning of this pulse, recent work [47] inferred a gas temperature of $3110K \pm 200K$ from the excited state C rotational temperature, indicating a significant gas temperature increase of about $600K \pm 200K$ from the end of the first pulse. With 0D simulations, the gas temperature ranges between 2900 and 3300 K depending on the model used, Model 1 staying approximately 200 K lower than the others. However, at a high temperature of about $T = 2500K$, the compression wave formed by the fast gas heating during the first pulse can not be neglected to analyze the results of the second pulse. Indeed, the acoustic timescale τ_a is of the order of 200 ns in a volume of size 225 µm and a speed of sound $a = \sqrt{\gamma RT_g/M} \simeq 1000\text{ms}^{-1}$. This rapid gas expansion is responsible for gas temperature decrease, which is a sign that gas temperature should be even higher than the one predicted by Models 2 and 3. However, the 0D model is not able to accurately estimate the density change resulting from the compression wave. Hence the reduced electric field can not be computed so that the pulse reflection was not included in our 0D simulation. Therefore the simulated temperature stays flat contrary to the experimental measurements. The gas dynamics effects could be analyzed in further works using 1D or 2D simulations.

5.2. Radical production

Radical production is another important characteristic of the discharge studied in the experiment of Case A [21]. The time evolution of $O(^3P)$ density has been measured using Two-Photon Absorption Laser Induced Fluorescence (TALIF) technique during the pulse and the afterglow. The atomic oxygen from CANTERA simulations is displayed in Fig. 5 for the different models studied. For the same reasons as in the fast gas heating study, only the results for GRI 3.0 mechanism is shown. Models 1 to 3 yield satisfactory results when compared with experimental measurements [21] and Popov's simulation [20]. Model 3 exhibits a slighlty higher O production of ~ 8 % than Model 1 which seems to be in better agreement with the experimental results. However, the TOSU mechanism [7] underestimates the atomic oxygen by approximately 30%. This is attributed to the quenching reaction of N_2^* with $O(^3P)$ atoms (Eq. (24)) which occurs with a very high rate. The addition of this reaction in Model 1 leading to Model 4 of Fig. 5 strongly disturbs the atomic oxygen production. This observation questions the effectiveness of reaction (24) in our conditions.

$$N_2^* + O(^3P) \longrightarrow NO + N(^2D) \quad k = 3 \times 10^{-10}\text{cm}^3\text{s}^{-1} \tag{24}$$

Case B [28] setup monitors atomic oxygen decay after a discharge in both air and methane-air mixtures [28]. Konnov and GRI plasma mechanisms using Model 3 give satisfactory results concerning the rate of decay of atomic oxygen compared with experimental results and reference simulations as shown in Fig. 6. In pure air, the difference between the two combustion mechanisms is due to the different rates used for O_3 chemistry. Atomic oxygen decay in methane-air mixtures are similar using both mechanisms.

5.3. Slow gas heating

Slow gas heating is the result of the relaxation of vibrationally excited nitrogen molecules. The detailed simulation using VIB. 1

Table 7
Ignition time and maximum values of radical mole fractions.

τ_{ig}	X_O	X_H	X_{CH3}	Reference
332 µs	2.79×10^{-3}	2.7×10^{-4}	2.47×10^{-4}	[35]
135 µs	3.43×10^{-3}	3.06×10^{-3}	5.94×10^{-4}	GRI 3.0 (This work)
334 µs	3.14×10^{-3}	2.67×10^{-3}	7.26×10^{-4}	Konnov v0.6 (This work)
448 µs	2.92×10^{-3}	2.56×10^{-3}	6.78×10^{-4}	Konnov v0.6 NO FGH (This work)

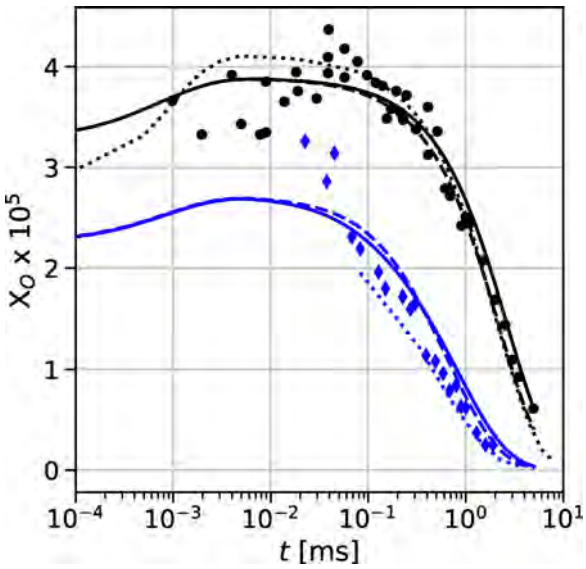


Fig. 6. Time evolution of atomic oxygen mole fraction for Case B [28] using Model 3 along with GRI 3.0 (solid) or Konnov (dashed) mechanisms. Measurements (point) [28] and reference simulations (dotted) [28] are also given for comparison. Black and blue colors are respectively used for air and air-methane cases. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

model on the Montello experiment yields good results for the gas temperature (Fig. 7a) as well as for the vibrational populations (Fig. 7b) over time irrespective of the combustion mechanism. The VV N_2-N_2 and VT N_2-O are thus the main vibrational relaxation channels in air. The simpler VIB. 2 model shows good agreement with the detailed model for the gas temperature in Fig. 7a, validating the method and the relaxation times considered. This approach is able to drastically reduce the number of species in the mechanism and must be considered as a valuable option for multi-dimensional works. For these reasons VIB. 2 is kept for all other calculations.

5.4. Ignition

In case D, the discharge used [29] is characterized by a combination of radical production and fast gas heating promoting the ignition process. The gas heating during the discharge remains relatively low which leads to a gas temperature increase of about 100 K. The ignition delay time τ_{ig} measured experimentally is compared with the one extracted from the simulation using the same procedure based on the OH mole fraction. τ_{ig} is defined at the intersection of two OH density tangents: one at the minimum value and the other at the inflection point during the sharp increase corresponding to the ignition process. This method is described in Fig. 8a. The ignition delay times obtained for the different combustion mechanisms are given in Table 7. A very good agreement with the experimental value $\tau_{ig}^{exp} = 376\,\mu s$ [29] is found for the case using the Konnov mechanism. The case using the GRI-3.0 mechanism exhibits a 50% lower ignition time compared to the Konnov mechanism, which is consistent with the preliminary study of ignition delay time performed in Section 3.4. An additional simulation, referred to as "NO FGH", was performed without gas heating during the discharge phase (i.e. $t \leq 1\,\mu s$) as done in [35]. In that case, the ignition delay time increases to 448 µs showing the importance of accounting for gas heating effect during the discharge.

Table 8
Summary of targets and errors used in the reduction process.

Cases	Targets	Errors
Case A [21]	O, $\dot{\omega}_T$	T_g (10%), n_O^{end} (10%)
Case C [28]	O	n_O^{end} (10%)
Case D [35]	O, CO, $\dot{\omega}_T$	τ_{ig}(10%)
Plasma Dynamics	e^-	n_e^{max} (5%), n_e^{end} (25%)
Auto-ignition	CO, $\dot{\omega}_T$	τ_{ig}(5%)
1D flame	CO, $\dot{\omega}_T$	s_L(5%), T_g^{max}(1%)

6. Chemistry reduction

The mechanisms derived above for methane-air plasma-assisted combustion are computationally heavy with respectively 100 and 226 species along with 964 and 2586 reactions for the GRI-3.0 [33] and Konnov [34] plasma mechanisms. Using such mechanisms in a multi-dimensional simulation is impossible and a reduction step must be performed to decrease the computational cost. The methodology used to reduce these mechanisms is first described in Section 6.1. The test cases used to validate the reduction process are then presented in Section 6.2. The resulting reduced mechanism for GRI-3.0 plasma mechanism is finally presented and evaluated in Section 6.3 in terms of computational cost and accuracy. GRI-3.0 plasma mechanism has been chosen due to its lower computational cost but the methodology also applies to any plasma assisted combustion mechanism.

6.1. Methodology

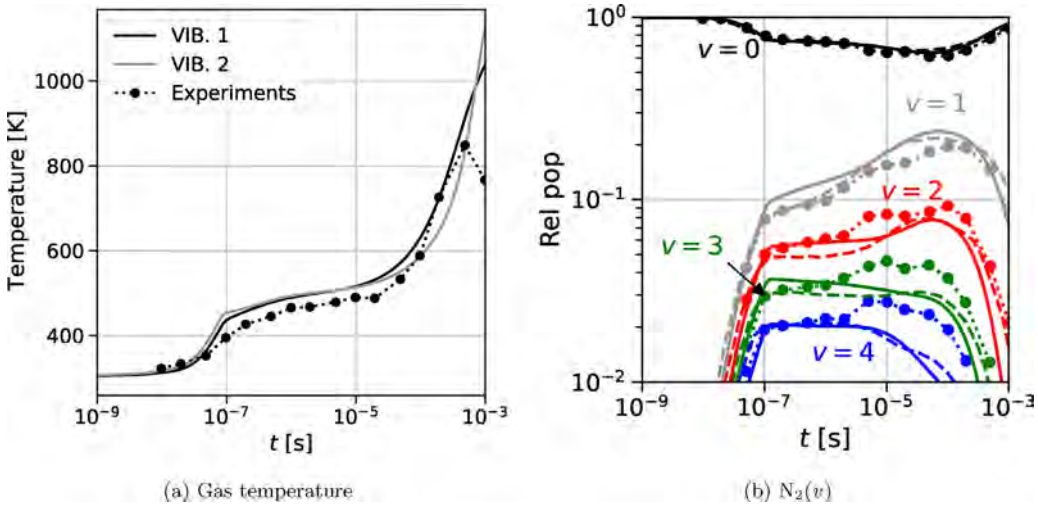
The methodology is the one of [15] using Directed Relation Graph with Error Propagation (DRGEP), implemented in the ARCANE code co-developed at CERFACS and Cornell University [16]. The graph relies on coefficients which are computed on a set of representative cases (reduction cases) that span the range of conditions on which the reduced mechanism will be used.

The DRGEP method was recently applied in [17], where contrary to Eq. (5) the time evolution of electron temperature T_e is governed by:

$$c_{ve} m_e n_e \frac{dT_e}{dt} = -\dot{\omega}_e u_e + \dot{\mathcal{Q}}_p \tag{25}$$

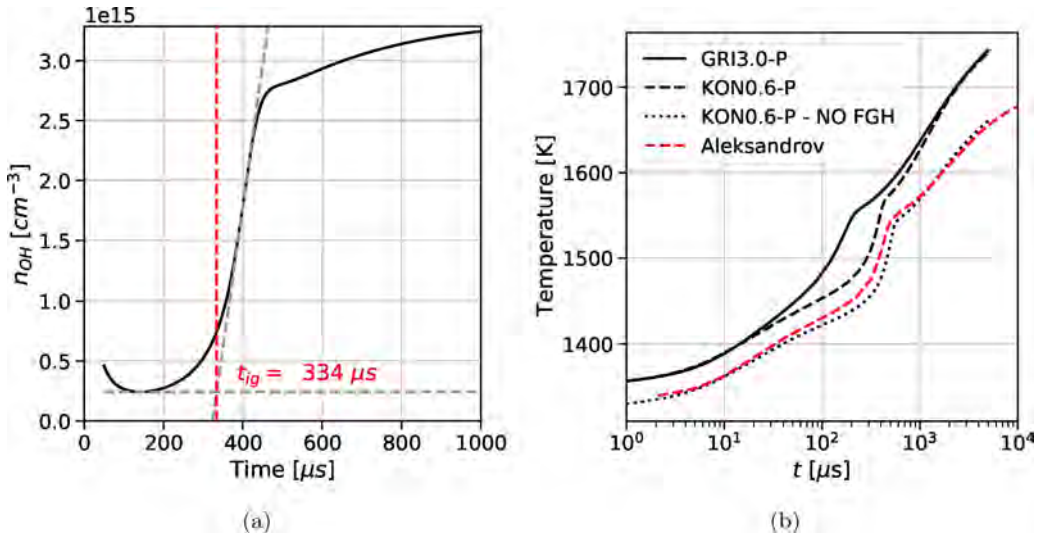
where n_e is the electron density, c_{ve} the electron specific heat at constant volume, $\dot{\omega}_e$ the electron mass production rate, u_e the electron specific internal energy and $\dot{\mathcal{Q}}_p$ the net energy source term which depends on T_e and electron concentration c_e.

In this work, the reaction rate coefficients of electron impact reactions $\mathcal{B}$ and electron temperature T_e are deduced from the EEDF (Eqs. (4)-(5)). Thus, these quantities remain unaltered as long as the EEDF remains the same. Differences could be induced by modified gas temperature or mixture composition after successive re-



(a) Gas temperature (b) $N_2(v)$

Fig. 7. Comparison of vibrational model on Case C [27] (a) gas temperature (solid, black and gray for simulation) against experimental measurements (point-dotted); (b) vibrational populations over time, $v = (0, 1, 2, 3, 4)$ correspond respectively to black, grey, red, green, blue for VIB. 2 model (solid), experimental measurements [27] (point-dotted) and reference simulation [7] (dashed). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



(a) (b)

Fig. 8. Results of Case D [35] (a) ignition time delay for the simulation using Konnov mechanism and (b) gas temperature.

duction steps since the EEDF depends on these quantities. However, no significant effect of that kind was observed in the cases considered here.

A major property to be preserved during the reduction process is the energy that is transferred from the discharge to the electrons and ultimately to the gas mixture. In the present model, the electron temperature is computed self-consistently with the reduced electric field, gas composition and temperature. With this approach, the discharge energy transfer to the gas mixture is preserved only if all cross-sections considered in the EEDF are associated to a reaction in the kinetic mechanism, which strongly limits the reduction. A method to relax this constraint is presented in the following.

While the set of cross-sections used to compute the EEDF is immutable, the associated reactions can be simplified. For instance, the production of an electronically excited state AB* by electron impact reaction on a species AB can be replaced by an inert heat production process described in Eq. (26). The underlying assumption is that the quenching of the excited state of a molecule does not change its formula but only transfers its excitation energy into a translational mode. For reaction Eq. (27), the chemical effect is kept when possible by quenching the excited species B* in the products, as done previously. In a case for which species B does not exist, the reaction reduces to Eq. (26). Applying theses rules allows to remove excited species AB* and B* while keeping the total energy transfer to the gas unchanged, thus reducing the mechanism.

$$e^- + AB \longrightarrow e^- + AB^* \longrightarrow e^- + AB + \text{heat} \tag{26}$$

$$e^- + AB \longrightarrow e^- + A + B^* \longrightarrow e^- + A + B + \text{heat} \tag{27}$$

6.2. Reduction cases

The aim of the reduced mechanism is to be used in a multi-dimensional plasma-assisted combustion simulation. The test cases on which the reduction process is based must therefore be representative of the various phenomena occuring in this type of simulation. The chosen set of reduction cases is detailed in the following and a summary can be found in Table 8.

6.2.1. Plasma and combustion

Plasma cases are selected to validate fast gas heating and radical production of the reduced mechanism. The discharge described

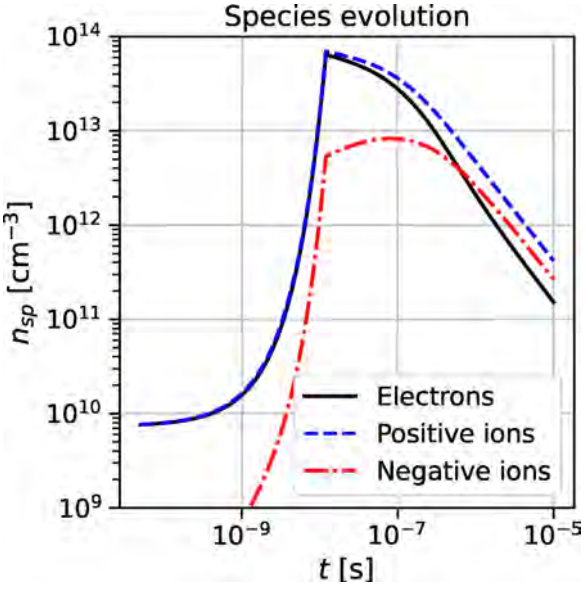


Fig. 9. Time evolution of electron and ion density for Plasma Dynamics case using the detailed mechanism.

in [21] highlights these two effects in air while [28] allows to validate radical production in a CH_4 containing mixture. In both cases, the atomic oxygen concentration is targeted and the error on the final atomic oxygen density is checked. Due to the important gas heating, the heat release is also targeted in [21] and the error on the final gas temperature is also assessed.

Canonical combustion cases and targets (CO, $\dot{\omega}_T$) are chosen following [16]: a 0D isochoric reactor with an initial temperature of 1000 K and a 1D flame with a fresh gas temperature of 300 K. Both combustion cases are performed for stoichiometric CH_4-air mixtures at atmospheric pressure. The error on auto-ignition delay time τ_{ig} is checked on the 0D case while the laminar flame speed S_L and maximum temperature T_g^{max} are assessed on the 1D flame case.

Finally, the plasma-combustion coupling is assessed on the case of single pulse plasma-assisted ignition of [35]. For these 0D combustion and plasma cases, the targets are: O, CO and $\dot{\omega}_T$. The error on the ignition delay time τ_{ig} is used to validate the reduced mechanism.

6.2.2. Charged species dynamics

In all the studied cases A to D used so far in this work, the time evolution of the electron density is imposed. Consequently, electroneutrality must be imposed and then the growth and decay of charged species are not monitored by reduction cases of Section 6.2.1. They are crucial in multi-dimensional simulations as they control the discharge propagation [48–50]. A simple reference test-case without an imposed electron density profile, referred to as Plasma Dynamics, is therefore included to reproduce the net ionization rate and plasma decay phase.

For this case, a 12 ns pulse with constant reduced electric field $E/N = 200$ Td is used to mimic a discharge leading to a peak electron density of $\approx 1e14\text{cm}^{-3}$ as in [49]. The plasma decay is simulated up to 10 µs, which is representative of a 100 kHz repetition frequency of an NRP discharge. These are typical parameters used experimentally for NRP discharges [21,46,51] The initial mixture is a stoichiometric methane-air mixture at 1000 K and atmospheric pressure.

The time evolution of electron and ion density obtained with the GRI-3.0 plasma mechanism developed in this work is shown in Fig. 9. An increase of charged species density is observed during the first 12 ns followed by a plasma decay phase. Electron density is considered as the target for DRGEP coefficients and the error on the maximum value of electron density n_e^{max} and its final value n_e^{end} is systematically checked after the removal of a species.

Table 9
Species kept in the reduced mechanism using the cases of Table 8.

	Species
Combustion	O, O_2, H, OH, H_2, HO_2, H_2O_2, CH, CO, CH_2, HCO, CH_2(S) CH_3, CH_2O, CH_4, CO_2, CH_3O, CH_3OH, C_2H_4, C_2H_5, C_2H_6 H_2O, N_2, AR, N, NO, NO_2, H_2CN
Plasma	e^-, N_2^+, O_2^+, CH_4^+, CH_3^+, AR^+, NO^+, O^-, O_2^- N_2(A), N_2(B), N_2(a), N_2(C), N(^{2}D), O_2(a1) O(^{1}D), O(^{1}S), O_3, AR*

6.3. Results

The reduction was performed starting from the GRI 3.0 mechanism updated with our plasma mechanism using the simplified vibrational model VIB. 2 resulting in a detailed mechanism composed of 100 species and 950 reactions. The reduction performed with ARCANE led to a mechanism of 47 species and 429 reactions. Species included in the reduced mechanism are listed in Table 9: 28 species come from the GRI 3.0 mechanism while 19 species are kept from the plasma mechanism. A more detailed comparison of reduced and detailed mechanisms in the different reduction cases can be found in the supplementary materials S1.

Another reduction considering only 0D auto-ignition and 1D flame cases (i.e. without plasma cases) was also performed for comparison. In that case, five other species of the GRI 3.0 mechanism (CH, CH_3OH, N, NO_2 and H_2CN) are removed. This highlights the importance of new chemical paths in plasma-assisted combustion and validates the reduction strategy including both plasma and combustion chemistries.

The reduction procedure adopted here keeps the effective electron temperature almost unchanged as shown in Fig. 10a contrary to what was observed in [17]. This is a consequence of the use of an EEDF solver within the plasma reactors in our simulations. As pointed out previously, it must be verified that the energy transferred from electrons to heavy molecules is preserved during the reduction. In Fig. 10b, the total energy has been split into chemical, heating, vibrational and excitation channels to precisely identify the potential changes caused by the reduction. While chemical and vibrational energies remain unchanged, the heating energy slightly increases to compensate for a decrease in the excitation energy. This is a consequence of the procedure described by reactions (26) and (27). The total energy transferred from the electrons to the gas in the Case A [21] is well reproduced by the reduced mechanism with a final error lower than 0.2%.

The reliability of the reduced mechanism has been also evaluated outside of the reduction conditions. An energy density of $8e6\text{Jm}^{-3}$ is deposited in a single pulse at a constant reduced electric field between 150 and 300 Td to ignite a mixture initially at $T_g = 300K$, $P = 1$ atm and equivalence ratio $\phi = 1$. Simulations have been performed in a CH_4-air mixture and with 80% argon dilution. The auto-ignition delay times for detailed and reduced mechanisms along with relative errors are depicted in Fig. 11. In the argon diluted case shown in Fig. 11a, the error on the ignition delay time remains lower than 10% for a large range of reduced electric field. This error is particularly low from 150 to 200 Td where most of the reduction cases operate. For a higher reduced electric field, it is observed that the ignition delay time increases. This can be explained by the change in dominant electron-impact reactions from electronic excitation to ionization. While electronically excited species are able to efficiently produce radi-

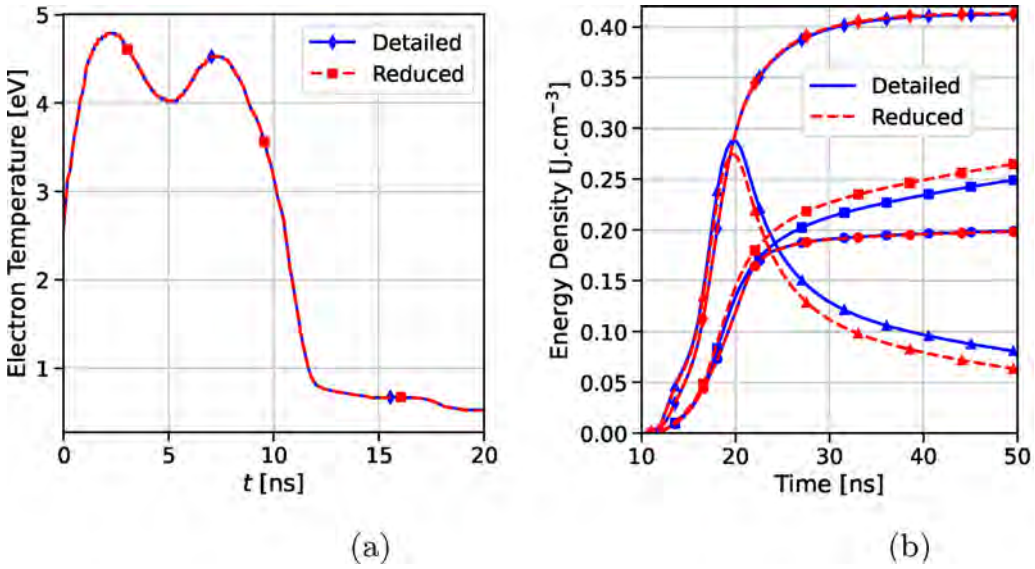


Fig. 10. Comparison of the detailed and reduced mechanisms on (a) the effective temperature of the electrons (Case A) and (b) the discharge energy transfer in different channels (Case A) (▲ Electronic, ● Vibration, ■ Heat, ♦ Chemical).

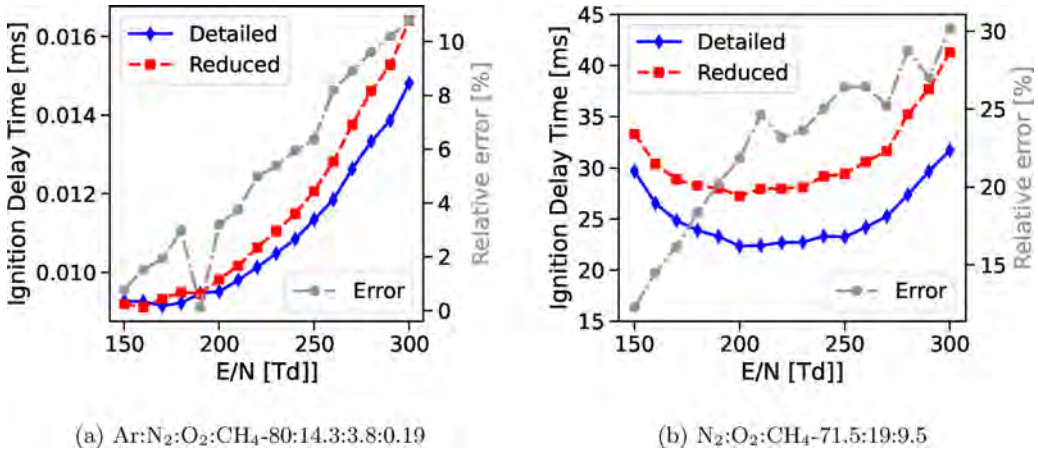


Fig. 11. Comparison of detailed and reduced mechanisms on plasma-assisted ignition delay times at various reduced electric fields for (a) argon dilluted mixture and (b) CH_4-air mixture.

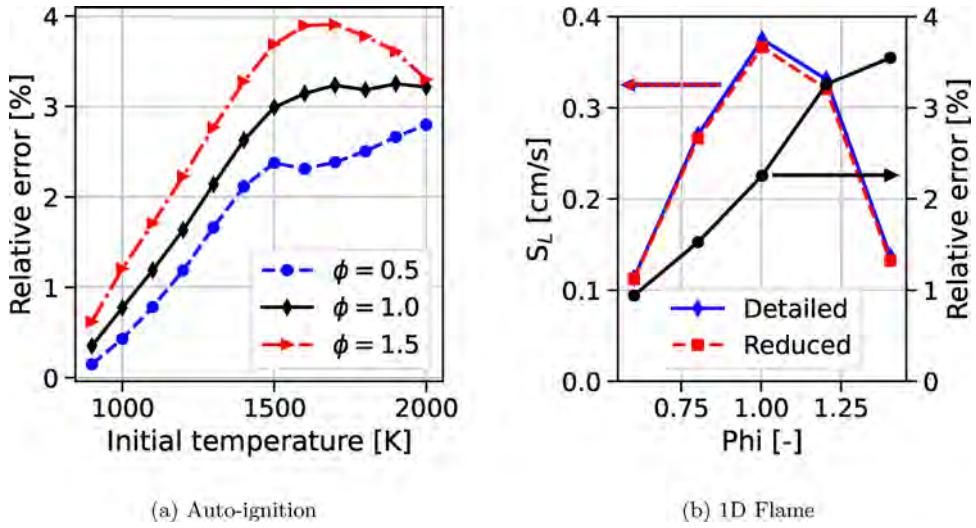


Fig. 12. Comparison of detailed and reduced mechanisms: (a) auto-ignition time (relative error) and (b) laminar flame speed (value and relative error).

cals, ionized species mostly recombine which results in gas heating. The formation of active radicals is more efficient to ignite a mixture compared to heat deposition leading to smaller ignition delay times. The reduced mechanism is less accurate in the CH_4-air mixture as depicted in Fig. 11b, highlighting the noticeable importance of argon dilution on plasma-assisted ignition behavior. Indeed, when argon is the dominant mixture component, most of the electron energy is transferred to Ar*, Ar+ which then transfer their energy to the other molecules. However, in CH_4-air mixture, electron energy transfer is more complex as a wide variety of processes exist (i.e. vibrational excitation and different electronic states of N_2). In Fig. 11b, the minimum ignition delay time is found around 200 Td, which corresponds to the case where electronic excitation is at its maximum. The slower ignition observed around 150 Td and 300 Td, respectively due to an increase of vibrational and ionization energies, illustrates the efficiency of the electronic excitation.

A comparison of the reduced and detailed mechanisms on the combustion features is shown in Fig. 12. The error on the auto-ignition delay time remains lower than 5% on a large range of temperature [800–2000] K and equivalence ratio [0.5-1.5] as shown in Fig. 12a. Despite the use of stoichiometric conditions in the auto-ignition test cases during the reduction, a very good agreement is found for lean and rich mixtures. Satisfying results are also obtained for the laminar flame speed in a large range of equivalence ratio [0.6-1.4] as demonstrated in Fig. 12b.

7. Conclusion

In this work, a consistent and detailed plasma mechanism has been derived from [18,22,23] following an incremental methodology. After construction of the plasma chemistry, combustion is added considering methane as a fuel using the GRI-3.0 [33] and Konnov v0.6 [34] mechanisms. Common reactions between plasma and combustion chemistries are treated carefully to yield plasma assisted combustion mechanisms.

These mechanisms have been tested on zero-dimensional reactors using Cantera [31] with a two-temperature model incorporating the resolution of the EEDF [25,32]. The focus has been made on three aspects of the chemistry: fast gas heating, slow gas heating and radical production.

To validate those three aspects, four experiments have been chosen [21,27,28,35]. Temperature measurements in [21,27] validate the fast gas heating reactions of the chemistry. The temporal evolution of vibrational population in [27] verifies the detailed description of vibration. Comparison with a relaxation vibrational model yields good results regarding slow gas heating allowing to reduce the size of the kinetic scheme. Finally radical production of atomic oxygen is monitored in the experiments of [21,52] and a good agreement is found. In accordance with electron impact rates, $N_2\left(a^1\Pi_g\right)$ is dominant among $N_2(a'^1\Sigma_u^-,\ a^1\Pi_g,\ w^1\Delta_u)$ and the rate of O_2 dissociation by $N_2(a'^1\Sigma_u^-,\ a^1\Pi_g,\ w^1\Delta_u)$ has been chosen accordingly [39]. Compared to [20] which uses rates from [38], better results are found in [21]. Combination of these effects is tested on the plasma assisted ignition case of [29,35]. Only on this case, noticeable differences arise between the GRI-3.0 and Konnov v0.6 plasma assisted combustion mechanisms, the latter being closer to experimental results. The mechanisms are thus valid between 300-1500 K, 0.1-1 bar and 150–300 Td in air, methane-air and argon diluted mixtures using glow and spark discharges. To the authors' knowledge, validation of plasma-assisted combustion mechanisms on such a variety of configurations and conditions is a novelty in the community.

GRI-3.0 and Konnov v0.6 plasma mechanisms contain respectively 100 and 264 species and 964 and 2860 reactions. In the purpose of using those mechanisms in multi-dimensional simulations, chemical reduction is necessary. Using the ARCANE code co-developed at CERFACS and Cornell University with a DRGEP reduction algorithm [15], the number of species has been reduced by a factor of 2, yielding a 47-species, 429-reactions reduced mechanism from the GRI-3.0 plasma mechanism. During the reduction process, essential properties of plasma assisted combustion have been targeted. Reduction cases include real cases from experiments [21,29,52] which allow to monitor the errors of the reduced mechanism against physical cases with confidence. This opens the way to reliable multi-dimensional simulations that could be done using the low temperature plasma code AVIP [60] coupled to the combustion code AVBP [61], both developed at CERFACS.

When testing the reduced mechanisms on plasma assisted ignition in CH_4-air mixtures, less than 30% error is observed on ignition delay times. This is a satisfying result since the conditions considered were not targeted within the reduction. Experimental cases of such configurations would benefit the construction of more reliable chemistries at those conditions.

Declaration of interest

none

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the ANR projects PASTEC (ANR-16-CE22-0005) and GECCO (ANR-17-CE06-0019). The authors thank Prof. Christophe Laux (EM2C, CentraleSuplec) for the fruitful discussions.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.combustflame.2022.111990.

References

[1] A. Starikovskiy, N. Aleksandrov, Plasma-assisted ignition and combustion, Prog. Energ. Combust. 39 (1) (2013) 61–110.
[2] G. Pilla, D. Lacoste, D. Veynante, C. Laux, Stabilization of a swirled propane-air flame using a nanosecond repetitively pulsed plasma, IEEE T. Plasma Sci. 36 (2008) 940–941.
[3] A.Y. Starikovskii, N.B. Anikin, I.N. Kosarev, E.I. Mintoussov, M.M. Nudnova, A.E. Rakitin, D.V. Roupassov, S.M. Starikovskaia, V.P. Zhukov, Nanosecond-pulsed discharges for plasma-assisted combustion and aerodynamics, J. Propul. Power 24 (6) (2008) 1182–1197.
[4] C.D. Cathey, T. Tang, T. Shiraishi, T. Urushihara, A. Kuthi, M.A. Gundersen, Nanosecond plasma ignition for improved performance of an internal combustion engine, IEEE T. Plasma Sci. 35 (6) (2007) 1664–1668.
[5] S. Bozhenkov, S. Starikovskaia, A. Starikovskii, Nanosecond gas discharge ignition of h_2 and CH_4 containing mixtures, Combust. Flame 133 (1) (2003) 133–146.
[6] Y. Ju, J.K. Lefkowitz, C.B. Reuter, S.H. Won, X. Yang, S. Yang, W. Sun, Z. Jiang, Q. Chen, Plasma assisted low temperature combustion, Plasma Chem. Plasma P. 36 (1) (2016) 85–105.
[7] I.V. Adamovich, T. Li, W.R. Lempert, Kinetic mechanism of molecular energy transfer and chemical reactions in low-temperature air-fuel plasmas, Philos. T. R. Soc. A 373 (2048) (2015) 20140336.
[8] A.C. DeFilippo, J.-Y. Chen, Modeling plasma-assisted methane-air ignition using pre-calculated electron impact reaction rates, Combust. Flame 172 (2016) 38–48.
[9] N. Deak, A. Bellemans, F. Bisetti, Plasma-assisted ignition of methane/air and ethylene/air mixtures: efficiency at low and high pressures, P. Combust. Inst. 38 (4) (2021) 6551–6558.

[10] S. Yang, S. Nagaraja, W. Sun, V. Yang, Multiscale modeling and general theory of non-equilibrium plasma-assisted ignition and combustion, J. Phys. D Appl. Phys. 50 (43) (2017) 433001.

[11] T.A. Casey, J. Han, M. Belhi, P.G. Arias, F. Bisetti, H.G. Im, J.-Y. Chen, Simulations of planar non-thermal plasma assisted ignition at atmospheric pressure, P. Combust. Inst. 36 (3) (2017) 4155–4163.

[12] M.S. Bak, H. Do, M.G. Mungal, M.A. Cappelli, Plasma-assisted stabilization of laminar premixed methane/air flames around the lean flammability limit, Combust. Flame 159 (10) (2012) 3128–3137.

[13] D. Breden, L.L. Raja, C.A. Idicheria, P.M. Najt, S. Mahadevan, A numerical study of high-pressure non-equilibrium streamers for combustion ignition application, J. Appl. Phys. 114 (8) (2013) 083302.

[14] Y. Bechane, B. Fiorina, Numerical investigations of turbulent premixed flame ignition by a series of nanosecond repetitively pulsed discharges, P. Combust. Inst. 38 (4) (2021) 6575–6582.

[15] P. Pepiot, Automatic strategies to model transportation fuel surrogates, Stanford University, 2008 Ph.D. thesis.

[16] Q. Cazéres, P. Pepiot, E. Riber, B. Cuenot, A fully automatic procedure for the analytical reduction of chemical kinetics mechanisms for computational fluid dynamics applications, Fuel 303 (2021) 121247.

[17] A. Bellemans, N. Kincaid, N. Deak, P. Pepiot, F. Bisetti, P-DRGEP: A novel methodology for the reduction of kinetics mechanisms for plasma-assisted combustion applications, P. Combust. Inst. 38 (4) (2021) 6631–6639.

[18] N. Popov, Fast gas heating in a nitrogen–oxygen discharge plasma: i. kinetic mechanism, J. Phys. D Appl. Phys., Volume 44, Issue 28 (2011).

[19] V. Guerra, A.T. del Caz, C.D. Pintassilgo, L.L. Alves, Modelling n_2-o_2 plasmas: volume and surface kinetics, Plasma Sources Sci. T. 28 (7) (2019) 073001.

[20] N. Popov, Fast gas heating initiated by pulsed nanosecond discharge in atmospheric pressure air, Proc. AIAA (2013).

[21] D. Rusterholtz, D. Lacoste, G. Stancu, D. Pai, C. Laux, Ultrafast heating and oxygen dissociation in atmospheric pressure air by nanosecond repetitively pulsed discharges, J. Phys. D Appl. Phys. 46 (2013) 4010.

[22] M. Capitelli, F. C.M, B. Gordiets, A. Osipov, Plasma kinetics in atmospheric gases, Springer, Berlin, 2000.

[23] I.A. Kossyi, A.Y. Kostinsky, A.A. Matveyev, V.P. Silakov, Kinetic scheme of the non-equilibrium discharge in nitrogen-oxygen mixtures, Plasma Sources Sci. T. 1 (3) (1992) 207–220.

[24] Y. Ju, W. Sun, Plasma assisted combustion: dynamics and chemistry, Prog. Energ. Combust 48 (2015) 21–83.

[25] G.J.M. Hagelaar, L.C. Pitchford, Solving the boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models, Plasma Sources Sci. T. 14 (4) (2005) 722–733.

[26] I. Shkurenkov, D. Burnette, W.R. Lempert, I.V. Adamovich, Kinetics of excited states and radicals in a nanosecond pulse discharge and afterglow in nitrogen and air, Plasma Sources Sci. T. 23 (6) (2014) 065003.

[27] A. Montello, Z. Yin, D. Burnette, I.V. Adamovich, W.R. Lempert, Picosecond CARS measurements of nitrogen vibrational loading and rotational/translational temperature in non-equilibrium discharges, J. Phys. D Appl. Phys. 46 (46) (2013) 464002.

[28] M. Uddi, N. Jiang, E. Mintusov, I. Adamovich, W. Lempert, Atomic oxygen measurements in air and air/fuel nanosecond pulse discharges by two photon laser induced flourescence, 46th AIAA Aerospace Sciences Meeting and Exhibit, American Institute of Aeronautics and Astronautics, 2008.

[29] S. Starikovskaia, E. Kukaev, A. Kuksin, M. Nudnova, A. Starikovskii, Analysis of the spatial uniformity of the combustion of a gaseous mixture initiated by a nanosecond discharge, Combust. Flame 139 (3) (2004) 177–187.

[30] C. Winters, Z. Eckert, Z. Yin, K. Frederickson, I.V. Adamovich, Measurements and kinetic modeling of atomic species in fuel-oxidizer mixtures excited by a repetitive nanosecond pulse discharge, J. Phys. D Appl. Phys. 51 (1) (2017) 015202.

[31] D.G. Goodwin, R.L. Speth, H.K. Moffat, B.W. Weber, Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes, 2021, Version 2.5.1, https://www.cantera.org. 10.5281/zenodo.4527812

[32] A.T. del Caz, V. Guerra, D. Goncalves, M.L. da Silva, L. Marques, N. Pinhao, C.D. Pintassilgo, L.L. Alves, The lisbon KInetics boltzmann solver, Plasma Sources Sci. T. 28 (4) (2019) 043001.

[33] G.P. Smith, D.M. Golden, M. Frenklach, N.W. Moriarty, B. Eiteneer, M. Goldenberg, C.T. Bowman, R.K. Hanson, S. Song, W.C. Gardiner, J.V.V. Lissianski, Z. Qin, http://www.me.berkeley.edu/gri_mech/.

[34] M.L. Lavadera, C. Brackmann, A.A. Konnov, Experimental and modeling study of laminar burning velocities and nitric oxide formation in premixed ethylene/air flames, P. Combust. Inst. 38 (1) (2021) 395–404.

[35] N.L. Aleksandrov, S.V. Kindysheva, E.N. Kukaev, S.M. Starikovskaya, A.Y. Starikovskii, Simulation of the ignition of a methane-air mixture by a high-voltage nanosecond discharge, Plasma Phys. Rep. 35 (10) (2009) 867–882.

[36] N.A. Popov, Investigation of the mechanism for rapid heating of nitrogen and air in gas discharges, Plasma Phys. Rep. 27 (10) (2001) 886–896.

[37] N.A. Popov, Pulsed nanosecond discharge in air at high specific deposited energy: fast gas heating and active particle production, Plasma Sources Sci. T. 25 (4) (2016) 044003.

[38] L.G. Piper, Quenching rate coefficients for n_2(a'1), J. Chem. Physics 87 (3) (1987) 1625–1629.

[39] W. Marinelli, W. Kessler, B. Green, W. Blumberg, Quenching of $n_2(a^1\pi_g, v' = o)$ by n_2, o_2, CO, CO_2, CH_4, h_2 and ar, J. Chem. Physics 90 (1989) 9.

[40] Lisbon, (IST-LISBON database, retrieved on October 19, 2020). www.lxcat.net/IST-Lisbon.

[41] Phelps, (PHELPS database, retrieved on July 30, 2020). www.lxcat.net/Phelps.

[42] W.M. Huo, T.L. Gibson, M.A.P. Lima, V. McKoy, Schwinger multichannel study of the $^2\Pi_g$ shape resonance in n_2, Phys. Rev. A 36 (1987) 1632–1641.

[43] R.C. Millikan, D.R. White, Systematics of vibrational relaxation, J. Chem. Phys. 39 (12) (1963) 3209–3213.

[44] A.A. Konnov, On the role of excited species in hydrogen combustion, Combust. Flame 162 (10) (2015) 3755–3772.

[45] D.Z. Pai, G.D. Stancu, D.A. Lacoste, C.O. Laux, Nanosecond repetitively pulsed discharges in air at atmospheric pressure - the glow regime, Plasma Sources Sci. T. 18 (4) (2009) 045030.

[46] D.Z. Pai, D.A. Lacoste, C.O. Laux, Nanosecond repetitively pulsed discharges in air at atmospheric pressure - the spark regime, Plasma Sources Sci. T. 19 (6) (2010) 065015.

[47] A.C. Tibére-Inglesse, S.D. McGuire, C.O. Laux, Inferring the gas temperature from the rotational temperatures of n_2(b) and n_2(c), To be submitted.

[48] A.A. Kulikovsky, Positive streamer between parallel plate electrodes in atmospheric pressure air, J. Phys. D Appl. Phys. 30 (3) (1997) 441–450.

[49] G.V. Naidis, Modelling of transient plasma discharges in atmospheric-pressure methane-air mixtures, J. Phys. D Appl. Phys. 40 (15) (2007) 4525–4531.

[50] S. Kobayashi, Z. Bonaventura, F. Tholin, N.A. Popov, A. Bourdon, Study of nanosecond discharges in h_2-air mixtures at atmospheric pressure for plasma assisted combustion applications, Plasma Sources Sci. T. 26 (7) (2017) 075004.

[51] J.K. Lefkowitz, T. Ombrello, An exploration of inter-pulse coupling in nanosecond pulsed high frequency discharge ignition, Combust. Flame 180 (2017) 136–147.

[52] M. Uddi, Non-equilibrium kinetic studies of repetitively pulsed nanosecond discharge plasma assisted combustion, 2008 Ph.D. thesis. The Ohio State University

[53] Triniti, (TRINITI database, retrieved on September 14, 2020). www.lxcat.net/Triniti.

[54] Hayashi, (HAYASHI database, retrieved on September 14, 2020). www.lxcat.net/Hayashi.

[55] Morgan, (MORGAN database, retrieved on October 6, 2020). www.lxcat.net/Morgan.

[56] E.I. Mintoussov, S.J. Pendleton, F.G. Gerbault, N.A. Popov, S.M. Starikovskaia, Fast gas heating in nitrogen-oxygen discharge plasma: II. energy exchange in the afterglow of a volume nanosecond discharge at moderate pressures, J. Phys. D Appl. Phys. 44 (28) (2011) 285202.

[57] M. Hayashi, Electron collision cross-sections for molecules determined from beam and swarm data, Swarm Studies and Inelastic Electron-Molecule Collisions, Springer (1987).

[58] S. Heijkers, M. Aghaei, A. Bogaerts, Plasma-based CH_4 conversion into higher hydrocarbons and h_2: modeling to reveal the reaction mechanisms of different plasma sources, J. Phys. Chem. C 124 (13) (2020) 7016–7030.

[59] Z. Eckert, Energy Transfer in Non-Equilibrium Reacting Gas Flows: Applications in Plasma Assisted Combustion and Chemical Gas Lasers, 2018 Ph.D. thesis. The Ohio State University

[60] 2022. AVIP: a low temperature plasma code.

[61] Thilo Schonfeld, Michael Rudgyard, Steady and Unsteady Flow Simulations Using the Hybrid Flow Solver AVBP, AIAA Journal (1999) In this issue, doi:10.2514/2.636.

9.2 Supplementary material around the EEDF

The computation of the EEDF plays a central role in non-thermal discharges. In CANTERA, an open-source implementation of BOLSIG+, called BOLOS written by Alejandro Luque in Python, is used. Some features, such as the Coulomb collisions and spatial growth model have been added in our own Python version (`https://github.com/lionelchg/bolos`). The implementation of the BOLOS algorithm in C++ inside CANTERA has been made by another PhD student, Bang-Shiuh Chen, (see `https://github.com/Cantera/cantera/pull/700`) and we reported those back into the CERFACS version of CANTERA.

9.2.1 Discretization

Some details about the discretization and solving procedure used in BOLOS are given here. The energy space is discretized on a grid in energy space, consisting of n cells of energy ε_i with boundaries $\varepsilon_{i-1/2}$ and $\varepsilon_{i+1/2}$. Integrating the equation for the EEDF Eq. (4.26) yields:

$$\left[\tilde{W}F_0 - \tilde{D}\frac{\partial F_0}{\partial \varepsilon}\right]_{i+1/2} - \left[\tilde{W}F_0 - \tilde{D}\frac{\partial F_0}{\partial \varepsilon}\right]_{i-1/2} = \int_{\varepsilon_{i-1/2}}^{\varepsilon_{i+1/2}} \tilde{S}d\varepsilon + \int_{\varepsilon_{i-1/2}}^{\varepsilon_{i+1/2}} \tilde{R}d\varepsilon \tag{9.1}$$

where

$$\begin{aligned}
\tilde{W} &= -\gamma\varepsilon^2\sigma_\varepsilon - 3a\frac{n}{N}A_1 \\
\tilde{D} &= \frac{\gamma}{3}\left(\frac{E}{N}\right)^2\frac{\varepsilon}{\tilde{\sigma}_\mathrm{m}} + \frac{\gamma k_\mathrm{B}T}{e}\varepsilon^2\sigma_\varepsilon + 2a\frac{n}{N}\left(A_2 + \varepsilon^{3/2}A_3\right) \\
\sigma_\varepsilon &= \sum_{k=\text{elastic}}\frac{2m}{M_k}x_k\sigma_k \\
\tilde{S} &= \sum_{k=\text{inelastic}}\tilde{C}_{0,k}
\end{aligned}$$

9.2.1.1 Convective-diffusive flux

A Scharfetter-Gummel scheme, detailed in Section. (7.2.2) and already used for advection diffusion, is used in energy space here to compute the convective-diffusive flux:

$$\left[\tilde{W}F_0 - \tilde{D}\frac{\partial F_0}{\partial \varepsilon}\right]_{i+1/2} = \frac{\tilde{W}_{i+1/2}F_{0,i}}{1-\exp\left[-z_{i+1/2}\right]} + \frac{\tilde{W}_{i+1/2}F_{0,i+1}}{1-\exp\left[z_{i+1/2}\right]} \tag{9.2}$$

Defining the coefficients b_i and c_i for compacity:

$$b_i = \frac{\tilde{W}_{i+1/2}}{1-\exp\left[-z_{i+1/2}\right]} \tag{9.3}$$

$$c_i = \frac{\tilde{W}_{i+1/2}}{1-\exp\left[z_{i+1/2}\right]} \tag{9.4}$$

the left hand side of Eq. 9.1 yields

$$\left[\tilde{W}F_0 - \tilde{D}\frac{\partial F_0}{\partial \varepsilon}\right]_{i-1/2}^{i+1/2} = -b_{i-1}F_{0,i-1} + (b_i - c_{i-1})F_{0,i} + c_i F_{0,i+1} \quad \text{for } i \in [\![2, n-1]\!] \tag{9.5}$$

We suppose that there is no flux in energy space at zero energy and a zero flux condition can be applied at the other boundary. The matrix $SG(F_0)$ representing this flux is thus tri-diagonal:

$$\begin{pmatrix} b_1 & c_1 & & & \\ -b_1 & b_2 - c_1 & c_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & c_{n-1} \\ & & & -b_{n-1} & -c_{n-1} \end{pmatrix} \tag{9.6}$$

9.2.1.2 Inelastic collisions

The inelastic collision terms are non-local in energy but linear in F_0. The matrix associated with these inelastic terms is denoted $Q(F_0)$. We discretize the cell ε_i as follows:

$$\int_{\varepsilon_{i-1/2}}^{\varepsilon_{i+1/2}} \tilde{S}\mathrm{d}\varepsilon \equiv -P_i F_{0,i} + \sum_j Q_{i,j} F_{0,j} \tag{9.7}$$

where x_k is the molar fraction associated to target k and where the two terms represent scattering-in and scattering-out terms:

$$P_i = \sum_{\text{inelastic}} \gamma x_k \int_{\varepsilon_{i-1/2}}^{\varepsilon_{i+1/2}} \varepsilon\sigma_k \exp\left[(\varepsilon_i - \varepsilon)\, g_i\right] \mathrm{d}\varepsilon \tag{9.8}$$

$$Q_{i,j} = \sum_{\text{inelastic}} \gamma x_k \int_{\varepsilon_1}^{\varepsilon_2} \varepsilon\sigma_k \exp\left[(\varepsilon_i - \varepsilon)\, g_i\right] \mathrm{d}\varepsilon \tag{9.9}$$

and where the interval $[\varepsilon_1, \varepsilon_2]$ is the overlap of cell j and cell i shifted by the threshold energy u_k associated to the collisional process:

$$\varepsilon_1 = \min\left(\max\left(\varepsilon_{i-1/2} + u_k, \varepsilon_{j-1/2}\right), \varepsilon_{j+1/2}\right) \tag{9.10}$$

$$\varepsilon_2 = \min\left(\max\left(\varepsilon_{i+1/2} + u_k, \varepsilon_{j-1/2}\right), \varepsilon_{j+1/2}\right) \tag{9.11}$$

Finally we assume the distribution F_0 to be piecewise exponential (as is the case with the Gaussian distribution) so that a local logarithmic slope can be used to evaluate g_i:

$$g_i = \frac{1}{\varepsilon_{i+1} - \varepsilon_{i-1}} \ln\left(\frac{F_{0,i+1}}{F_{0,i-1}}\right) \tag{9.12}$$

9.2.1.3 Incorporation of the growth term

The EEDF satisfies the following non-linear system where the matrix A takes different values depending on the growth model:

$$A(F_0)F_0 = Q(F_0)F_0 \tag{9.13}$$

In all cases the matrix $A(F_0)$ can be written as a sum of the SG matrix and a term coming from the growth model used:

$$A(F_0) = SG(F_0) - \mathrm{diag}(G_i) \tag{9.14}$$

where G_i is a vector in energy space that incorporates growth effects.

- In the case of no growth model $G_i = 0$ and we simply have $A(F_0) = SG(F_0)$.
- For the temporal growth model:

$$\tilde{R} = -\frac{\bar{\nu}_i}{N}\varepsilon^{1/2}F_0 \tag{9.15}$$

  Assuming a constant value of the EEDF within the cell:

$$\int_{\varepsilon_{i-1/2}}^{\varepsilon_{i+1/2}} \tilde{R}d\varepsilon = -\frac{\bar{\nu}_i}{N}\left[\frac{2}{3}\varepsilon^{3/2}\right]_{i-1/2}^{i+1/2} F_{0,i} = G_i F_{0,i} \tag{9.16}$$

- For the spatial growth model we rewrite the growth renormalization term:

$$\begin{aligned}
\tilde{R} &= \frac{\alpha}{N}\frac{\gamma}{3}\left[\frac{\varepsilon}{\sigma_\mathrm{m}}\left(2\frac{E}{N}\frac{\partial F_0}{\partial\varepsilon} + \frac{\alpha}{N}F_0\right) + \frac{E}{N}F_0\frac{\partial}{\partial\varepsilon}\left(\frac{\varepsilon}{\sigma_\mathrm{m}}\right)\right] \\
\implies \tilde{R} &= \frac{\alpha}{N}\frac{\gamma}{3}\left[2\frac{E}{N}\frac{\partial}{\partial\varepsilon}\left(\frac{\varepsilon}{\sigma_\mathrm{m}}F_0\right) + \frac{\varepsilon}{\sigma_\mathrm{m}}\frac{\alpha}{N}F_0 - \frac{E}{N}F_0\frac{\partial}{\partial\varepsilon}\left(\frac{\varepsilon}{\sigma_\mathrm{m}}\right)\right] \qquad (9.17) \\
\implies \tilde{R} &= \frac{\gamma}{3}\left[\frac{\partial}{\partial\varepsilon}\left(2\frac{\alpha}{N}\frac{E}{N}\frac{\varepsilon}{\sigma_\mathrm{m}}F_0\right)\right] + \frac{\alpha}{N}\frac{\gamma}{3}\left[\frac{\varepsilon}{\sigma_\mathrm{m}}\frac{\alpha}{N}F_0 - \frac{E}{N}F_0\frac{\partial}{\partial\varepsilon}\left(\frac{\varepsilon}{\sigma_\mathrm{m}}\right)\right] \qquad (9.18)
\end{aligned}$$

  The first term in the right hand side of $\tilde{R}$ is added to the convective part of the Scharfetter-Gummel flux. The second part which we will name $\tilde{R}_2$ is integrated as follows:

$$\int_{i-1/2}^{i+1/2} \tilde{R}_2 d\varepsilon = \frac{\alpha}{N}\frac{\gamma}{3}\left(\left[\frac{\varepsilon}{2}\right]_{i-1/2}^{i+1/2}\frac{\alpha}{\sigma_\mathrm{m}N} - \frac{E}{N}\left[\frac{\varepsilon}{\sigma_\mathrm{m}}\right]_{i-1/2}^{i+1/2}\right)F_{0,i} = G_i F_{0,i} \tag{9.19}$$

9.2.2 Iteration procedure

Now that the non-linear system of the EEDF is derived we need to find the solution F_0 of this non-linear system. We start from a Maxwell distribution at a certain mean energy:

$$F_0^0 = \frac{2}{\sqrt{\pi}}\frac{1}{(kT)^{3/2}}\exp\left(-\frac{\varepsilon}{kT}\right). \tag{9.20}$$

Convergence to the steady state is achieved adding a fictious time derivative:

$$\frac{\partial F_0}{\partial t} + A(F_0)F_0 = Q(F_0)F_0. \tag{9.21}$$

We introduce a fictive time step Δ and iterations are performed the following way from iteration n to iteration $n+1$:

$$\begin{aligned} & F_0^{n+1} - F_0^n + A(F_0^n)F_0^{n+1}\mathrm{d} = Q(F_0^n)F_0^{n+1}\mathrm{d} \\ \Longrightarrow\ & F_0^{n+1}(I + A(F_0^n)\mathrm{d} - Q(F_0^n)\mathrm{d}) = F_0^n \end{aligned} \tag{9.22}$$

We have here a quasi linear system to solve where the equation is implicit in F_0^{n+1}. Convergence is achieved when the L_1 norm of the error goes below a certain threshold:

$$L_1(|F_0^{n+1} - F_0^n|) = \int |F_0^{n+1} - F_0^n|\varepsilon^{1/2}d\varepsilon \tag{9.23}$$

As done in LokiB [Tejero-del Caz et al., 2019], the grid is adapted after each pass of the iteration procedure to have a certain number of decades in the EEDF. The actual parameters that are used at $D_{\min} = 15$ and $D_{\max} = 25$. To ensure this number of decades at the end of an iteration procedure with a certain relative error E_{rel}:

$$\varepsilon_{\max} = \begin{cases} \varepsilon_{\max}/(1+f) & \text{if} \quad D \geq D_{\max} \\ (1+f)\varepsilon_{\max} & \text{otherwise} \quad . \end{cases} \tag{9.24}$$

9.2.3 Examples of EEDF

Examples of EEDF in air and CH_4-air mixtures are now given. The first step is to select a complete set of cross-sections for the studied mixture. The LXCat database (`https://nl.lxcat.net/home/`) contains complete sets of cross-sections from a variety of sources for a large range of neutral targets. The reader is referred to Table. 1 of the article for the chosen databases for each species.

9.2.3.1 EEDF in air

Nitrogen plays a critical role in the energy transfer from the electron to the gas. The set of cross-sections from the Phelps database [Phelps], used in the article, is shown in Fig. 9.1. The three types of excitation already mentioned in Section. 4.1.2 are observed: rotational excitation (one cross-section), vibrational excitation (all the N2(v) cross sections) and electronic excitation (N_2(A, B, C) and so on). Rotational excitation does not showcase any threshold effect whereas an energy activation, below which the cross-section is zero, exist for vibrational and electronic excitation. Note that the energy activation of electronic excitation is higher than vibrational excitation.

O_2 cross-sections are similar to that of N_2 (not shown) and the resulting EEDF F_0 for air composition is shown in Fig. 9.2 for increasing reduced electric field E/N from 0.1 to 1000 Td. As the reduced electric field is increased, more and more energetic electrons are produced so that the tail of the EEDF increases while the head decreases. This leads to the activation of electronic and even ionization reactions as the tail of the EEDF is populated.

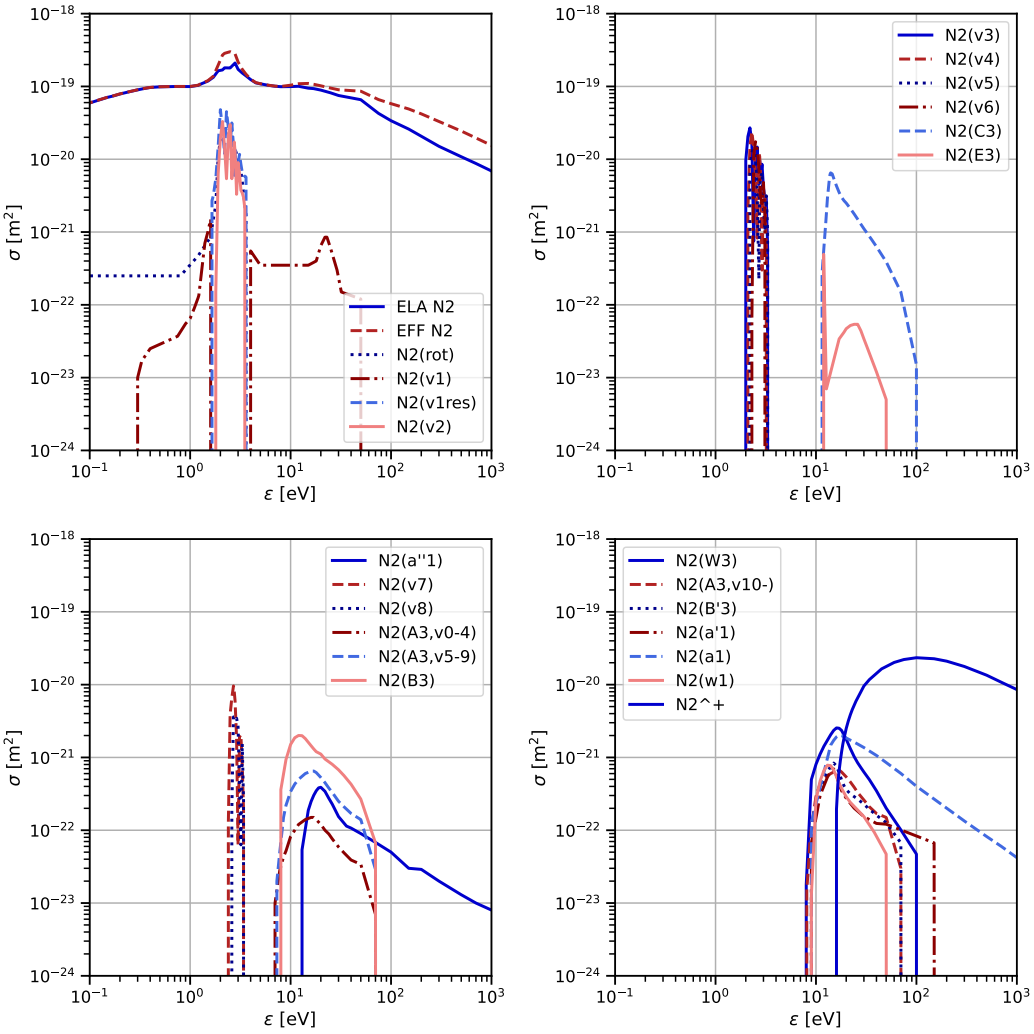


Figure 9.1: N_2 cross-sections from the Phelps database.

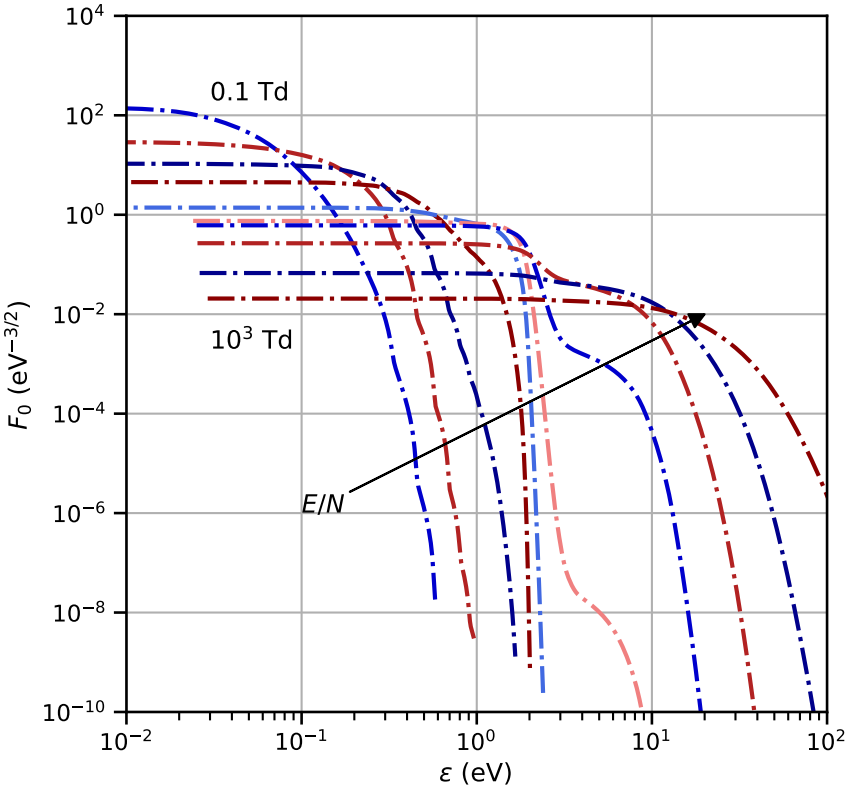


Figure 9.2: EEDF for air composition at varying reduced electric field E/N from 0.1 to 1000 Td.

The different ratios of electron energy going to elastic, rotational, vibrational, electronic and ionization channels are shown in Fig. 9.3. Below 4 Td, vibration of O_2 is dominant whereas at medium values vibration of N_2 is dominant up to around 100 Td. At very high values, as expected electronic excitation becomes the main energy deposition channel. Hence depending on the value of the reduced electric field, the interaction of the plasma discharge with the gas will be different: at intermediate values vibration of N_2 is excited causing a slow relaxation of energy whereas N_2 electronically excited states cause a faster gas heating and dissociation.

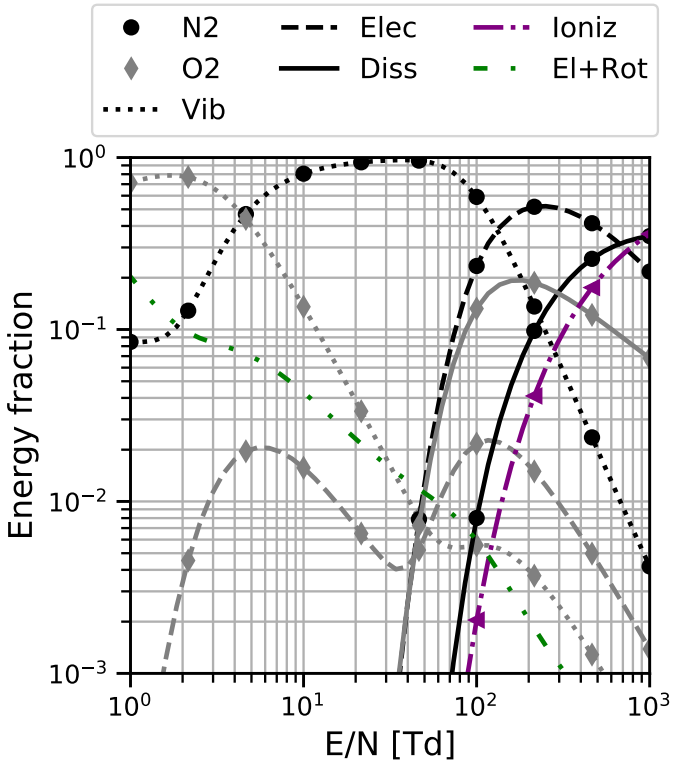


Figure 9.3: Energy channels from the EEDF in air composition at varying reduced electric field E/N from 0.1 to 1000 Td.

9.2.3.2 EEDF in methane-air

To include methane, a corresponding set of cross-sections must be added on top of the N_2-O_2 cross sections and the Hayashi database has been chosen. A set of six collisions can be found: an elastic collision, an attachment collision, three excitation collisions and one ionization collision shown in Fig. 9.4.

The energy channels associated to a stochiometric air-methane mixture are shown in Fig. 9.5. The vibration of CH_4 seems to be significant as it becomes the main energy deposit at very low reduced electric field. However at intermediate and high electric fields N_2 vibration and excitation are respectively still the main channel of energy deposition from the discharge.

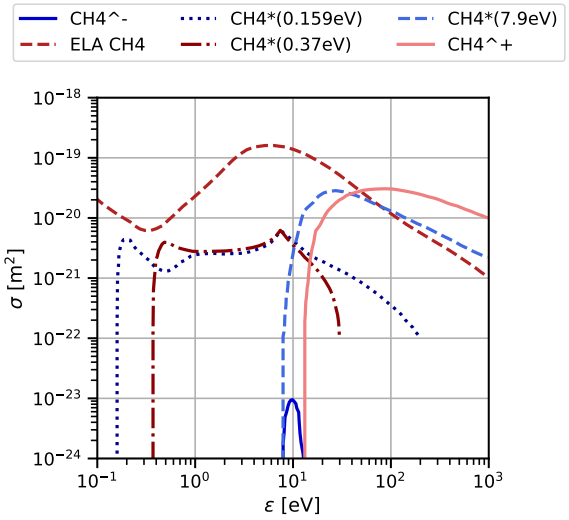


Figure 9.4: CH_4 cross-sections from the Hayashi database.

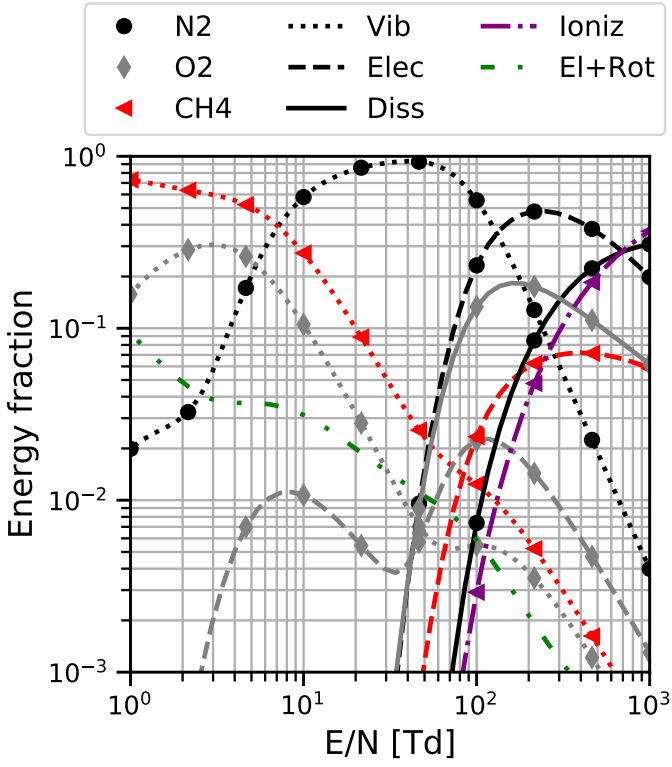


Figure 9.5: Energy channels from the EEDF in stochiometric air-methane composition at varying reduced electric field E/N from 0.1 to 1000 Td.

9.3 Conclusion

A detailed plasma assisted combustion chemistry has been derived with multiple experimental validations using zero-dimensional reactors in CANTERA. Fast gas heating, vibrational slow gas heating and radicals production are all covered by the derived chemistry.

The size of the detailed mechanism, impractical for multi-dimensional simulations, has been reduced using ARCANE yielding mechanisms of reasonable size for the upcoming multi-dimensional simulations.

Chapter 10

Streamer Simulations

Contents

In this chapter plasma discharges are simulated in axisymmetric conditions using drift-diffusion modeling of plasma species. Validation of the streamer propagation capabilities of AVIP is first performed by comparing the code results to a streamer code benchmark done in Bagheri et al. [2018] using the two schemes presented in Chap. 7: the limited Lax-Wendroff scheme (LLW) and the improved Scharfetter Gummel scheme (ISG). Streamer propagation simulations are then carried out in pin-pin configurations with hyperbole electrodes using a simple three-species chemistry representative of air. Next the methodology to simulate complete NRP discharges is presented where three phases are distinguished. Finally a more realistic chemistry derived from the previous 0D study is used to simulate a pin-pin configuration with delta electrodes.

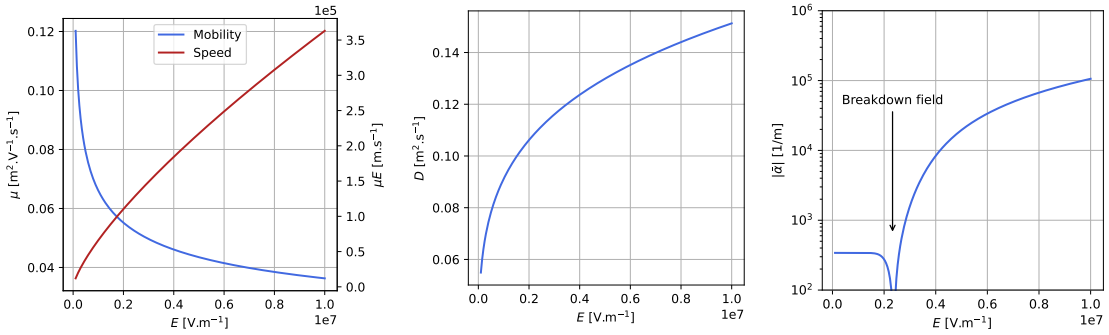
10.1 Bagheri benchmark

AVIP is compared to the six streamer codes benchmark Bagheri et al. [2018]. The geometry is the same across all test cases: 2D axisymmetric square of 1.25×1.25 cm^2. Electrons e and one species of positive ions p are considered so that the governing equations are

$$\nabla^2 \phi = -\frac{e(n_p - n_e)}{\varepsilon_0} \longrightarrow \mathbf{E} = -\nabla\phi \tag{10.1}$$

$$\frac{\partial n_e}{\partial t} + \nabla \cdot (-n_e \mu_e \mathbf{E} - D_e \nabla n_e) = \bar{\alpha} \mu_e ||\mathbf{E}|| n_e + S_{ph} \tag{10.2}$$

$$\frac{\partial n_p}{\partial t} = \bar{\alpha} \mu_e ||\mathbf{E}|| n_e + S_{ph} \tag{10.3}$$



(a) Mobility and speed (b) Diffusion coefficient (c) Effective ionization

Figure 10.1: Transport and chemistry coefficients as functions of the electric field norm for the Bagheri et al. [2018] validation case.

where μ_e is the electron mobility, D_e the electron diffusion coefficient, $\bar{\alpha}$ the effective ionization coefficient (ionization α minus attachment η), $\mathbf{E}$ the electric field and S_{ph} the photoionization source term. The chemistry is composed of three reactions where we denote the neutral A, the positive ion A^+ and the negative ion A^- (not described):

$$\text{Ionization} \quad e^- + A \longrightarrow 2\,e^- + A^+ \tag{10.4}$$

$$\text{Attachment} \quad e^- + A \longrightarrow A^- \tag{10.5}$$

$$\text{Photoionization} \quad A + \gamma \longrightarrow e^- + A^+ \tag{10.6}$$

We consider streamer discharges in dry air at $p = 1\,\text{bar}$ and $T = 300\,\text{K}$ for which the coefficients are tabulated as a function of of the reduced electric field (local field approximation) from Bagheri et al. [2018]. The coefficients for the chosen conditions are plotted in Fig. 10.1. These coefficients are only function of the electric field since the neutral density has been set for $p = 1\,\text{bar}$ and $T = 300\,\text{K}$. Mobility decreases and diffusion increases as the electric field increases due to more collisions. Finally the effective ionization as a function of the electric field allows to retrieve the breakdown field E_b. This is a critical parameter as it controls the onset of the electron avalanche process described in Chap. 2: above E_b discharges can multiply exponentially and a discharge can propagate. The value of the breakdown field is around $2.2 \times 10^6\,\text{MV m}^{-1}$ for this chemistry.

The positive streamer is initialized with a Gaussian profile on the positive species density only so that a strong charge density triggers the positive streamer density. A common background for both electrons and positive ions is also set and the densities are initialized as

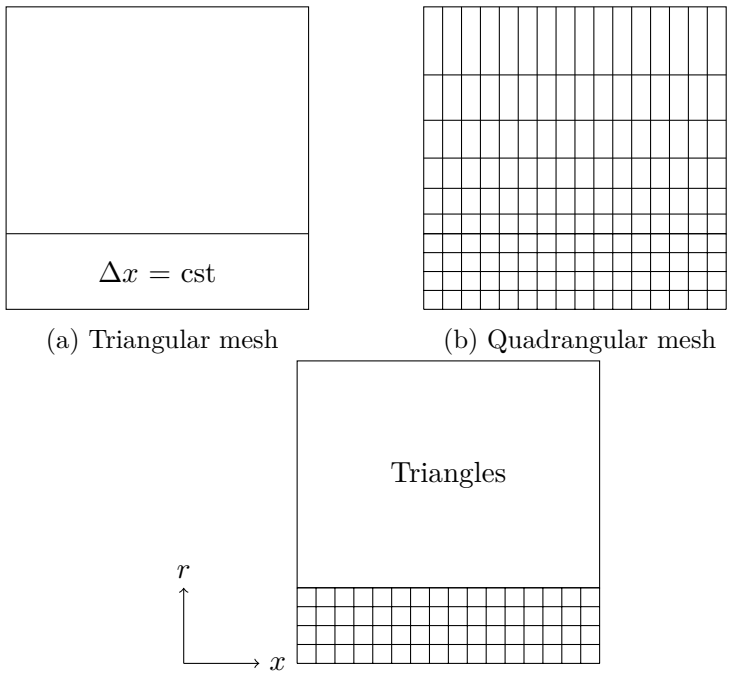
$$n_i(x, r) = n_0 \exp\left(-\frac{r^2 + (x - x_0)^2}{\sigma^2}\right) + n_{\text{back}} \tag{10.7}$$

$$n_e(x, r) = n_{\text{back}} \tag{10.8}$$

where $n_0 = 5 \times 10^{18}\ \text{m}^{-3}$ and $x_0 = 10^{-2}\ \text{m}$ are taken for all cases.

10.1.1 Meshes

Triangular and quadrangular elements are considered for the validation of the schemes since AVIP can work on both types of elements. Three kind of meshes are considered: triangular meshes, quadrangular meshes and finally hybrid meshes which are shown in Fig. 10.2. The refinement zone is a rectangle at the bottom of the computational domain around the axis $r = 0$. In triangular meshes, only the limited LW scheme can be used since the improved SG scheme only works in topologically dual meshes which is not ensured easily for triangular meshes. In quadrangular meshes, both schemes can be applied and finally in hybrid meshes the improved SG scheme can be applied in quadrangles whereas the limited Lax-Wendroff scheme is used in triangles.



(a) Triangular mesh

(b) Quadrangular mesh

(c) Hybrid mesh

Figure 10.2: Meshes for Bagheri et al. [2018] test case.

Different mesh sizes are considered for each type of elements and are summarized in Tab. 10.1. Note that the typical mesh size used in the benchmark is 3 µm.

10.1.2 Test cases

Three test cases are considered in this benchmark to assess different aspects of streamer propagation. A constant background electric field is set to $1.5\,\mathrm{MV\,m^{-1}}$ in the x direction: this value is below the breakdown electric field of $2.2\,\mathrm{MV\,m^{-1}}$ so that it is not enough to propagate a streamer by itself: the initial gaussian seed of charge density Eq. (10.7) imposed at the beginning allows to locally increase the electric field *above the breakdown field* so that a streamer can propagate. The three cases have two varying parameters: background charged

Name	Type	Δx_{min}	N_{nodes}	N_{cells}
`tri5micro`	Tri	5 µm	4.2×10^5	8.4×10^5
`tri3micro`	Tri	3 µm	1×10^6	2.1×10^6
`tri2micro`	Tri	2 µm	2.33×10^6	4.66×10^6
`quad5micro`	Quad	5 µm	7.5×10^5	7.5×10^5
`quad3micro`	Quad	3 µm	1.8×10^6	1.8×10^6
`quad2micro`	Quad	2 µm	3.8×10^6	3.8×10^6
`hybrid5micro`	Hybrid	5 µm	6.5×10^5	7.1×10^5
`hybrid25micro`	Hybrid	2.5 µm	2.5×10^6	2.6×10^6

Table 10.1: Summary of the meshes used for the benchmark

species density n_{back} and photoionization which are summarized in Tab. 10.2. Each test case is now going to be described along with its results.

Casename	n_{back} [m^{-3}]	Photoionization
Case 1	10^{13}	No
Case 2	10^9	No
Case 3	10^9	Yes

Table 10.2: Summary of the three cases parameters

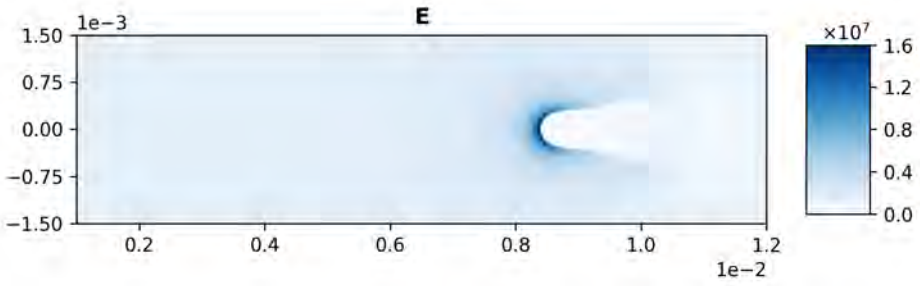
10.1.2.1 Case 1 results

The first case has a rather strong background density at $1 \times 10^{13}\,\text{m}^{-3}$ and no photoionization: it is a canonical streamer propagation where photoionization has been replaced by a strong background density.

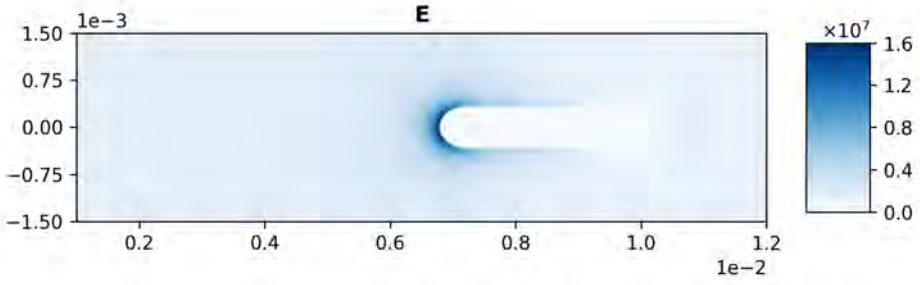
The electron density and electric field for the ISG scheme is shown in Fig. 10.3 for the full quadrangular mesh at $\Delta x = 3\,\mu\text{m}$. The propagation of the positive streamer from right to left is observed and the norm electric field at the head is around $15\,\text{MV}\,\text{m}^{-1}$ which is above the breakdown field of the chemistry, causing the electron avalanche and the streamer propagation. Profiles of electron density and electric field norm along the axis are shown in Fig. 10.4 where the propagation of the streamer head is observed: the peak of electric field moves from right to left and causes ionization leaving behind a higher electron density.

On the other hand the LLW scheme at the same resolution of $\Delta x = 3\,\mu\text{m}$ using a triangular mesh produces oscillations. A comparison of the electron density at $t = 16\,\text{ns}$ in Fig. 10.5 shows that the LLW scheme (Fig. 10.5b) using a triangular mesh produces oscillations. These oscillations, present in both the electron density and electric field profiles, indicate that the streamer was on the verge of branching. By increasing the resolution to $\Delta = 2\,\mu\text{m}$ (Fig. 10.5c), these oscillations disappear and hence trianglular elements are less stable than quadrangular elements.

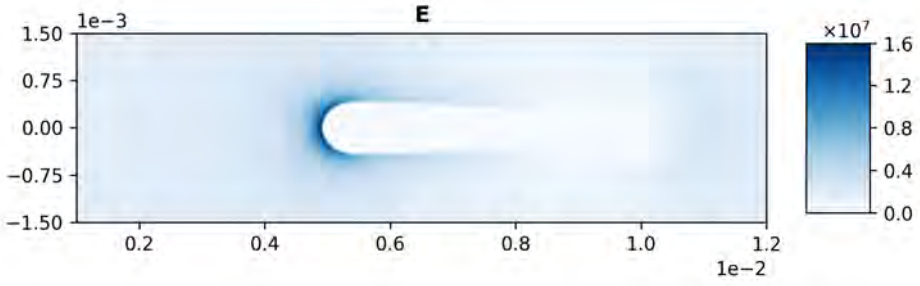
A comparison of those three runs with the six codes of the benchmark is shown in Fig. 10.6 for the streamer length which is defined as the distance between the initial gaussian seed and the maximum of the norm of the electric field. We can see that all three runs are satisfactory and lie between all the codes of the benchmark. However when looking at the value of the



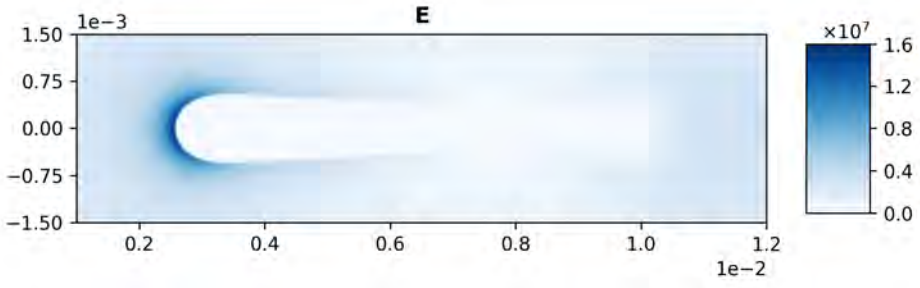
(a) 4 ns



(b) 8 ns



(c) 12 ns



(d) 16 ns

Figure 10.3: Electric field [V/m] of the streamer at different instants for ISG at $\Delta x = 3\,\mu\text{m}$.

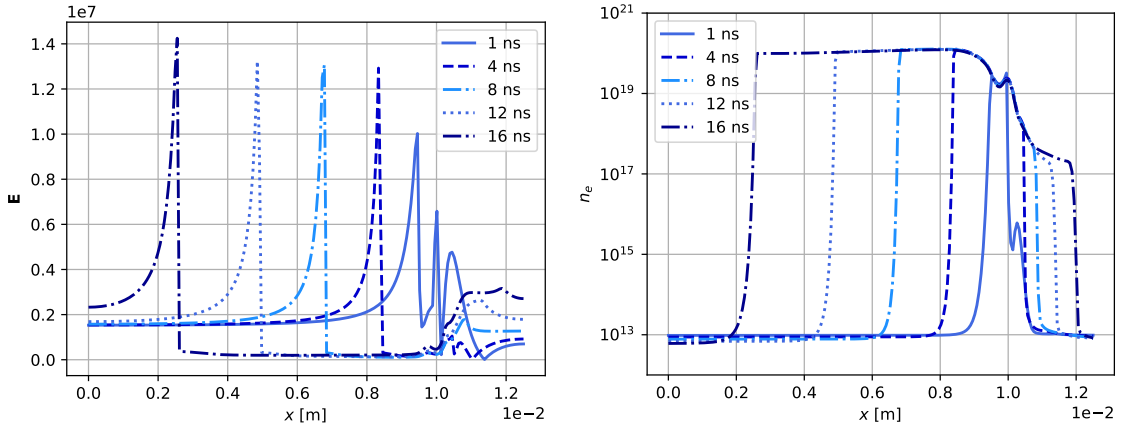
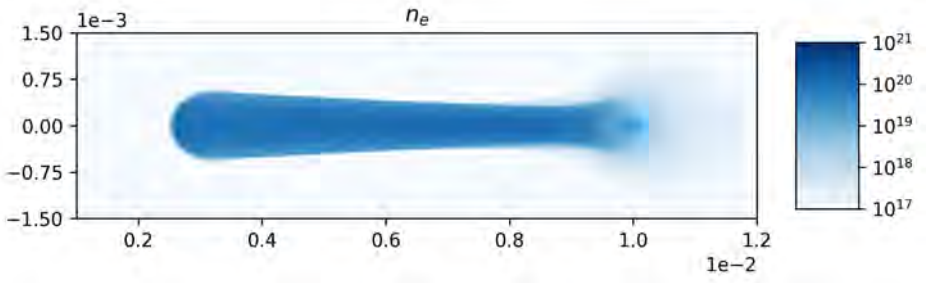
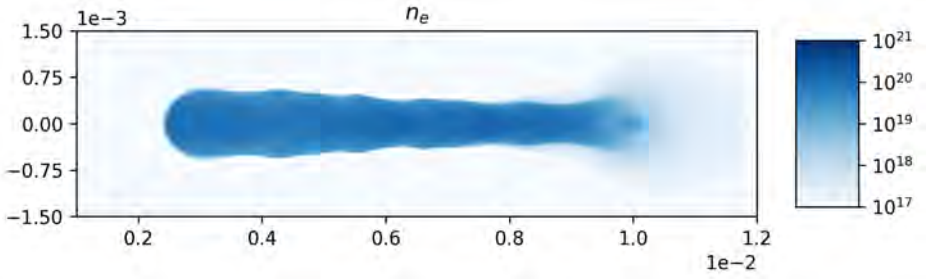


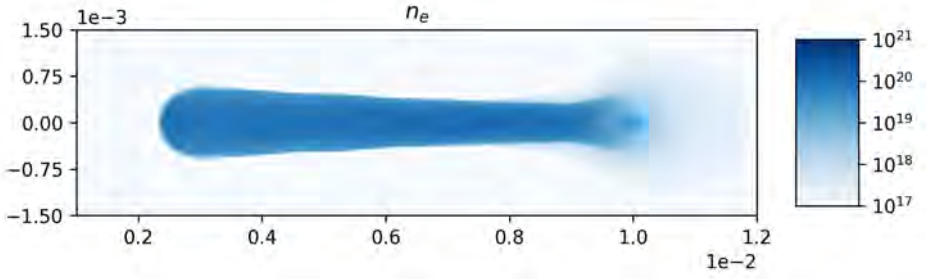
Figure 10.4: The norm of electric field [V/m] (left) and the electron density [m^{-3}] (right) at different instants on the axis $r = 0$ for the ISG scheme with $\varepsilon = 0.01$.



(a) ISG $\Delta x = 3\,\mu m$ - Quad



(b) LLW $\Delta x = 3\,\mu m$ - Tri



(c) LLW $\Delta x = 2\,\mu m$ - Tri

Figure 10.5: Electron density [m^{-3}] at $t = 16\,ns$ for different schemes and resolutions.

maximum electric field as a function of the streamer length in Fig. 10.7 we can clearly see the oscillations present in the LW $\Delta x = 3\,\mu\text{m}$ run.

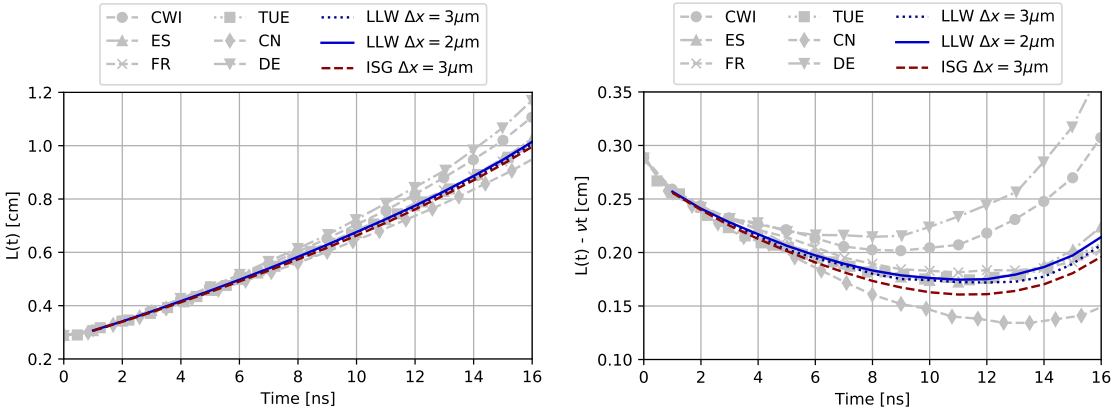


Figure 10.6: Streamer length as a function of time for all the streamer codes and AVIP.

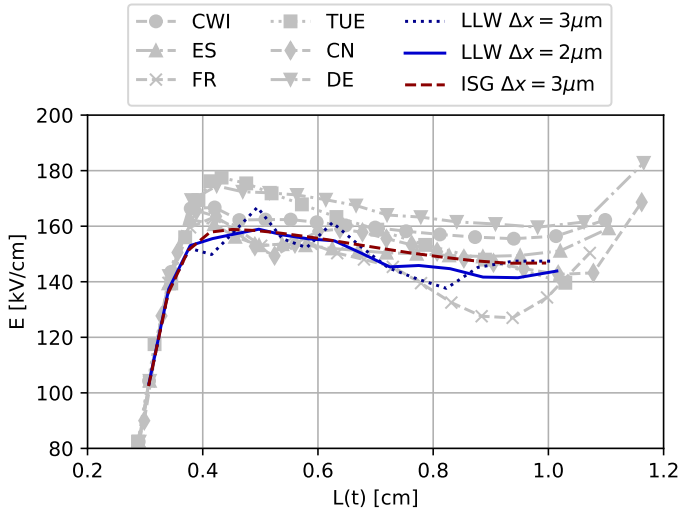


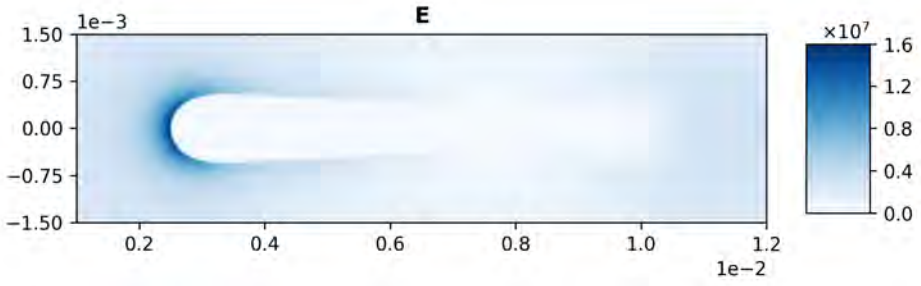
Figure 10.7: Electric field as a function of the streamer length for all the streamer codes and AVIP.

From this first case we can conclude that triangular meshes are able to propagate streamer but are less robust than quadrangular meshes. In all these axisymmetric simulations, the key is to keep the streamer on the axis because in this setting branching is not supposed to occur.

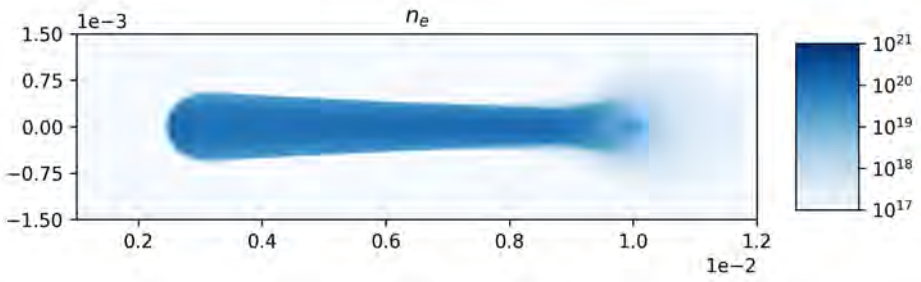
To combine the advantages of the stability of quadrangular elements with the flexibility of triangular elements we decide to use a hybrid approach whenever necessary which is shown in Fig. 10.2c: SG is applied in the regular and topologically dual quadrangular elements while the limited LW scheme is applied in the triangular elements. This new scheme is labeled ISG-LLW and it is implied that hybrid meshes are used whenever we use this scheme. This allows to lower the number of nodes and elements of the mesh while keeping the stability of

the simulation.

Using this hybrid approach a run with $\Delta x = 5\,\mu\text{m}$ has been tested using the `hybrid5micro` mesh. The electron density and electric field at the last instant are shown in Fig. 10.8 and are very close to the results using the full quadrangular mesh of Fig. 10.3 which validates the approach of the hybrid scheme and shows again the robustness of quadrangular elements compared to triangular elements which are stable only for a resolution of $\Delta x = 2\,\mu\text{m}$.



(a) Norm of electric field [V/m]



(b) Electron density [m^{-3}]

Figure 10.8: Electron density and norm of electric field at $t = 16\,\text{ns}$ using the hybrid ISG-LLW numerical scheme on `hybrid5micro` mesh.

Finally few words are given about performance for this test case. The six codes used in the benchmark along with their mesh characteristics and run time are presented in Tab. 10.3. The octree code from the CWI group is by far the fastest at around 20 min core time thanks to a very low number of cells due to very efficient adaptive mesh refinement. The other codes use at least five times more cells, increasing significanty their core time from 6 h to 90 h. The equivalent results using AVIP and the various schemes and meshes mentioned in the section are presented in Tab. 10.4. We use for all runs a bi-socket Intel node with 2 x 18 core Xeon Gold 6140 (2.3 Ghz clock speed and 96 Gb memory). The hybrid option, due to its low number of cells, is the fastest option putting AVIP in second place in terms of performance compared with the other codes.

10.1.2.2 Case 2

The second test case is the stiffest one: the background density is lowered by four orders of magnitude compared to the first case to $1 \times 10^9\,\text{m}^{-3}$. This case is rather artificial since photoionization would not be negligible at these densities but it evaluates the robustness of the schemes implemented.

Code	CWI	ES	FR	CN	TUE	DE
Adaptive refinement	yes	yes	no	yes	no	no
Min grid size	3 µm	3 µm	3 µm	3 µm	3 µm	3 µm
Max grid size				8 µm		5 µm
N_{cells}	1.2×10^5	2.0×10^6	1.1×10^6	6.5×10^5	4.2×10^6	5.1×10^5
Time step	dyn.	1 ps	dyn.	dyn.	dyn.	dyn.
CPU cores	4	1	1	4	1	6
Run time	5 min	20 h	6 h	18 h	25 h	15 h
Core time	20 min	20 h	6 h	72 h	25 h	90 h

Table 10.3: A summary of simulation settings for case 1 of the different codes used in the benchmark Bagheri et al. [2018], adapted from the benchmark paper.

Mesh	`tri3micro`	`tri2micro`	`quad3micro`	`hybrid5micro`
Adaptive refinement	no	no	no	no
Min grid size	3 µm	2 µm	2 µm	5 µm
Max grid size	100 µm	100 µm	100 µm	100 µm
N_{cells}	2.1×10^6	4.66×10^6	1.8×10^6	7.1×10^5
Time step	dyn.	dyn.	dyn.	dyn.
CPU cores	36	36	36	36
Run time	370 s	1337 s	1280 s	220 s
Core time	3.7 h	13.4 h	12.8 h	2.2 h

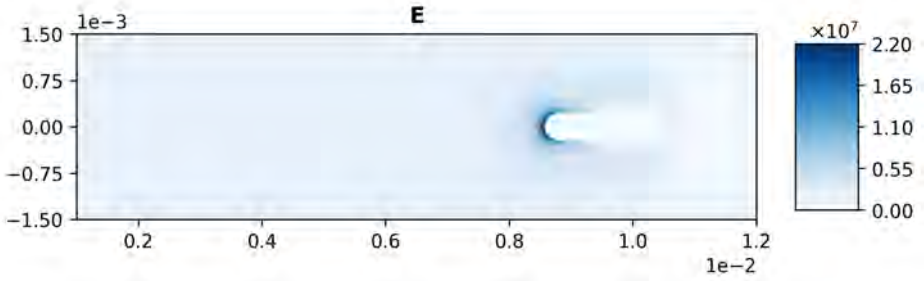
Table 10.4: A summary of simulation settings for case 1 using AVIP.

The resolutions used in the previous case using any scheme are not able to propagate the streamer without branching and here only below a resolution of $\Delta x = 2.5\,\mu\text{m}$ the streamer propagates correctly as shown in Fig. 10.9 for the ISG-LLW scheme using the `hybrid25micro` mesh. Since the background density is lower there is less charged species to collide with and therefore the speed of the streamer is lower in this case by a factor of 2. The peak electric field of the streamer head is also around 30% higher for this case: gradients of charge density are higher due to the lower background causing a higher potential drop and thus higher electric field magnitude.

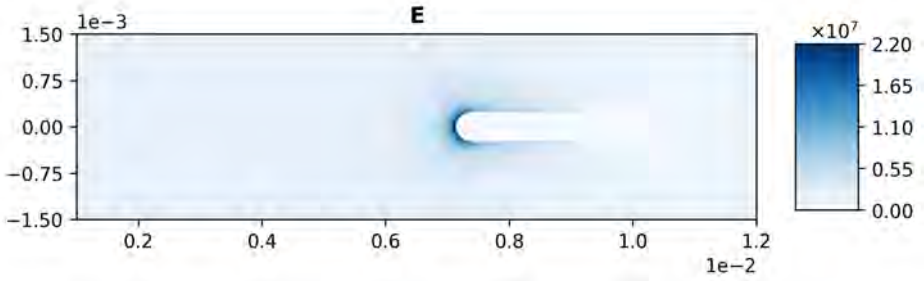
The comparisons of the streamer length as a function of time and peak electric field as a function of the streamer length are shown in Fig. 10.10. Only three out of the six streamer codes are able to correctly propagate the streamer in the benchmark paper [Bagheri et al., 2018] for this case and only those results are shown. As for the first case, good agreement is found between AVIP and the other codes validating the hybrid approach for very stiff cases.

10.1.2.3 Case 3

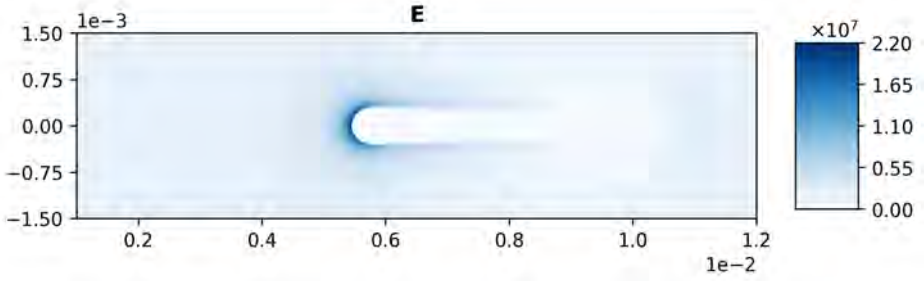
The third test case is the same as the second one but with photoionization activated. Photoionization in AVIP is implemented using the three-term exponential fitting from Bourdon et al. [2007] as detailed in Sections. 2.4.4 and 6.5 which is one of the three models used in Bagheri et al. [2018]. This case is the less stiff among the three cases as this photoionization source term allows to stabilize the streamer: in front of the peak electric field the photoion-



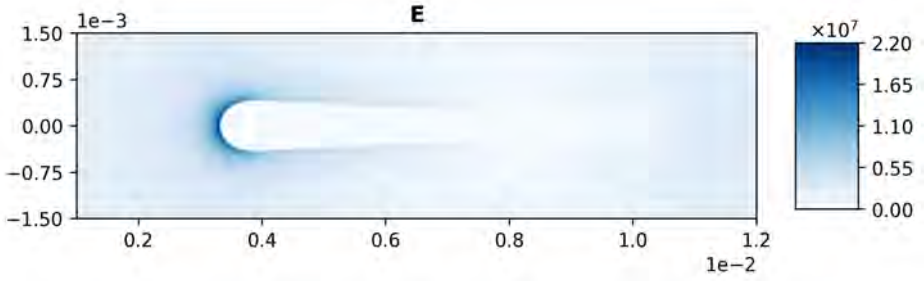
(a) 5 ns



(b) 10 ns



(c) 15 ns



(d) 20 ns

Figure 10.9: Electric field [V/m] of the streamer at different instants for hybrid ISG-LLW at $\Delta x = 2.5\,\mu\text{m}$.

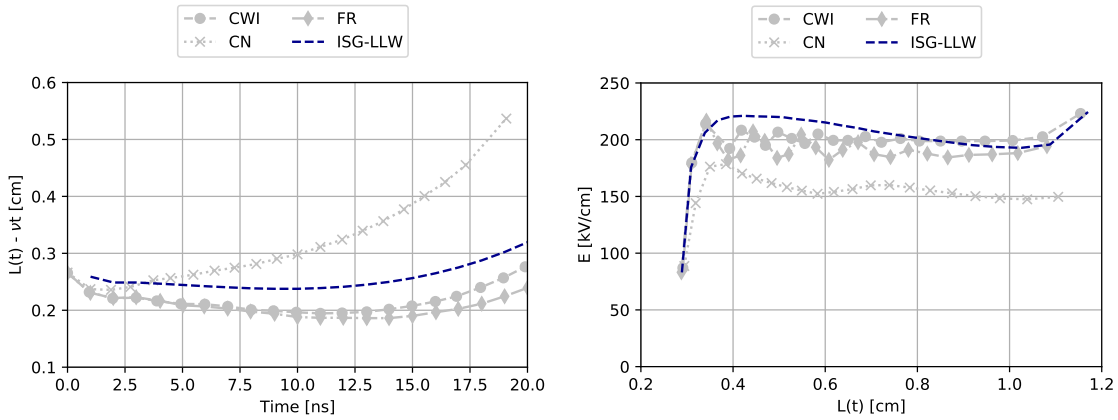


Figure 10.10: Streamer length as a function of time for the streamer codes and AVIP for case 2.

ization feeds the streamer head with charged species. The photoionization source term at different instants is shown in Fig. 10.11. We can see that it provides the streamer a diffusive stream of charged species which ionize the whole area around the streamer head.

All the streamer codes presented in the benchmark are able to run this case with rather coarse resolutions compared to the previous cases. The comparison of AVIP with the three codes already present in Case 2 is shown in Fig. 10.12. Good agreeement is found with LLW on triangular meshes and ISG-LLW on hybrid meshes validating the photoionization implementation in AVIP in both triangular and quadrangular meshes.

10.2 Hyperbole electrodes

10.2.1 Chemistry

We now turn to a more complex case with hyperbolic electrodes. The chemistry is taken from Morrow and Lowke [1997] to be representative of air. It involves three types of species: electrons e, positive p and negative ions n for which the governing equations read:

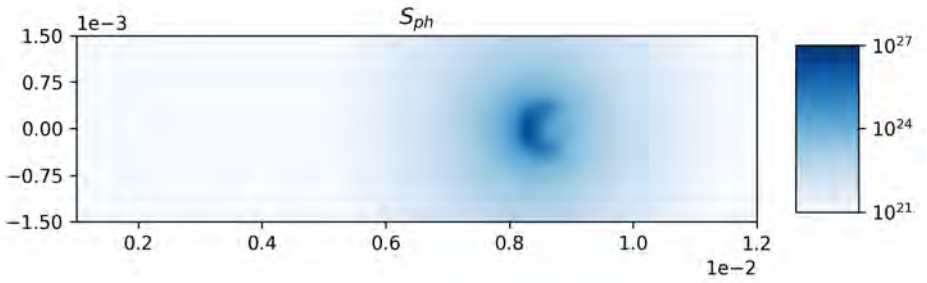
$$\nabla^2\phi = -\frac{e(n_p - n_e - n_n)}{\varepsilon_0} \longrightarrow \mathbf{E} = -\nabla\phi \tag{10.9}$$

$$\frac{\partial n_e}{\partial t} + \nabla\cdot(n_e\mathbf{W_e} - D_e\nabla n_e) = n_e\alpha|W_e| - n_e\eta|W_e| - n_e n_p\beta \tag{10.10}$$

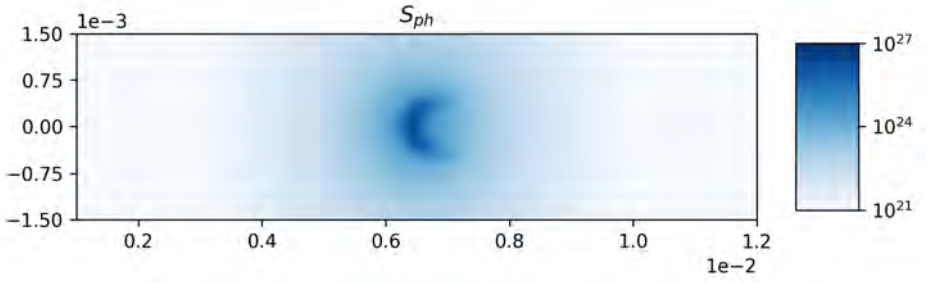
$$\frac{\partial n_p}{\partial t} = n_e\alpha|W_e| - n_e n_p\beta - n_n n_p\beta \tag{10.11}$$

$$\frac{\partial n_n}{\partial t} = n_e\eta|W_e| - n_n n_p\beta \tag{10.12}$$

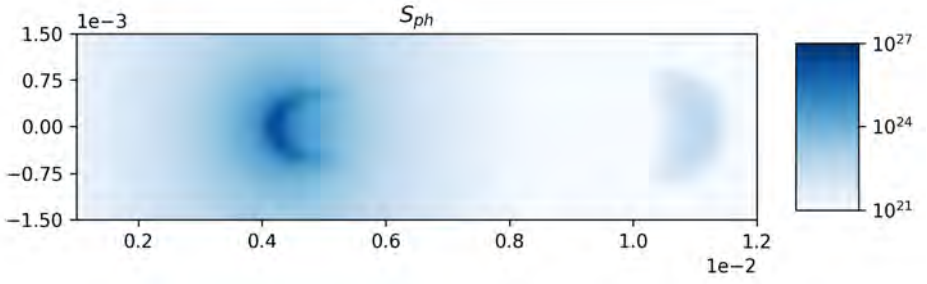
where $\alpha = \alpha(E/N)$ is the ionization coefficient, $\eta = \eta(E/N)$ the attachment coefficient, N the neutral gas density, β the recombination rate, $\mathbf{W}_e = -\mu_e\mathbf{E}$ the drift-velocity of the electrons and $\mu_e = \mu_e(E/N)$ the electron mobility. The chemistry thus involves six reactions which are summarized below:



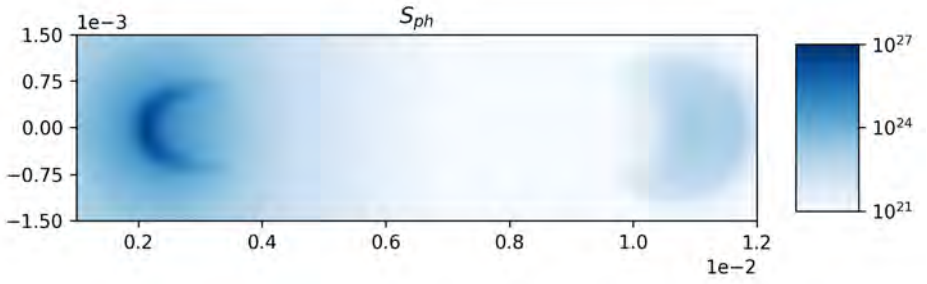
(a) 4 ns



(b) 8 ns



(c) 12 ns



(d) 15 ns

Figure 10.11: Photoionization source term [$m^{-3}.s^{-1}$] at different instants for hybrid ISG-LLW at $\Delta x = 5\,\mu m$.

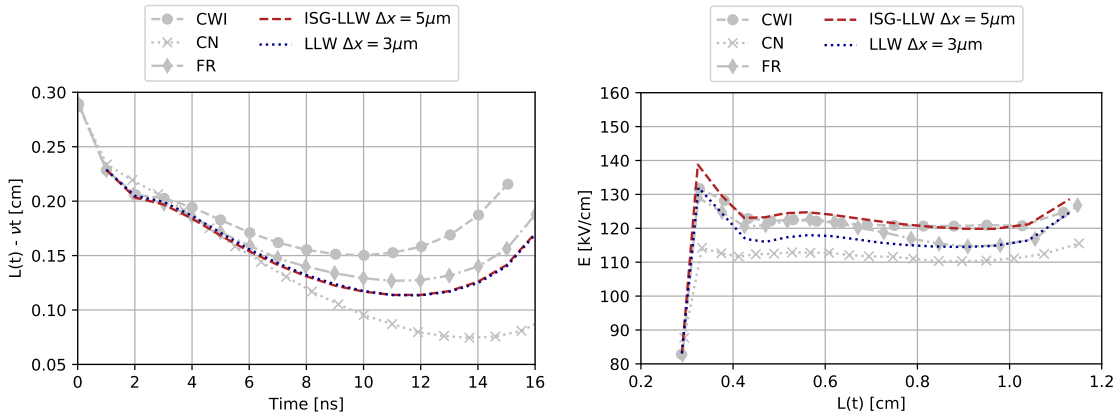


Figure 10.12: Streamer length as a function of time for all the streamer codes and AVIP.

$$\text{Ionization} \quad \mathrm{e}^- + \mathrm{A} \longrightarrow 2\,\mathrm{e}^- + \mathrm{A}^+ \tag{10.13}$$

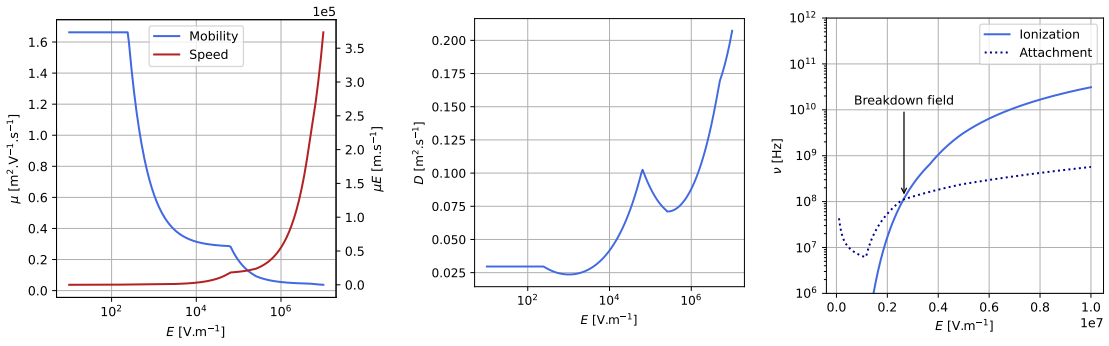
$$\text{Attachment} \quad \mathrm{e}^- + \mathrm{A} \longrightarrow \mathrm{A}^- \tag{10.14}$$

$$\text{Photoionization} \quad \mathrm{A} + \gamma \longrightarrow \mathrm{e}^- + \mathrm{A}^+ \tag{10.15}$$

$$\text{Recombination e-p} \quad \mathrm{e}^- + \mathrm{A}^+ \longrightarrow \mathrm{A} \tag{10.16}$$

$$\text{Recombination n-p} \quad \mathrm{A}^- + \mathrm{A}^+ \longrightarrow 2\,\mathrm{A} \tag{10.17}$$

The local field approximation is assumed for this chemistry, *i.e.* the transport coefficients depend on the reduced electric field E/N. The transport and chemistry coefficients are plotted at 300 K and 1000 K. These coefficients essentially follow the same trends as the chemistry from the Bagheri et al. [2018] benchmark. The breakdown field at 300 K is around $24\,\mathrm{MV\,m^{-1}}$ (Fig. 10.13) which is very close to the value of the Bagheri chemistry. At 1000 K this breakdown field decreases to $8\,\mathrm{MV\,m^{-1}}$ (Fig. 10.14) so that discharges propagate more easily at higher temperatures for a given voltage.

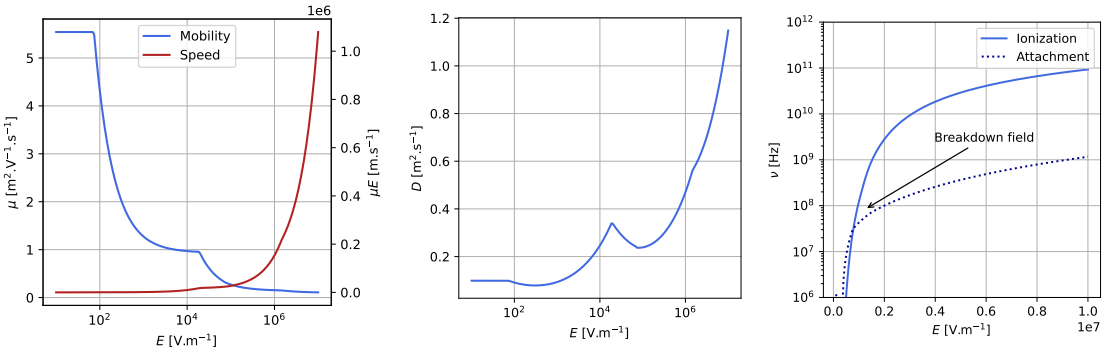


(a) Mobility and speed

(b) Diffusion coefficient

(c) Effective ionization

Figure 10.13: Transport and chemistry coefficients as a function of the electric field norm at 300 K and 1 bar for the Morrow and Lowke [1997] chemistry at 300 K.



(a) Mobility and speed (b) Diffusion coefficient (c) Effective ionization

Figure 10.14: Transport and chemistry coefficients as a function of the electric field norm for the Morrow and Lowke [1997] chemistry at 1000 K.

10.2.2 The spark regime

After the two streamers bridge the gap between the electrodes, a conducting channel is formed and the electric field becomes rather uniform at the value of the Laplacian electric field, *i.e.* the voltage difference divided by the gap distance. At these conditions if the Laplacian electric field is above the breakdown field, we have over-voltage conditions which leads to gas heating [Tholin, 2012]. The electron density increases while the electric field shape stays the same. Hence the dielectric time step can go as low as 10^{-14} s which makes simulations unpractical.

To alleviate the dielectric time step restriction the spark model is adopted during this stage. The Poisson equation is not solved anymore and the electric field and charge density are assumed to follow the same time evolution as the voltage:

$$\frac{1}{||\mathbf{E}||}\frac{\partial||\mathbf{E}||}{\partial t} = \frac{1}{V}\frac{\partial V}{\partial t} \tag{10.18}$$

$$\frac{1}{\rho_q}\frac{\partial \rho_q}{\partial t} = \frac{1}{V}\frac{\partial V}{\partial t} \tag{10.19}$$

The charge difference is applied on the negatively charged species with the highest density (often the electrons).

10.2.3 The discharge energy

An important quantity to monitor is the discharge energy, *i.e.* the energy gained by the charged species due to the electric field. The discharge energy density e^p and energy E^p are defined by:

$$e^p(\mathbf{x}, t) = \int_0^t \mathbf{J} \cdot \mathbf{E} \, \mathrm{d}t \tag{10.20}$$

$$E^p(t) = \int_{V_p} e^p(\mathbf{x}, t) \, \mathrm{d}V \tag{10.21}$$

where V_p is the discharge volume and $\mathbf{J}$ is the total current density. Since the ions are considered frozen during the propagation of the streamer, the total current density reduces to that of the electrons $\mathbf{J}_e$.

This kinetic energy gained by charged species due to the electric field acceleration then goes into the mixture through electron-neutral collisions that can excite and ionize the neutral particles.

10.2.4 Simulations results

We choose a geometry shown in Fig. 10.15 with a 5 mm gap between the two hyperbolic shape electrodes. To easily mesh the electrodes, triangular meshes with a smallest cell-size of about 3 µm in the middle of the gap are used and prove to yield stable simulations.

Dirichlet boundary conditions are applied at the electrodes, the left electrode is the grounded cathode $V_c = 0$ whereas the right electrode is the anode at potential V_a. A rise time of 2 ns from zero to the maximum value of the potential is set and the time evolution of the anode potential is shown in fig. 10.16. The electric field is oriented from right to left and we expect two streamers to propagate across the gap. For the remaining Poisson boundary conditions (BCs), Neumann conditions are enforced at the axis and in the farfield. For plasma transport species, Neumann boundary conditions are applied at the electrodes, a symmetry is applied at the axis whereas an outlet condition is imposed at the farfield.

For all simulations presented in this section, a constant neutral background of electrons and positive ions at $1 \times 10^{15}\,\mathrm{m}^{-3}$ is set initially.

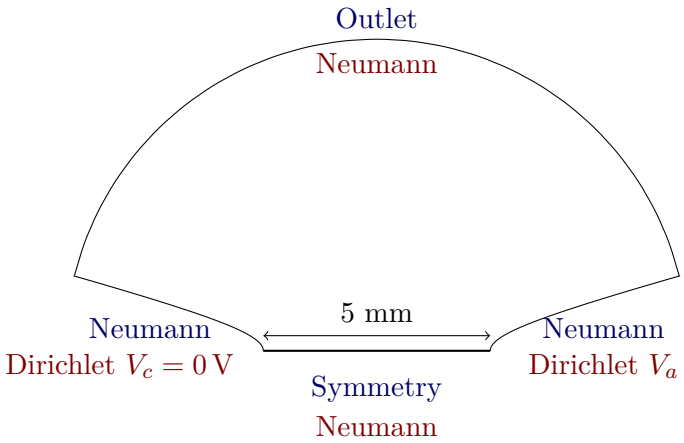


Figure 10.15: Hyperbolic shape electrodes configuration with Poisson boundary conditions (red) and plasma species transport boundary conditions (blue).

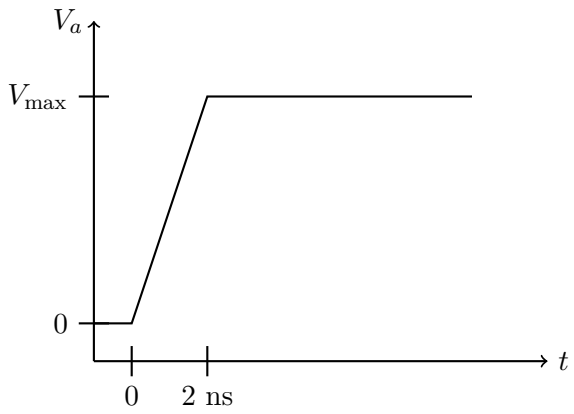


Figure 10.16: Anode potential time evolution.

10.2.4.1 1000 K discharges

Simulations at 1000 K using two hyperbolic shape electrodes of radius $R_c = 200\,\mu\text{m}$ are shown in Figs. 10.19, 10.20 and 10.21 for the norm of electric field, electron density and discharge energy density, respectively. The mesh contains 6.3×10^5 nodes and 1.3×10^6 triangular cells. Three phases are observed: a first phase where the streamer heads are created due to the voltage rise. Once the electric field at the heads (especially the positive streamer around the anode) is high enough, the streamers can propagate inside the gap. Both streamers eventually merge to create a conducting channel. Only after the creation of this conducting channel does the discharge energy density in Fig. 10.21 significantly increases. We also note that the highest values of the discharge energy density are located close to the tips of the electrodes. The time evolution of the discharge energy is shown in Fig. 10.17 where two regimes can be observed: the first one is the slow increase of the discharge energy up to 12 ns. This time corresponds to the creation of the conducting channel and after 12 ns the discharge energy rises much faster.

The streamer coming from the anode, the positive streamer, is much more concentrated than the negative streamer coming from the anode. This can be explained by the sketch shown in Fig. 10.18. The electric field follows the isopotentials shown in gray in the sketch which are hyperboles as well. It can be seen that locally the electric field and hence the electron drift-velocity $-\mu_E \mathbf{E}$ have focusing directions around the anode and diffusive directions around the cathode.

One-dimensional profiles of electron density and electric field norm are shown in Fig. 10.22. At $t = 0.5$ ns the voltage is still rising and no propagation can be seen. At $t = 2.5$ ns the onset of the negative streamer, with a local maximum electric field, is observed. This negative streamer has already moved around 0.5 mm while the positive streamer is still stuck aroudn the anode. We can note that the peak electric field is two times higher for the positive streamer than for the negative streamer at this time. At $t = 7.5$ ns the negative streamer is at the middle of the gap and the peak electric field has considerably decreased while the positive streamer moved about half the distance and maintained its peak electric field. However, once the positive streamer is started it bridges the gap faster and overall streamer propagation speed is on the order of cm/ns approximately in these cases.

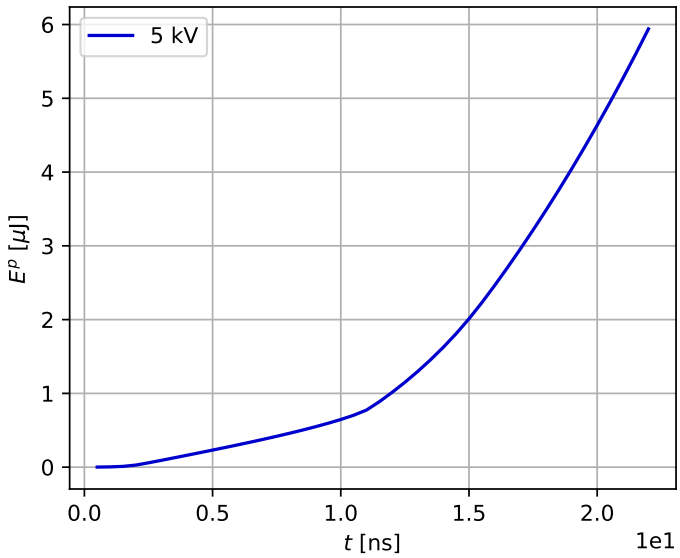


Figure 10.17: Discharge energy time evolution for the $R_c = 200\,\mu\text{m}$ at 1000 K and with peak anode potential of 5 kV.

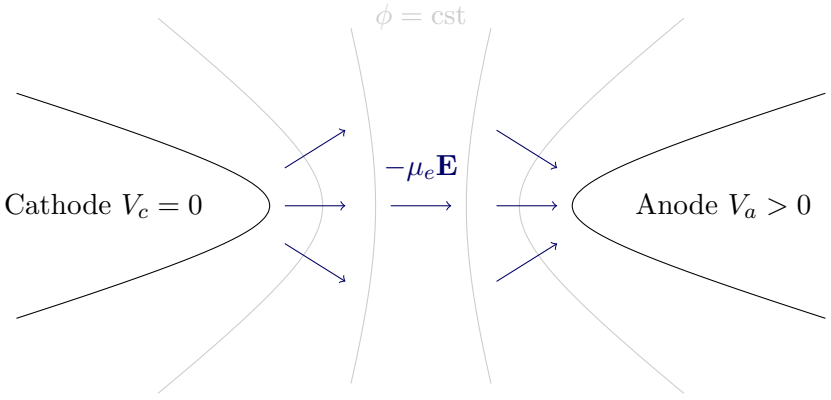


Figure 10.18: Sketch of streamer dynamics in pin-pin configurations.

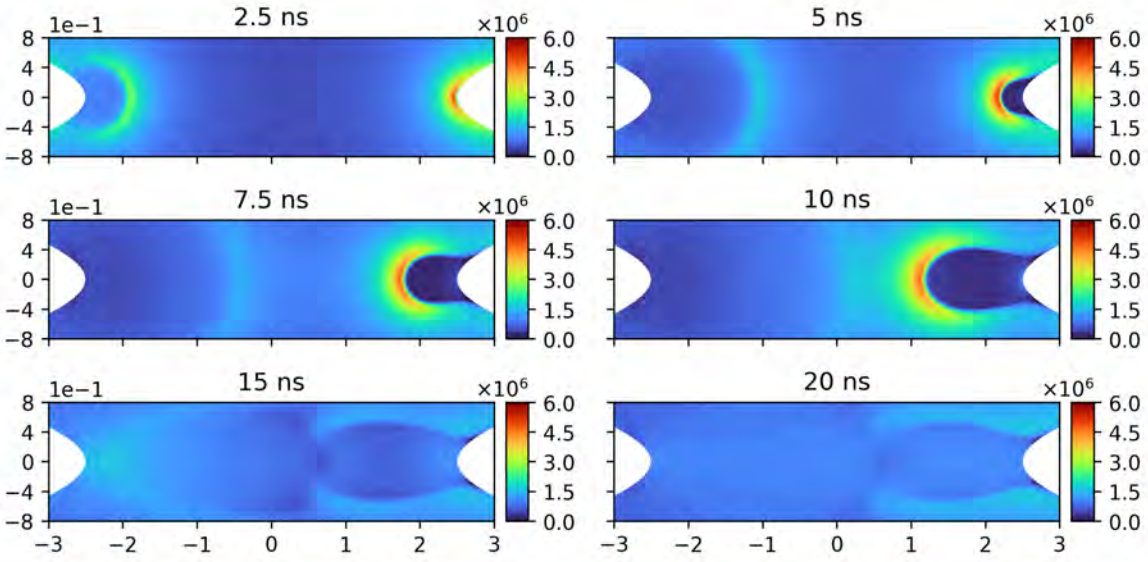


Figure 10.19: Electric field [V.m^{-1}] at different instants for two hyperbolic shape electrodes with $R_c = 200\,\mu\text{m}$ at 1000 K. Length units are in mm.

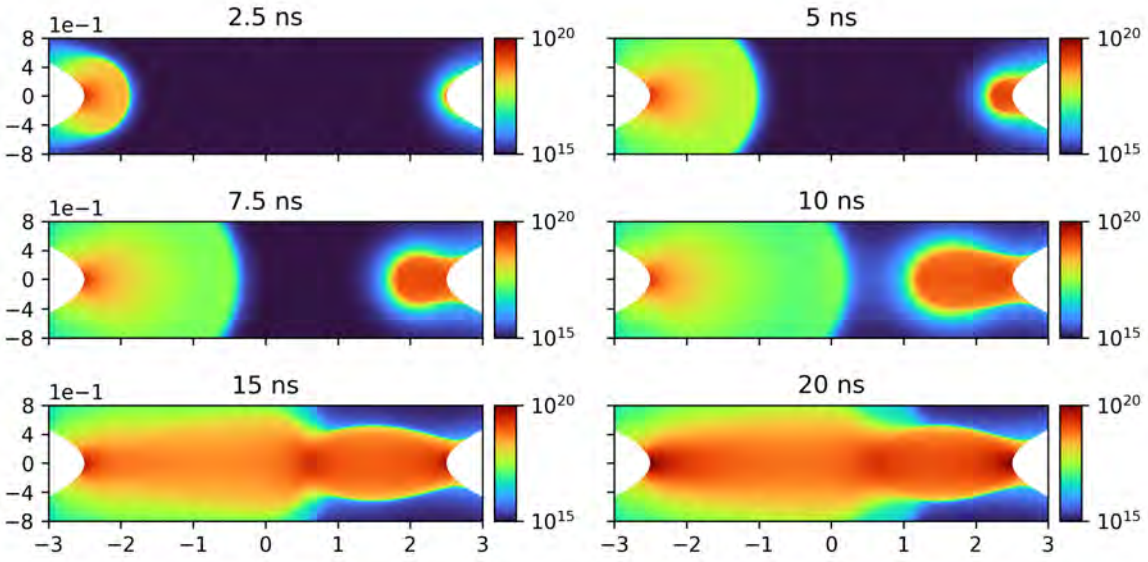


Figure 10.20: Electron density [m^{-3}] at different instants for two hyperbolic shape electrodes with $R_c = 200\,\mu\text{m}$ at 1000 K. Length units are in mm.

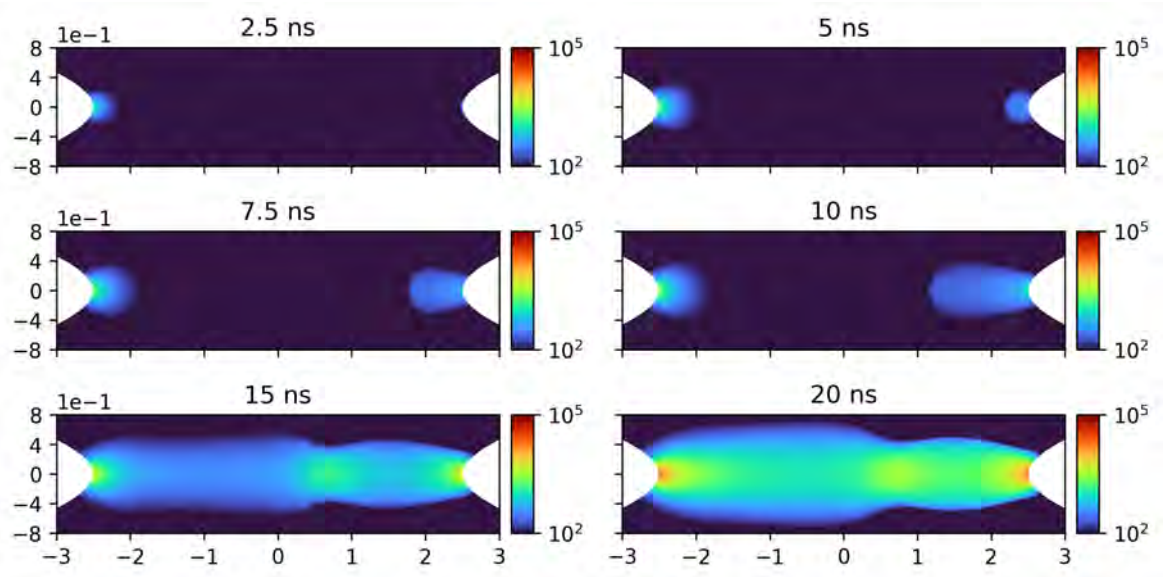


Figure 10.21: Discharge energy density [J.m^{-3}] at different instants for two hyperbolic shape electrodes with $R_c = 200\,\mu$m at 1000 K. Length units are in mm.

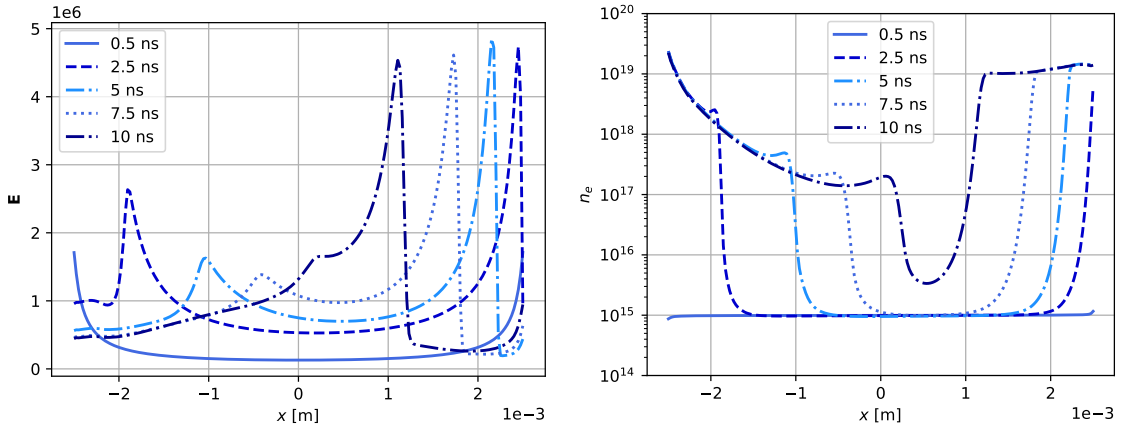


Figure 10.22: The norm of electric field (left) and the electron density (right) at different instants on the axis $r = 0$ for the LLW scheme with $\beta = 1.0$.

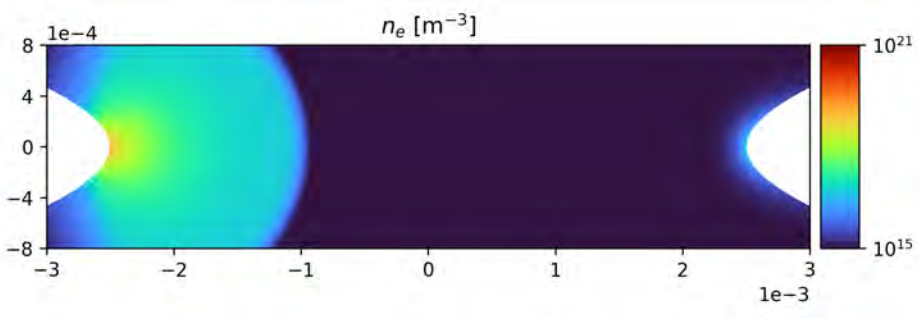


Figure 10.23: Electron density at 10 ns for 3 kV applied at the anode at 1000 K.

Lowering the value of the anode potential to 3 kV causes no streamer propagation as shown in Fig. 10.23 to be compared with Fig. 10.20. No positive streamer propagation can be seen: the electric field in the positive streamer head is not high enough to cause its propagation so that only the negative streamer is able to propagate correctly. Hence 5 kV in this case is not very far above the limit for the onset of the streamer propagation.

10.2.4.2 300 K discharges

We now turn to discharge simulations in standard conditions of temperature and pressure at 300 K and 1 bar, respectively. To obtain a similar streamer as the 5 kV streamer at 1000 K we need to have a similar reduced electric field E/N. Hence since density has been increased threefold the applied voltage also needs to be increased by the same amount: hence a value of 15 kV is chosen and results are show in Figs. 10.24, 10.25 and 10.26 for the electric field, the electron density and the discharge energy density, respectively. Compared to the 1000 K case the streamer is much faster bridging the gap in about 7 ns compared to 12 ns at 1000 K. This is explained by the higher value of neutral background density N that is increased more than threefold compared to 1000 K: electrons have more colliding partners resulting in higher charged species densities due to more ionization events. This drives the electric field up and leads to faster streamer propagation.

The shape of the discharge is also different from the 1000 K discharge: the negative streamer is more concentrated than at 1000 K but the positive streamer has overall a bigger radial size. Due to the higher densities involved, the discharge energy density is an order of magnitude higher at 300 K than it is at 1000 K.

10.2.4.3 Hybrid meshing

As discussed in the Bagheri et al. [2018] benchmark results, triangular elements need to have finer resolution than quadrangular elements in order to propagate correctly streamers and quadrangular elements with a coarser resolution of about two times propagate equally well streamers. A hybrid mesh design is considered shown in Fig. 10.27 to be able to still use the ISG scheme where a rectangle block of rectangles is considered in the gap between the electrodes and triangles are used elsewhere.

A mesh of resolution $\Delta x = 5\,\mu$m is used in this case, increasing the mesh size by a factor of 1.7 and results for the 1000 K - 5 kV propagation is shown in Fig. 10.28. The propagation

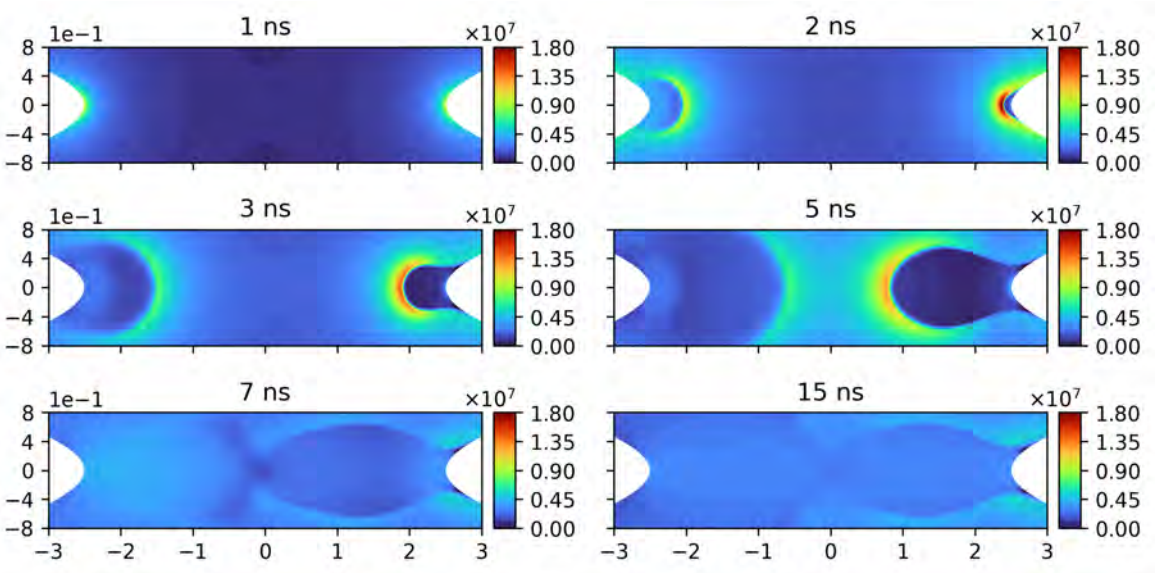


Figure 10.24: Electric field [V.m^{-1}] at different instants for two hyperbolic shape electrodes with $R_c = 200\,\mu m$ at 300 K. Length units are in mm.

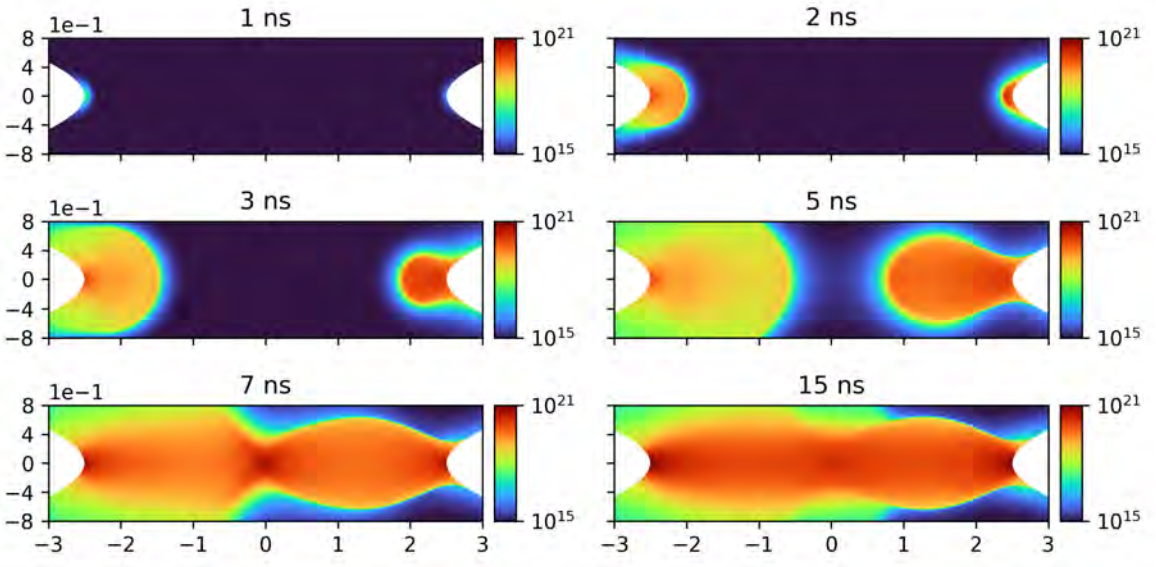


Figure 10.25: Electron density [m^{-3}] at different instants for two hyperbolic shape electrodes with $R_c = 200\,\mu m$ at 300 K. Length units are in mm.

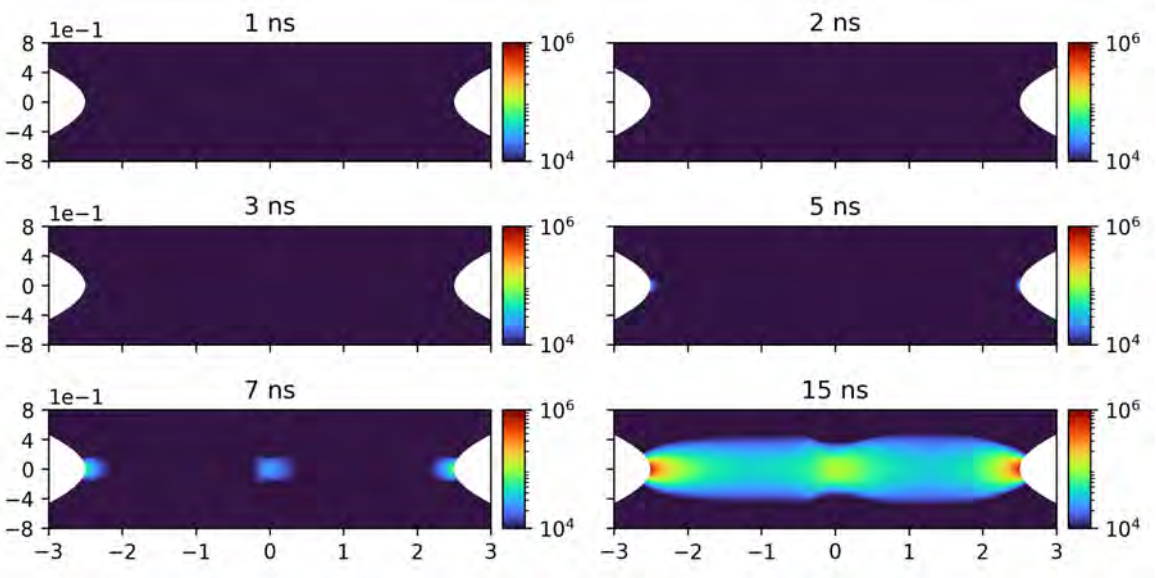
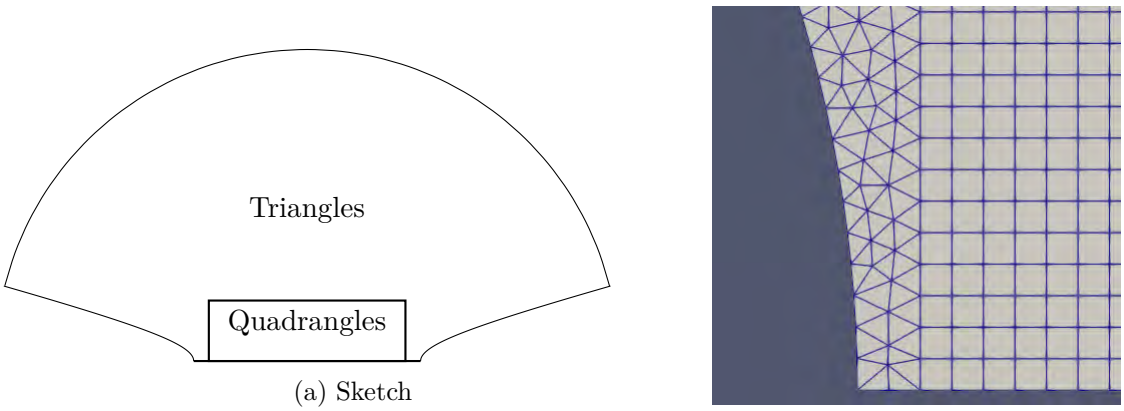


Figure 10.26: Discharge energy density [J.m^{-3}] at different instants for two hyperbolic shape electrodes with $R_c = 200\,\mu\text{m}$ at 300 K. Length units are in mm.

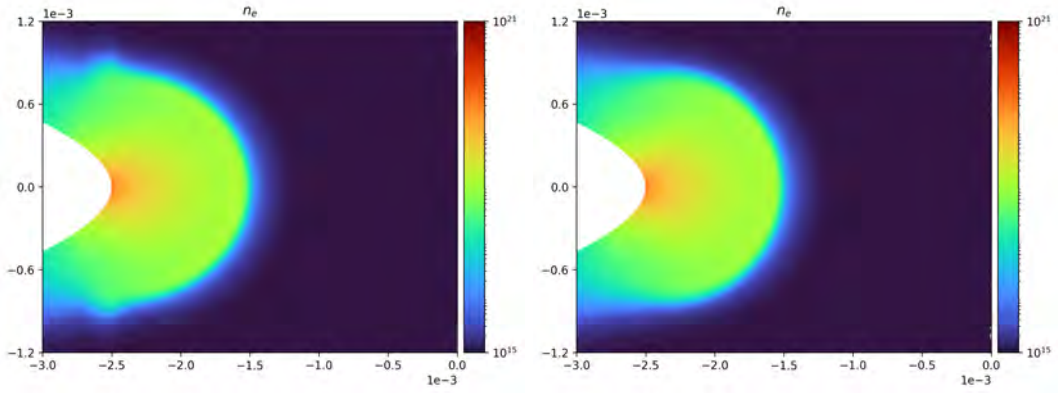


(a) Sketch

(b) Mesh transition

Figure 10.27: Hyperbolic shape electrodes hybrid meshing.

is identical to that of the triangular mesh with resolution $\Delta x = 3\,\mu m$ in Fig. 10.20. However the mesh transition from quadrangles to triangles can be clearly observed and leads to a discontinuity in the profiles so that the radial size of the rectangular box of quadrangular elements had to be set quite high to not see this transition. This transition discontinuity needs to be fixed in order for this hybrid scheme to be used in predictive cases and could be the subject of future works in the code.



(a) Hybrid mesh (b) Triangular mesh

Figure 10.28: Electron density [m^{-3}] at the same time for two hyperbolic shape electrodes with $R_c = 200\,\mu m$ at 1000 K using a hybrid mesh/ISG-LLW hybrid scheme and using a triangular mesh/LLW scheme.

Chapter 11

Laminar PAC simulations

Contents

In this chapter, detailed simulations of plasma assisted combustion in laminar conditions are performed in a pin-pin configuration. The methodology to model plasma discharges is first presented in Section. 11.1. The complete fully-coupled simulation iteration procedure is then detailed in Section. 11.2. The fully-coupled simulations are then presented in an article, submitted at the 39th International Symposium on Combustion.

11.1 NRP discharge simulation in pin-pin configurations

NRP discharges are on the order of 10 ns long and have a frequency around 10-100 kHz. Depending on the voltage applied three regimes can be observed in pin-pin configurations [Pai et al., 2010]: the corona regime, the glow regime and the spark regime. In the corona regime only the extremities of the point electrodes emit light. The glow regime is obtained for higher voltages and a diffusive emitting channel is observed without gas heating so that the discharge remains non-thermal. At even higher voltage, the spark regime is obtained where the discharge begins to heat the gas and emits strongly. To simulate NRP discharges, three phases are adopted: first the propagation phase where the streamers bridge the gap between the electrodes corresponding to the corona and glow regimes. Depending on the voltage applied there can be a spark phase where gas heating occurs. Finally the interpulse phase where only diffusion effects are noticeable is simulated assuming ambipolar diffusion allowing the use of larger time steps.

11.1.1 Propagation phase - the glow regime

The propagation phase lasts tens of nanoseconds and since the mass of the electrons is at least three orders of magnitude lighter than the ions only the electrons are considered moving

in this phase. The resulting drift-diffusion equations for electrons e and ions i coupled to the electric field are

$$\nabla^2\phi = -\left(\sum_i q_i n_i - e n_e\right)/\varepsilon_0 \longrightarrow \mathbf{E} = -\nabla\phi \tag{11.1}$$

$$\frac{\partial n_e}{\partial t} + \nabla\cdot(-n_e\mu_e\mathbf{E} - D_e\nabla n_e) = S_e \qquad \frac{\partial n_i}{\partial t} = S_i \tag{11.2}$$

11.1.2 The spark regime

As explained in Section. 10.2.2 and recalled here, to alleviate the dielectric time step restriction the spark model is adopted during such a phase. The Poisson equation is not solved anymore and the electric field and charge density are assumed to follow the same time evolution as the voltage:

$$\frac{1}{||\mathbf{E}||}\frac{\partial||\mathbf{E}||}{\partial t} = \frac{1}{V}\frac{\partial V}{\partial t} \tag{11.3}$$

$$\frac{1}{\rho_q}\frac{\partial \rho_q}{\partial t} = \frac{1}{V}\frac{\partial V}{\partial t} \tag{11.4}$$

The charge difference is applied on the negatively charged species with the highest density (often the electrons).

11.1.3 Interpulse phase

During the 10-100 µs gap between the NRP discharges, the electric field is very low and the conducting channel diffuses. At very low electric field, the diffusion velocities of electrons is around ten thousand times the diffusion velocity of ions. A space charge is thus created and the resulting electric field slows down the electrons while accelerating ions so that in the end both electrons and ions diffuse at the same speed preserving the quasi-neutrality of the plasma. This is the so called ambipolar diffusion Raimbault [2018]. Simulation on a coarser mesh can be performed where ambipolar diffusion is assumed so that

$$\sum_i q_i \mathbf{F}_i = 0 \tag{11.5}$$

where $\mathbf{F}_i = n_i \text{sign}(q_i)|\mu_i|\mathbf{E} - D_i\nabla n_i$. The value of the ambipolar field is then

$$\mathbf{E}_{\text{amb}} = \frac{\sum \text{sgn}(q_i) D_i \nabla n_i}{\sum |n_i \mu_i|} \tag{11.6}$$

Following [Tholin, 2012, Chap. 3.2.2] the diffusion coefficients of all the species are taken equal to the diffusion of the positive species.

11.2 Multi-dimensional simulation methodology

So depending on the phase of the discharge, the electric field is not computed the same way and the resulting drift-diffusion equations are solved:

$$\mathbf{E}^{\text{poisson}} \quad \text{or} \quad \mathbf{E}^{\text{spark}} \quad \text{or} \quad \mathbf{E}^{\text{amb}} \tag{11.7}$$

$$\frac{\partial n_i}{\partial t} + \nabla \cdot \mathbf{\Gamma}_i = S_{0i} \qquad \text{with } \mathbf{\Gamma}_i = n_i \mu_i \mathbf{E} - D_i \nabla n_i \tag{11.8}$$

For the gas flow, the reactive Navier-Stokes equations are solved and we recall Eq. (4.61) without body forces:

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \nabla \cdot [\rho \mathbf{u}] = 0 \\
&\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot [\rho \mathbf{u}\mathbf{u} + p\mathbf{I} - \boldsymbol{\tau}] = 0 \\
&\frac{\partial \rho E}{\partial t} + \nabla \cdot [(\rho E + p)\mathbf{u} - \boldsymbol{\tau} \cdot \mathbf{u} + \mathbf{q}] = \dot{\omega}_T + \dot{Q} \\
&\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot [\rho Y_k (\mathbf{u} + \mathbf{V}_k)] = \dot{\omega}_k
\end{aligned} \tag{11.9}$$

Formally we rewrite the charged species conservative variables as $\mathbf{U}_p$ and the mixture conservative variables as $\mathbf{U}$ so that in compact form the drift-diffusion and reactive Navier-Stokes read:

$$\frac{\partial \mathbf{U}_p}{\partial t} + \nabla \cdot \mathbf{F}_p = \mathbf{S}_p \tag{11.10}$$

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{S} \tag{11.11}$$

For each set of equations, there is a transport source term (under the divergence) and a chemical source term (the right hand side). The chemical source term is where the drift-diffusion and the reactive Navier-Stokes equations are *coupled* since it is a common chemistry. We denote their respective residuals by $\mathbf{R}_p^t$ and $\mathbf{R}_p^c$ for the charged species and by $\mathbf{R}^t$ and $\mathbf{R}^c$ for the gas mixture. Charged species use the LLW scheme which has one temporal stage whereas the mixture equations are integrated using HLLC MUSCL RK2 so that there is an intermediate temporal stage. We denote the current iteration by superscript n so that one iteration of fully coupled simulation proceeds as follows

1. Compute the electric field $\mathbf{E}$

2. Compute the time step Δt

3. Compute the common chemistry source term $\mathbf{R}^c(\mathbf{U}, \mathbf{U}_p)$ for the gas and $\mathbf{R}_p^c(\mathbf{U}, \mathbf{U}_p)$ for the charged species

4. Time advancement of the gas conservative variables using HLLC MUSCL RK2 time integration with centered diffusive fluxes

$$\tilde{\mathbf{U}}^n = \mathbf{U}^n + \frac{\Delta t}{2}[\mathbf{R}^t(\mathbf{U}^n) + \mathbf{R}^c(\mathbf{U}, \mathbf{U}_p)] \tag{11.12}$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t[\mathbf{R}^t(\tilde{\mathbf{U}}^n) + \mathbf{R}^c(\mathbf{U}, \mathbf{U}_p)] \tag{11.13}$$

$$\tag{11.14}$$

5. Time advancement of the plasma conservative variables using RK1 time integration using LLW with centered diffusive fluxes

$$\mathbf{U}_p^{n+1} = \mathbf{U}_p^n + \Delta t[\mathbf{R}_p^t(\mathbf{U}^n) + \mathbf{R}_p^c(\mathbf{U}, \mathbf{U}_p)] \tag{11.15}$$

We have chosen two compatible time integration schemes: the common chemistry source term is only computed once so that only the transport residual is updated in the intermediate stage of the gas RK2 time integration.

Lastly, the EEDF is solved using BOLSIG+ and tabulated against the reduced electric field E/N, at the beginning of each pulse using an averaged composition in the discharge region. This allows to keep the memory storage and computational cost of solving the EEDF low while still preserving accuracy.

Concerning computational cost, the chemistry is the bottleneck as the number of equations scale with the number of species. For the reduced chemistry at hand, the actual computation of the source term of each species is around 30% higher than the computation of convection and diffusion scheme.

11.3 Multi-dimensional Ignition

Applying the methodology above to the three phases for streamer propagation the ignition of a methane-air mixture at equivalence ratio $\phi = 0.8$ using NRP discharges is now studied. The chemistry is the reduced coupled chemistry presented in Chap. 9. This study is the subject of a paper submitted to the 39th Int. Symposium on Combustion, which is reported in the following.

Investigation of the impact of NRP discharge frequency on the ignition of a lean methane-air mixture using fully coupled plasma-combustion numerical simulations

N. Barleon[a], L. Cheng[a], B. Cuenot[a], O. Vermorel [a], A. Bourdon[b]

[a] *CERFACS, 42 Avenue Gaspard Coriolis, 31057 Toulouse, Cedex 1, France*
[b] *Laboratoire de Physique des Plasmas (LPP), CNRS, Sorbonne Université, Ecole Polytechnique, Institut Polytechnique de Paris, 91120 Palaiseau, France*

Abstract

The ignition of a laminar premixed methane/air mixture by Nanosecond Repetitively Pulsed (NRP) discharges in a pin-pin configuration is studied using fully coupled plasma-combustion numerical simulations. These simulations are performed using the AVIP code specifically developed for low temperature plasma modeling and coupled to the combustion code AVBP. A reduced chemical scheme for plasma-assisted combustion previously derived and validated is used to investigate the effect of the frequency of NRP discharges and the benefits of their chemical enhancement. It is observed that the induced shock wave produced by strong discharges is of major importance for ignition and can lead to quenching of the ignition kernels through strong induced recirculation of gases. Increasing the frequency of the discharges reduces this effect by depositing less energy at each discharge and accumulating energy more homogeneously between the electrodes, leading to a faster and more stable ignition. The minimum energy necessary to ignite decreases with increasing frequency and at the highest studied frequency (100 kHz) ignition has been achieved with 30% less energy than with a single-pulse discharge.

Keywords: NRP discharge; Plasma-Assisted Combustion; Ignition; Detailed simulations

1. Introduction

In the context of lean combustion for reduced environmental impact, plasma-assisted combustion is currently investigated by several groups to address stability and ignition issues [1–3].

Among the various types of discharges, Nanosecond Repetitively Pulsed (NRP) discharges have experimentally been shown to have a significant impact on flame stabilization [2, 4] and ignition [1]. NRPs produce non-thermal plasmas and active radical species which lead to increased efficiency [5, 6].

Although their beneficial effect is well known, the mechanisms at play in plasma-assisted combustion using NRPs are still not fully understood. In particular the coupling between plasma physics and combustion thermo-chemistry is not clearly established. Complementary to experiments, numerical simulation may be used to analyze in detail the flame-discharge interaction and the underlying physical processes.

The simulation of plasma-assisted combustion requires a plasma model, a combustion model and a coupling method between them. NRP discharges are low temperature plasmas having a streamer structure, and their simulation is well documented in [7] and references therein. Likewise numerical combustion has been an ongoing research subject for the past 50 years and has reached today a good level of accuracy [8]. However, modeling the coupling between both phenomena raises a number of difficult issues. It is recognized [9] that the interaction between a NRP discharge and a flame is dominated by two main mechanisms increasing the mixture reactivity: a fast gas heating and radical production mainly due to the quenching of electronically excited nitrogen molecules. Simulating these mechanisms with sufficient accuracy and predictivity in real configurations remains a challenge. In particular the two-way plasma-combustion coupling is made difficult by the difference in their respective time scales, of the order of 10^{-9} s for a streamer and $10^{-6} - 10^{-3}$ s for combustion.

Efforts have been made recently in order to simulate plasma-assisted combustion in hydrogen-air and methane air mixtures [4, 10–13]. In [4, 13], the simulation of plasma is replaced by an empirical model based on experimental results. On the other hand other works have focused on the streamer simulation and the influence of the presence of fuel in radical production [10].

More detailed simulations have been conducted in hydrogen-air mixtures using a one-way coupling [12, 14], *i.e.*, with only the plasma influence on combustion. One-way coupling usually consists in taking the final solution of the discharge phase as an input for the combustion code. However, this approach is not able to capture the combustion and flow effects on the plasma. While this approach is justified for a single pulse study, it can not capture the two-way coupling occurring on the microsecond timescale that is of interest for multiple pulses studies. Simulations using a two-way coupling are usually limited to 0D or 1D computations [15] or 2D cylindrical simulations of a single-pulse ignition [16].

In the present work, a two-way coupled plasma-combustion numerical model is used to describe the plasma, the combustion and their interaction with a similar level of accuracy to simulate the ignition of a methane-air mixture using multiple discharges in a 2D axisymmetric geometry.

The modeling and numerical approach is first detailed in Section 2. In Section 3 the considered pin-pin electrode configuration for ignition of a lean methane-air mixture is presented. Then results obtained with single-pulse and repetitive pulses are analysed in Section 4 and compared with a conventional heat energy deposition.

2. Numerical methodologies

The plasma and combustion governing equations are discretized in cylindrical coordinates with a finite volume formulation using the AVIP code [17]. AVIP is a massively parallel unstructured plasma code able to simulate low-temperature plasma discharges, and validated on the streamer code benchmark [18]. It has been coupled to the combustion code AVBP [19] in a self-consistent approach based on the simultaneous time-integration of both sets of governing equations using the same domain discretization.

2.1. Combustion model

The set of conservation equations used to describe the evolution of a compressible flow with chemical reactions yields:

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_i u_j\right) = -\frac{\partial \sigma_{ij}}{\partial x_j}$$
$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho E u_j\right) = -\frac{\partial}{\partial x_j}\left[u_i \sigma_{ij} + q_j\right] + \dot{\omega}_T$$
$$\frac{\partial \rho_k}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho_k u_j\right) = -\frac{\partial}{\partial x_j}\left[J_{j,k}\right] + \dot{\omega}_k$$

where ρ is the mass density, u_i the velocity, E the total non-chemical energy, ρ_k the mass density of species k, σ_{ij} the stress tensor, q_j the heat flux, $J_{j,k}$ the diffusive flux of species k into the mixture, $\dot{\omega}_k$ the species k mass production rate and $\dot{\omega}_T$ the heat produced by the chemistry.

Notice that a DNS framework is employed in this work for the gaseous mixture without any flame model. The HLLC Riemann solver with MUSCL reconstruction and Sweby limiter [20] is employed to compute the Euler fluxes of the Navier-Stokes equations. Validation cases may be found in [17].

2.2. Low-temperature plasma model

A two-temperature fluid model was developed for the present study. The ion and neutral species are in thermal equilibrium at the gas temperature T_g whereas electrons are at a temperature T_e different from T_g. To retrieve the electron temperature T_e, the electron energy distribution function F_0, which is mostly a function of the reduced electric field E/N (N is the total gas number density) and the species molar fractions X_k, needs to be solved so that $T_e = T_e(F_0)$. By doing so, a local field approximation (LFA) is implied, meaning that F_0 is instantly correlated with the reduced electric field. This is done using BOLSIG+ [21] that has been embedded in AVIP.

All the charged species are transported following the drift diffusion approximation which reads for each species k:

$$\frac{\partial n_k}{\partial t} + \nabla \cdot (n_k \mu_k \mathbf{E} - D_k \nabla n_k) = \dot{\omega}_k \qquad (1)$$

where n_k is the particle density, $\mathbf{E}$ the electric field, μ_k the mobility coefficient, D_k the diffusion coefficient and $\dot{\omega}_k$ the source term associated to the plasma chemistry. The electron mobility and diffusion coefficient are computed using BOLSIG+ [21] at the beginning of each pulse considering an averaged mixture composition within the inter-electrode gap. All the ions mobilities and diffusion coefficients are taken from [22]. This model has been extensively used for high pressure streamer simulations [7, 11, 18]. The limited Lax-Wendroff numerical scheme [17] is used to integrate the drift-diffusion equations.

The electric field is derived from the Poisson equation:

$$\nabla^2 \phi = -\frac{1}{\epsilon_0} \sum_k q_k n_k \quad \text{with} \quad \mathbf{E} = -\nabla \phi \qquad (2)$$

where ϕ is the electric potential and q_k is the species k charge. The Poisson equation is discretized using a finite volume vertex-centered formulation [17] and the resulting linear system is solved using the PETSc library [23].

The spark phase model developed in [11] is employed to compute the electric field in order to relax the dielectric time step when it becomes lower than 5×10^{-14} s. Moreover, the inter-pulse phase is modeled assuming plasma neutrality and ambipolar diffusion model as presented in [11].

2.3. Vibrational energy

The vibration of nitrogen plays an important role in plasma-assisted combustion as it can store up to 50% of the discharge energy [1]. Detailed modeling of vibration is too costly and a global out-of-equilibrium vibrational energy model is considered here as done in [13, 24]. The non-equilibrium vibrational energy e_{vib} governing equation reads:

$$\rho \frac{D e_{\mathrm{vib}}}{Dt} = \frac{\partial}{\partial x_i} \left(\rho \mathcal{D}_{\mathrm{N_2}} \frac{\partial e_{\mathrm{vib}}}{\partial x_i} \right) + \dot{E}^p_{\mathrm{vib}} - \dot{R}^p_{VT} \qquad (3)$$

where $\mathcal{D}_{\mathrm{N_2}}$ is the diffusion coefficient of N_2 into the mixture, $\dot{E}^p_{\mathrm{vib}}$ is the vibrational energy production term computed from the electron-impact processes and $\dot{R}^p_{VT}$ is the vibrational to translational relaxation modeled using the Landau-Teller harmonic oscillator approach. Relaxation times τ^k_{VT} are computed using experimental coefficients of Capitelli [25] for H_2O, CO_2 and of Millikan and White [26] for N_2, O_2, O.

$$\dot{R}^p_{VT} = \rho \frac{e_{\mathrm{vib}}}{\tau_{VT}} \, , \; \tau_{VT} = \left(\sum_k \frac{1}{\tau^k_{VT}} \right)^{-1} \qquad (4)$$

3. Configuration and numerical set-up

A pin-pin electrode geometry is chosen as in [7, 11]. Multiple NRP discharges are simulated in a methane-air mixture at an equivalence ratio of 0.8 and an initial temperature of 600 K. The hyperbolic-shaped electrodes both have a 200 µm-curvature radius and are separated by a 4 mm gap. The voltage profile is shown in Fig. 1 where an initial rise of 1 ns followed by a constant plateau of 8 kV are imposed. This corresponds to a mean reduced electric field of 160 Td in the gap between electrodes at the initial temperature. The discharge energy E^p and energy density e^p are defined by:

$$E^p(t) = \int_{V_p} e^p \, \mathrm{d}V \quad \text{and} \quad e^p = \int_0^t \mathbf{j_e} \cdot \mathbf{E} \, \mathrm{d}t \qquad (5)$$

where $\mathbf{j_e}$ is the electron current density and V_p the discharge volume. In order to control the discharge energy deposition, the voltage is switched off at t_{off} when the discharge energy E^p reaches a targeted value E_d.

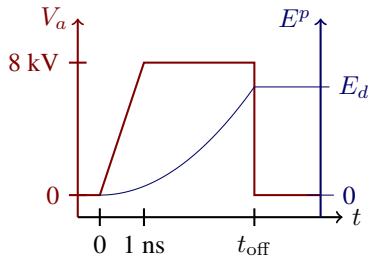


Fig. 1: Temporal evolution of the discharge energy.

A sketch of the geometry is represented in Fig. 2. An unstructured mesh composed of triangular elements is used to discretize the 2D numerical domain. Two mesh refinements are used for the different

phases of the computation as much finer cells are required during the voltage pulse compared to the interpulse period. A minimum of 15 points inside the thermal flame thickness is ensured for both meshes thus justifying the DNS approach for the reactive mixture. The domain is extended up to 10 cm to evacuate properly the pressure wave generated by the discharge.

Property	Pulse	Interpulse
Δx_{fine}	3 µm	15 µm
$\Delta x_{\mathrm{medium}}$	10 µm	30 µm
$\Delta x_{\mathrm{coarse}}$	250 µm	250 µm
N_{nodes}	6.9×10^5	8.5×10^4

Table 1: Pulse and inter-pulse mesh properties, see Fig. 2 for location of mesh zones.

	1	2	3
z [mm]	0	0	1.75
r [mm]	0	1	0

Table 2: Locations of the three probes displayed in Fig. 2.

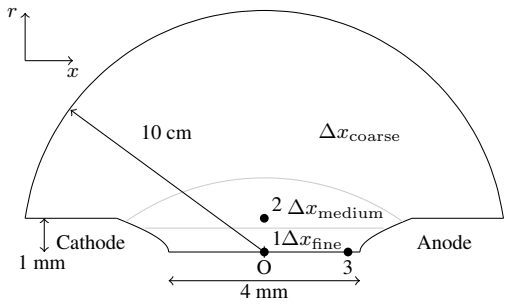


Fig. 2: Sketch of the geometry with mesh sizes and probes.

3.1. Boundary conditions

For the drift-diffusion equations, Neumann boundary conditions are applied on the electrode surfaces as in [7, 11], while an outlet is used for the far-field boundary and a symmetry condition is applied on the cylindrical axis.

A zero Neumann boundary condition is applied for the Poisson equation on the axis and the far field. A spatially constant Dirichlet boundary condition at potential V_a is used for the anode according the temporal profile shown in Fig. 1 while the cathode is grounded ($V_c = 0$).

Finally, for the Navier-Stokes equations, adiabatic non-slipping walls are considered on the electrodes, thus neglecting heat conduction through them. A non-reflecting NSCBC [27] boundary condition is applied on the far-field boundary. As for the remaining axis boundary, a symmetry is applied.

3.2. Chemistry

In previous work [28], a detailed chemical mechanism has been developed for plasma-assisted combustion of CH_4-Air mixtures based on the GRI 3.0 mechanism [29]. This mechanism has been extensively validated against various experiments including discharges in air [30] and a plasma-assisted ignition case [31]. Using the ARCANE library [32], the mechanism was reduced to a skeletal mechanism containing 44 species (O, O_2, H, OH, H_2, HO_2, H_2O_2, CH, CO, CH_2, HCO, CH_2(S) CH_3, CH_2O, CH_4, CO_2, CH_3O, CH_3OH, C_2H_4, C_2H_5, C_2H_6, H_2O, N_2, N, NO, NO_2, H_2CN, O_3, e^-, N_2^+, O_2^+, CH_4^+, CH_3^+, NO^+, O^-, O_2^-,N_2(A), N_2(B), N_2(a), N_2(C), N(^{2}D), $O_2(a_1)$, O(^{1}D), O(^{1}S)) and 410 reactions.

In addition to its validity for plasma discharge cases, the reduced mechanism is also able to reproduce correctly conventional auto-ignition delay times of methane-air mixtures in a large range of temperatures $[800 - 2000]$ K and equivalence ratios [0.5-1.5] as well as laminar flame speeds in a large range of equivalence ratios [0.6-1.4].

4. Results

In this work ignition using a single pulse (SP) and multiple pulses (MP) are studied to better understand their physical effects and their differences.

4.1. Single-pulse ignition

The capacity of non-equilibrium discharges to ignite a reactive mixture is first studied using a single pulse. The discharge energy E^p was varied from 200 µJ to 800 µJ and it was found that the minimum energy necessary to ignite the mixture is $E_{\mathrm{SP}}^{\mathrm{min}} = 500$ µJ. The 500 µJ-SP discharge is thus investigated in this section and shown in Fig. 3 (middle column).

Two-dimensional fields of electron density and discharge energy density at different instants obtained with this single-pulse can be found in Fig. 4. During the first nanoseconds, both positive and negative streamers are propagating towards the center and no significant energy is transmitted to the electrons yet as shown in Figs. 4a and 4b at 3 ns. Both streamers eventually connect to create a discharge channel around 4 ns. After connection, the electron density in the discharge channel increases significantly as does the discharge energy density which are shown at the cut-off time $t_{\mathrm{off}} = 14.7$ ns with a peak ionization degree of 10^{-3}. The discharge density is strongly peaked close to the electrodes inducing a significant amount of heat and radicals. These radicals, produced by the non-equilibrium discharge, are assumed to enhance combustion in several studies [2, 13].

To assess the importance of the chemical enhancement, the same ignition configuration with only energy deposition is also computed and referred to as the *heat deposit* (HD) case in which all the energy is assumed to be converted to heat using the same chemistry. An energy of $E_{\mathrm{HD}} = 500$ µJ is deposited in the form of an energy source term within

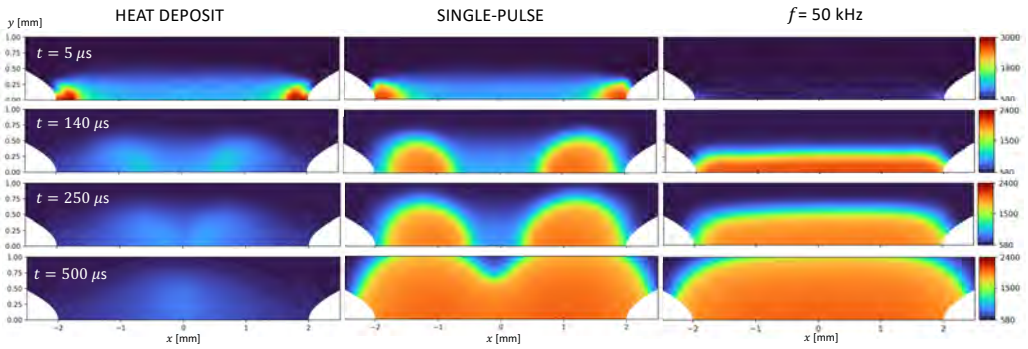
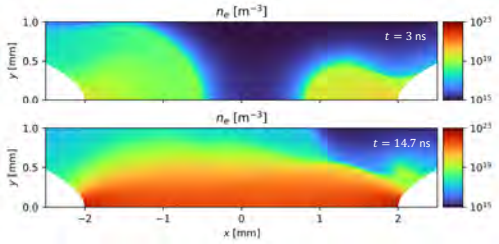
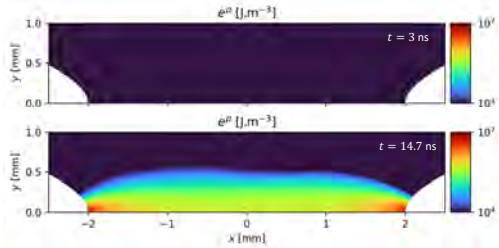


Fig. 3: Temperature fields in Kelvin at different instants for the HD, SP and 50 kHz-MP cases with an energy deposition of 500, 500 and 350 µJ respectively.



(a) Electron density



(b) Discharge energy density

Fig. 4: Two-dimensional fields for the single-pulse discharge at different instants.

$\tau_{\mathrm{HD}} = 20$ ns as in the discharge case. The spatial distribution of the energy deposition is determined from the normalized discharge energy density $\mathcal{F}(x,r) = e^p(x,r,t_{\mathrm{off}})/E^p(t_{\mathrm{off}})$ of the SP case so that the constant power source term, deposited during $[0, \tau_{\mathrm{HD}}]$, reads

$$\dot{Q}_{\mathrm{HD}}(x,r) = \frac{E_{\mathrm{HD}}}{\tau_{\mathrm{HD}}}\mathcal{F}(x,r) \qquad (6)$$

For $E_{\mathrm{HD}} = 500\,\mu$J, the mixture does not ignite while for the same energy using a plasma discharge it does. We need to increase the energy to $E_{\mathrm{HD}}^{\mathrm{min}} = 600\,\mu$J to observe ignition using only a heat deposit. The 600 µJ-HD and 500 µJ-SP ignitions are compared in Fig. 5 where selected species at probe 3, close to the anode, are shown. Radicals O, OH and H are produced directly by the plasma discharge in tens of nanoseconds in the SP case (blue curve) whereas it is a result of the temperature increase in the HD case (red curve) so that their peak value is lower and happening later. These radicals accelerate the ignition as the consumption of fuel and oxidizer starts earlier for the SP case compared to the HD case as shown in the right part of Fig. 5.

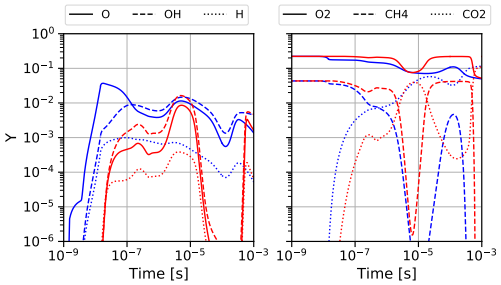


Fig. 5: Time evolution of selected species mass fractions in a single-pulse discharge with $E_{\mathrm{SP}} = 500\,\mu$J (blue), and in the heat deposit case with $E_{\mathrm{HD}} = 600\,\mu$J (red) at probe 3.

Shortly after the end of the pulse, two strong shocks are formed at electrode tips as shown in Fig. 6 and move outward in a cylindrical shape after merging at $\approx 3\,\mu$s. The expanding pressure wave contains part of the discharge energy which is then no more available for ignition. However the associated acoustic energy e_a defined in Eq. (7) is about 15 µJ for the SP case which represents less than 2% of the discharge energy. In the 500 µJ-HD case, a stronger shock wave is formed with a higher acoustic energy loss of about 35 µJ ($\approx 7\%$ of the discharge energy).

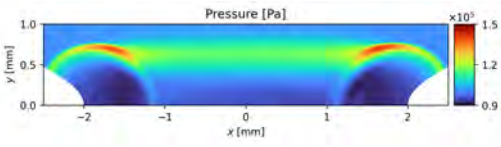


Fig. 6: Pressure field at $t = 1\,\mu$s for the 500 µJ-SP case.

$$e_{\mathrm{a}} = \frac{1}{2}\rho_0 \mathbf{u}_1^2 + \frac{1}{2}\frac{P_1^2}{\rho_0 c_0^2} \qquad (7)$$

A more important consequence of this shock wave is the induced flow close to the electrodes illustrated by Fig. 7. The two red spots correspond to the ignition kernels and the arrows indicate the flow direction: it is moving alongside the electrodes towards the center of the gap and leaves it in a radial upward motion. This gaseous flow has two effects: the initial flame kernels are convected towards the center of the inter-electrode gap and the regions around the electrode tips are cooled down. The gaseous flow effects can be seen in Fig. 5 around $t \simeq 10^{-4}$ s for the SP case (red) where both O_2 and CH_4 mass fractions are increasing close to the anode due the entry of fresh gases. A stronger shock in the 600 µJ-HD case fills the anode tip area (Probe 3) with fresh gases earlier than the SP case at $t \simeq 10\,\mu$s and ignition occurs later compared to the SP case even with a 20% energy increase.

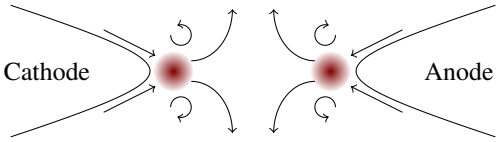


Fig. 7: Sketch of the flow around the ignition kernels (in red).

According to [33], a critical flame radius r_c, above which ignition is ensured, can be estimated under the unity Lewis number assumption as

$$r_c = \frac{\lambda(T_{\rm ad})}{C_p(T_{\rm ad})\,\rho(T_f)\,S_L^0} \tag{8}$$

where $T_{\rm ad}$ is the adiabatic flame temperature, T_f the fresh gas temperature, λ the heat conductivity, C_p the heat capacity at constant pressure and S_L^0 the laminar flame speed. In the present conditions for methane-air combustion at $\phi = 0.8$ ($T_f = 600\,$K, $T_{\rm ad} = 2200\,$K, $S_L^0 = 1.05\,{\rm m\,s^{-1}}$) we have $r_c = 150\,\mu$m.

5 µs after the end of the pulse, for both the 500 µJ-HD and SP cases two hot spots are formed at the electrode tips as shown in the first line of Fig. 3. At 140 µs, two isolated spherical kernels are formed in the SP case with an approximate radius of 400 µm. Thus, it is very likely that the flame kernels will continue to grow and expand which is confirmed by the subsequent snapshots as the flame kernels merge and start to expand radially as shown in the middle column of Fig. 3.

On the other hand, as stated in the beginning of the section, the 500 µJ-HD case does not ignite as shown by the left column of Fig. 3 due to the stronger shock-wave and recirculation. Increasing the deposited energy to 600 µJ allows *one* flame kernel, at the anode, to grow and travel towards the center of the inter-electrode zone. But to obtain a growing kernel comparable to that of the plasma discharge case, the deposited energy must be increased to 700 µJ, *i.e.*, 40% more than the plasma discharge.

4.2. Multi-pulse effect

Multiple discharges at various frequencies are now investigated to ignite the methane-air mixture. The energy per pulse is decreased tenfold compared to the single-pulse at $E_p = 50\,\mu$J and the frequency is varied from 10 kHz to 100 kHz.

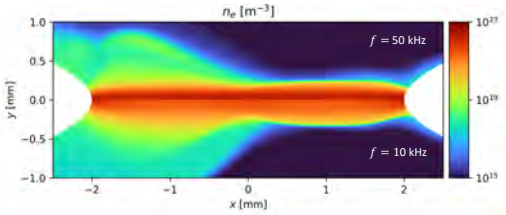


Fig. 8: Electron density at the end of the 5th pulse for $f = 50\,$kHz (top) and $f = 10\,$kHz (bottom).

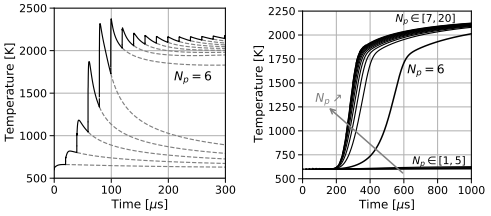
To evaluate the impact of the pulse frequency on ignition, the gas temperature signal at probe 2 is used to define the ignition delay time. Probe 2, located 1 mm above the axis, is not directly reached by the plasma discharges and thus may be used to indicate the possible passage of a growing flame kernel. For all the studied frequencies f, 20 pulses were applied in order to deposit a total energy of 1 mJ. After each burst of $N_p \in [1, 20]$ pulses, an additional simulation was continued without further discharges up to 1 ms. Figure 9a shows the time evolution of the gas temperature at probe 1 for the first 8 pulses. At the end of each pulse, the gas temperature evolution without further plasma discharges (grey-lines) shows that 6 pulses are necessary to initiate a flame kernel close to the axis. This initial kernel is able to self-propagate to a radius of 1 mm as indicated by the temperature signal at probe 2 in Fig. 9b. As a result, a burst of low-energy discharges is able to ignite the mixture with a total energy of only $E_{\rm MP}^{\rm min} = 300\,\mu$J, i.e., 40% lower than the minimum energy required for ignition with a single pulse discharge as seen in Section 4.

Increasing further the number of pulses then leads to faster ignition, until an asymptotic limit is reached as shown in Fig. 9b. This behavior was observed for all frequencies in the range of [20-100] kHz as shown in Fig. 10 where the ignition delay time is defined at probe 2 by Eq. (9) where $\Delta T = 1000\,$K.

$$\tau_{\rm ig} = \min(t \mid T(t) \geq T_0 + \Delta T) \tag{9}$$

Note that for the 25 kHz case, the minimum energy required for ignition is equivalent to that of the SP-ignition, thus limiting the benefit of MP discharges. For lower frequencies (i.e., $f \leq 25\,$kHz), a detrimental effect is even observed as the 20 kHz case requires a total energy of 750 µJ for a successful ignition.

Moreover, it was observed that for frequencies lower than 15 kHz, no ignition event occurs before 1 ms. At low frequency, the cumulative effects of the successive discharges are gradually lost mostly due to the diffusion of temperature and species during the inter-pulse period. Conversely at high fre-



(a) Probe 1 (b) Probe 2

Fig. 9: Gas temperature evolution at $f = 50\,\mathrm{kHz}$ in (a) the plasma region and (b) at 1mm height.

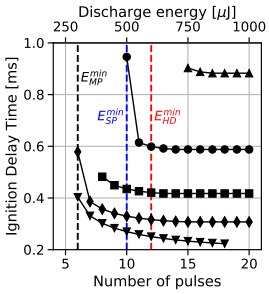


Fig. 10: Ignition delay time depending on the number of pulses at various frequencies: ▲ 20 kHz ● 25 kHz ■ 33 kHz ◆ 50 kHz ▼ 100 kHz. The corresponding total discharge energy is indicated on the top axis of the figure. $E_{\mathrm{HD}}^{\mathrm{min}}$, $E_{\mathrm{SP}}^{\mathrm{min}}$ and $E_{\mathrm{MP}}^{\mathrm{min}}$ correspond respectively to the minimum energy ignition of the HD, SP and MP cases.

quency, each discharge takes advantage of the previous one and concentrates on the axis as illustrated in Fig. 8. The discharge radius, defined with the Full-Width at Half Maximum of the $N_2(B_3)$ peak as in [30], is used to illustrate the cumulative effect of the discharges. Figure 11 shows the discharge radius for the 10 and 50 kHz cases at the end of each pulse, together with the gas temperature just before the pulse at the center of the inter-electrode gap. At 10 kHz, the discharge radius first decreases to 125 µm before slowly increasing at the same rate as the gas temperature. Indeed, electron and ion diffusion coefficients are increasing with temperature leading to a wider discharge propagation and stronger diffusion of the plasma channel during the inter-pulse. At 50 kHz, a sharp decrease of the discharge radius is observed during the first four pulses allowing to concentrate the energy on the axis. For the same amount of energy deposited, smaller volumes lead to higher gas temperatures and radical production allowing to trigger favourable conditions for flame kernel initiation. In the same time, the gas temperature increases drastically and marks the early stage of flame kernel formation. Then, as for the 10 kHz case, the discharge radius increases with temperature and the discharges lead to smaller temperature increase, as was observed for the 7th and later pulses in Fig. 9a.

5. Conclusions

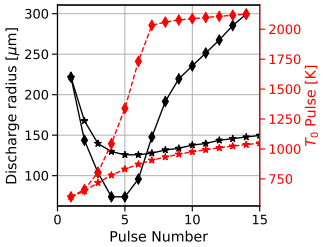


Fig. 11: Evolution of discharge radius and local gas temperature preceding each pulse (probe 1) at two different frequencies: ★ 10 kHz ◆ 50 kHz

Ignition of a lean methane air-mixture has been carried out using the coupled AVIP-AVBP codes with a reduced plasma-assisted combustion chemistry in a pin-pin configuration. Fully coupled plasma and combustion simulations have allowed to investigate the interaction between the plasma discharge and the reacting flow.

Using a single-pulse discharge of 500 µJ, the formation of two ignition kernels were observed at the end of the discharge close to the electrode tips. Along these two kernels a strong shock wave was formed inducing a recirculation of fresh gases into the inter-electrode gap and pushing the two kernels towards the center of the gap. The two kernels eventually merge to form a stable single kernel propagating radially outwards. The chemical effect of the discharge has been highlighted by simulating a similar discharge with energy deposited only as an energy source term. For the same amouny of deposited energy of 500 µJ, no ignition was observed as the kernels were quenched by the fresh gases recirculation induced by a stronger shock-wave.

The single pulse discharge ignition was then compared with low-energy (50 µJ), multiple pulses, applied at various frequencies from 10 to 100 kHz. At low frequency, the created spots of temperature have sufficient time to diffuse and below 15 kHz, no ignition is observed. Increasing the frequency above 25 kHz has a beneficial effect on ignition exploiting the synergetic effect of the discharges: the ignition delay time decreases and the minimum energy needed for successful ignition decreases. Indeed at high frequency, the plasma channel does not have time to diffuse significantly and narrower discharges can be produced, thus focusing both heating and radical production on the axis. This shows the benefits of applying smaller energy pulses repetitively as they are able to ignite faster and in a more stable way combustible mixtures.

In this study, using coupled plasma-combustion models has proved very efficient to investigate open questions about the impact of plasma chemistry and the interaction of the plasma with the flow. The coupled model may now be applied to more complex configurations to predict the impact of NRP discharges on flames.

6. Acknowledgments

This work was supported by the ANR projects PASTEC (ANR-16-CE22-0005) and GECCO (ANR-17-CE06-0019). This work was performed using HPC resources from GENCI-TGCC (Grant 2021-A0092B10157). The authors thank Deanna Lacoste for fruitful discussions on NRP discharges experiments and Gerjan Hagelaar for the use of BOLSIG+ inside AVIP.

References

[1] A. Starikovskiy, N. Aleksandrov, Plasma-assisted ignition and combustion, Prog. Energy Combust. Sci 39 (1) (2013) 61 – 110.

[2] Y. Ju, W. Sun, Plasma assisted combustion: Dynamics and chemistry, Prog. Energy Combust. Sci 48 (2015) 21–83.

[3] S. Starikovskaia, D. A. Lacoste, G. Colonna, Non-equilibrium plasma for ignition and combustion enhancement, Eur. Phys. J. D 75 (8) (2021).

[4] M. S. Bak, H. Do, M. G. Mungal, M. A. Cappelli, Plasma-assisted stabilization of laminar premixed methane/air flames around the lean flammability limit, Combust. Flame 159 (10) (2012) 3128 – 3137.

[5] C. D. Cathey, T. Tang, T. Shiraishi, T. Urushihara, A. Kuthi, M. A. Gundersen, Nanosecond plasma ignition for improved performance of an internal combustion engine, IEEE Trans. Plasma Sci 35 (6) (2007) 1664–1668.

[6] S. Bozhenkov, S. Starikovskaia, A. Starikovskii, Nanosecond gas discharge ignition of H_2 and CH_4 containing mixtures, Combust. Flame 133 (1) (2003) 133 – 146.

[7] S. Célestin, Study of the dynamics of streamers in air at atmospheric pressure, Ph.D. thesis, Ecole centrale de Paris (2008).

[8] T. Poinsot, D. Veynante, Theoretical and Numerical Combustion, R.T. Edwards Inc., 2005.

[9] N. A. Popov, Kinetics of plasma-assisted combustion: effect of non-equilibrium excitation on the ignition and oxidation of combustible mixtures, Plasma Sources Sci. Technol. 25 (4) (2016) 043002.

[10] D. Breden, L. L. Raja, C. A. Idicheria, P. M. Najt, S. Mahadevan, A numerical study of high-pressure non-equilibrium streamers for combustion ignition application, J. Appl. Phys 114 (8) (2013) 083302.

[11] F. Tholin, Numerical simulation of nanosecond repetitively pulsed discharges in air at atmospheric pressure : Application to plasma-assisted combustion, Ph.D. thesis, Ecole centrale de Paris (2012).

[12] A. Sharma, V. Subramaniam, E. Solmaz, L. L. Raja, Coupled computational studies of non-thermal plasma based combustion ignition.

[13] M. L. Gracio Bilro Castela, Direct Numerical Simulations of plasma-assisted ignition in quiescent and turbulent flow conditions, Theses, Université Paris-Saclay (2016).

[14] H. Sitaraman, R. Grout, Premixed combustion simulations with a self-consistent plasma model for initiation, 54th AIAA (2016).

[15] S. Yang, S. Nagaraja, W. Sun, V. Yang, Multi-scale modeling and general theory of non-equilibrium plasma-assisted ignition and combustion, J. Phys. D: Appl. Phys 50 (43) (2017) 433001.

[16] S. Kobayashi, Z. Bonaventura, F. Tholin, N. A. Popov, A. Bourdon, Study of nanosecond discharges in H_2-air mixtures at atmospheric pressure for plasma assisted combustion applications, Plasma Sources Sci. Technol. 26 (7) (2017) 075004.

[17] L. Cheng, N. Barleon, O. Vermorel, B. Cuenot, A. Bourdon, Avip: a low temperature plasma code, arXiv preprint arXiv (2021).

[18] B. Bagheri, J. Teunissen, U. Ebert, M. M. Becker, S. Chen, O. Ducasse, O. Eichwald, D. Loffhagen, A. Luque, D. Mihailova, J. M. Plewa, J. van Dijk, M. Yousfi, Comparison of six simulation codes for positive streamers in air, Plasma Sources Sci. Technol. 27 (9) (2018) 095002.

[19] T. Schonfeld, M. Rudgyard, Steady and unsteady flow simulations using the hybrid flow solver AVBP, AIAA 37 (11) (1999) 1378–1385.

[20] E. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, 2009.

[21] G. J. M. Hagelaar, L. C. Pitchford, Solving the boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models, Plasma Sources Sci. Technol. 14 (4) (2005) 722–733.

[22] R. Morrow, J. J. Lowke, Streamer propagation in air, J. Phys. D: Appl. Phys 30 (4) (1997) 614–627.

[23] S. Balay, W. D. Gropp, L. C. McInnes, B. F. Smith, Efficient management of parallelism in object oriented numerical software libraries, in: E. Arge, A. M. Bruaset, H. P. Langtangen (Eds.), Modern Software Tools in Scientific Computing, Birkhäuser Press, 1997, pp. 163–202.

[24] N. Popov, Fast gas heating initiated by pulsed nanosecond discharge in atmospheric pressure air, 51st AIAA (2013).

[25] M. Capitelli, F. C.M., B. Gordiets, A. Osipov, Plasma Kinetics in Atmospheric Gases, Springer, 2000.

[26] R. C. Millikan, D. R. White, Systematics of vibrational relaxation, J. Chem. Phys 39 (12) (1963) 3209–3213.

[27] T. Poinsot, S. Lelef, Boundary conditions for direct simulations of compressible viscous flows, J. Comput. Phys 101 (1) (1992) 104 – 129.

[28] L. Cheng, N. Barleon, B. Cuenot, O. Vermorel, A. Bourdon, Plasma assisted combustion of methane-air mixtures: Validation and reduction, Under review at Combust. Flame.

[29] `http://www.me.berkeley.edu/gri_mech/`.

[30] D. Rusterholtz, D. Lacoste, G. Stancu, D. Pai, C. Laux, Ultrafast heating and oxygen dissociation in atmospheric pressure air by nanosecond repetitively pulsed discharges, . Phys. D: Appl. Phys 46 (2013) 4010–.

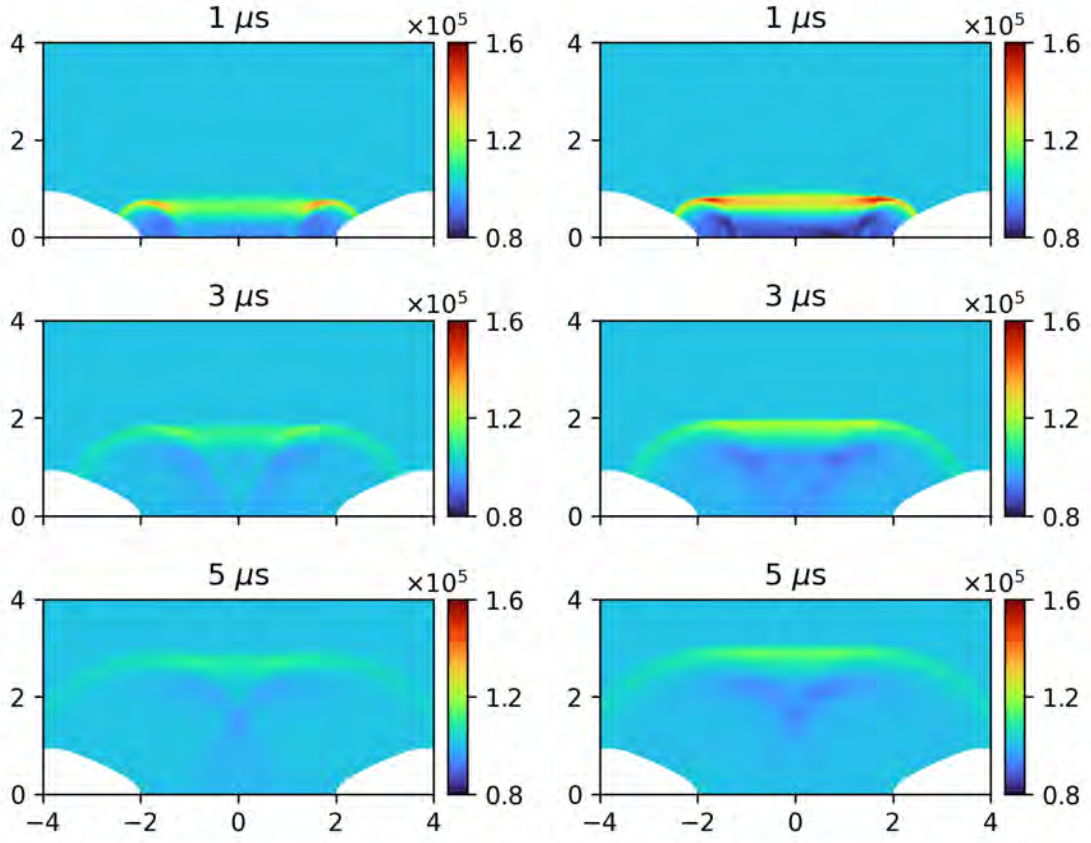
[31] N. L. Aleksandrov, S. V. Kindysheva, E. N. Kukaev, S. M. Starikovskaya, A. Y. Starikovskii, Simulation of the ignition of a methane-air mixture by a high-voltage nanosecond discharge, Plasma Phys. Rep. 35 (10) (2009) 867–882.

[32] Q. Cazères, P. Pepiot, E. Riber, B. Cuenot, A fully automatic procedure for the analytical reduction of chemical kinetics mechanisms for computational fluid dynamics applications, Fuel 303 (2021) 121247.

[33] M. Champion, B. Deshaies, G. Joulin, K. Kinoshita, Spherical flame initiation: Theory versus experiments for lean propane-air mixtures, Combust. Flame 65 (3) (1986) 319–337.

11.3.1 Supplementary material

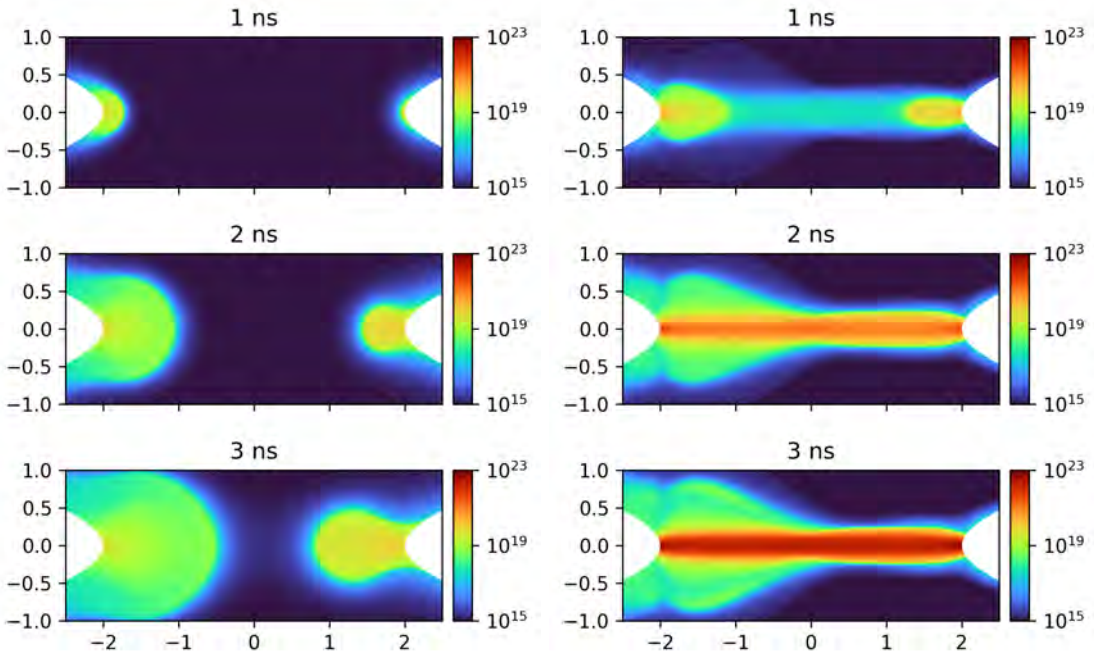
Some snapshots of the different fields that are not shown in the article, due to lack of space, are given here. The pressure wave from the 500 µJ-SP (single pulse) and 500 µJ-HD (heat deposit) cases are compared in Fig. 11.1 where the stronger pressure wave induced by the HD case can be appreciated. At $t = 1\,\mu s$ the peak of pressure of the shock wave is about 10% higher for the heat deposit case and this high pressure zone is distributed more broadly on the shock front. Correspondingly, the value of the pressure in the rarefaction zone is lower by 10% for the full-heat case compared the single-pulse case. Finally at $t = 5\,\mu s$ the shock-wave is thicker (green region) while being faster.



(a) Single-pulse (b) Heat-deposit

Figure 11.1: Pressure field at different instants for the single-pulse and full-heat cases.

The benefits of multiple discharges for the creation of a conducting channel is discussed in the article. This allows to more easily propagate discharges as evidenced by Fig. 11.2. At $t = 1\,ns$, the 5th 50kHz-pulse streamers have already propagated 1 mm at both sides whereas the single-pulse streamers have barely moved. At $t = 2\,ns$, the channel is already well-established for the 5th 50kHz-pulse whereas the two streamers have yet to bridge the gap for the single pulse discharge. At $t = 3\,ns$, the streamers are still propagating for the single-pulse while the conducting channel already had time to significantly increase the electron density in the 5th 50kHz-pulse.



(a) Single-pulse

(b) 50 kHZ 5th pulse

Figure 11.2: Electron density field at different instants for the single-pulse and the 50 khz 5th pulse cases.

This easier propagation of discharges after multiple pulses is related to the residual electron density channel that remains at the end of the interpulse. When the voltage is switched-off, electrons diffuse and recombine as shown in Fig. 11.3.

11.3.2 High-energy heat deposit ignition

A limit case highlighting the influence of the hydrodynamic effects is considered in this section to complement the heat deposit case of the article. The same configuration and mesh as that of the article is considered. We choose to deposit energy as only a heating source term just like in the heat deposit but this time the energy deposited is increased by a factor of 5 to 2.5 mJ. The duration of constant energy deposition is still 20 ns as the article heat deposit case. The shape of the energy density is simplified compared to the article heat deposit and is shown in Fig. 11.4: the energy density is constant axially and a sigmoid profile is applied on the radial direction.

During the first instants after the energy deposition, a strong shock-wave is formed with an over-pressure of around 10 bar as shown in Fig. 11.5. The temperature exceeds 5000 K in the gap during the first instants but quickly decreases as the shock-wave leaves the domain. A rarefaction wave, right behind the shock-wave, creates a region of low density (blue region in the density snapshots).

During a second phase the mixture ignites as evidenced by the heat release shown in Fig. 11.6. The flow induced by the shock-wave has an impact clearly shown on the density and temperature fields: fresh gases are entering the ignition kernel from both sides pushing

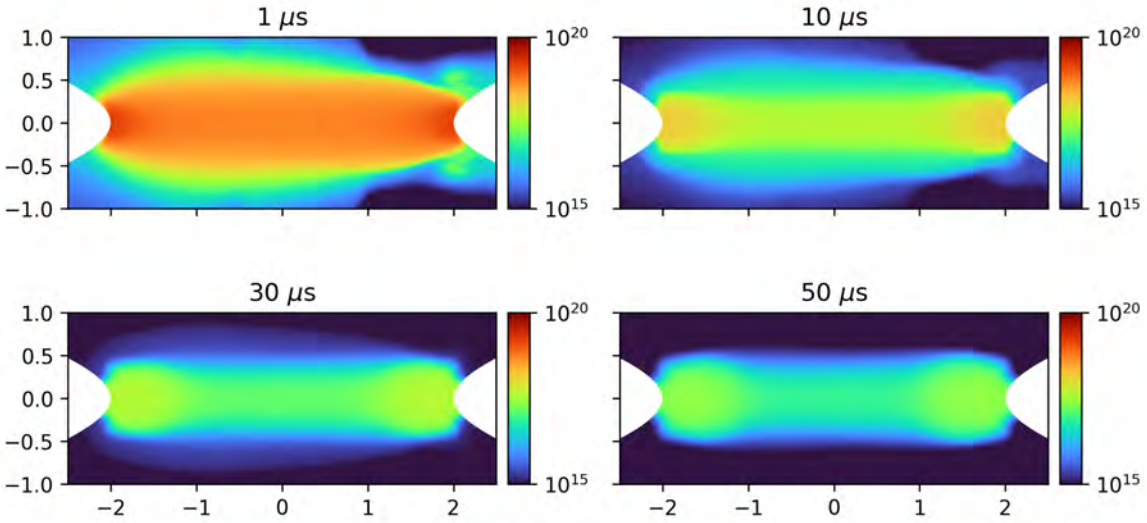


Figure 11.3: Electron density [m^{-3}] field at different instants during the interpulse of the single-pulse case.

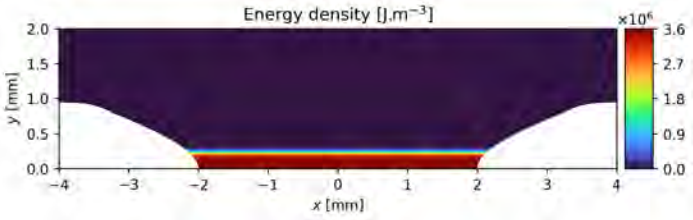
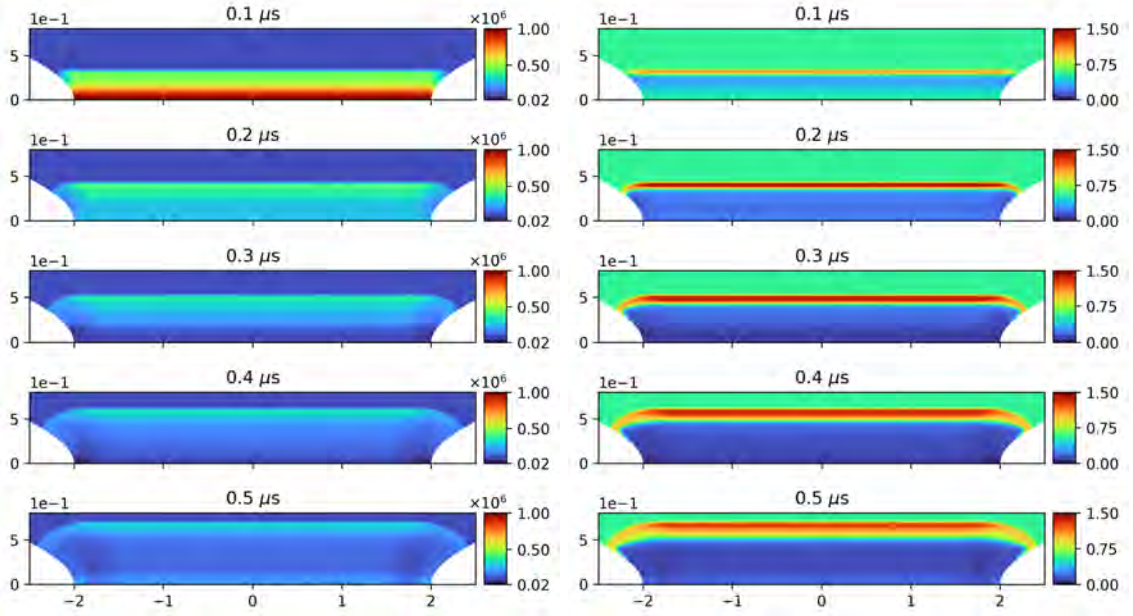
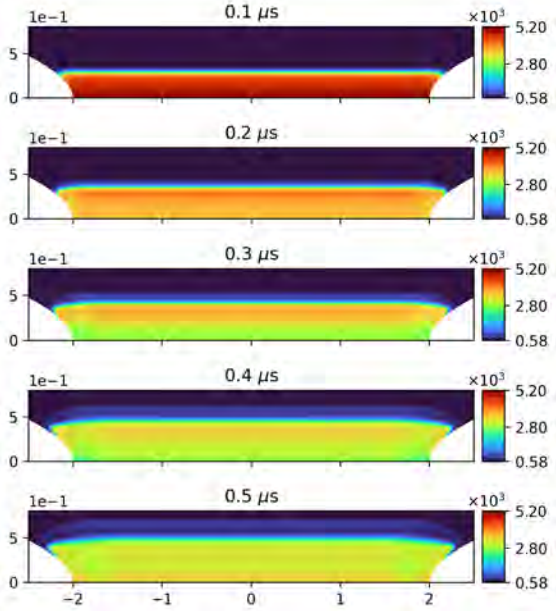


Figure 11.4: Energy density deposit shape for the high-energy deposit ignition.



(a) Pressure [Pa]

(b) Density [m^{-3}]

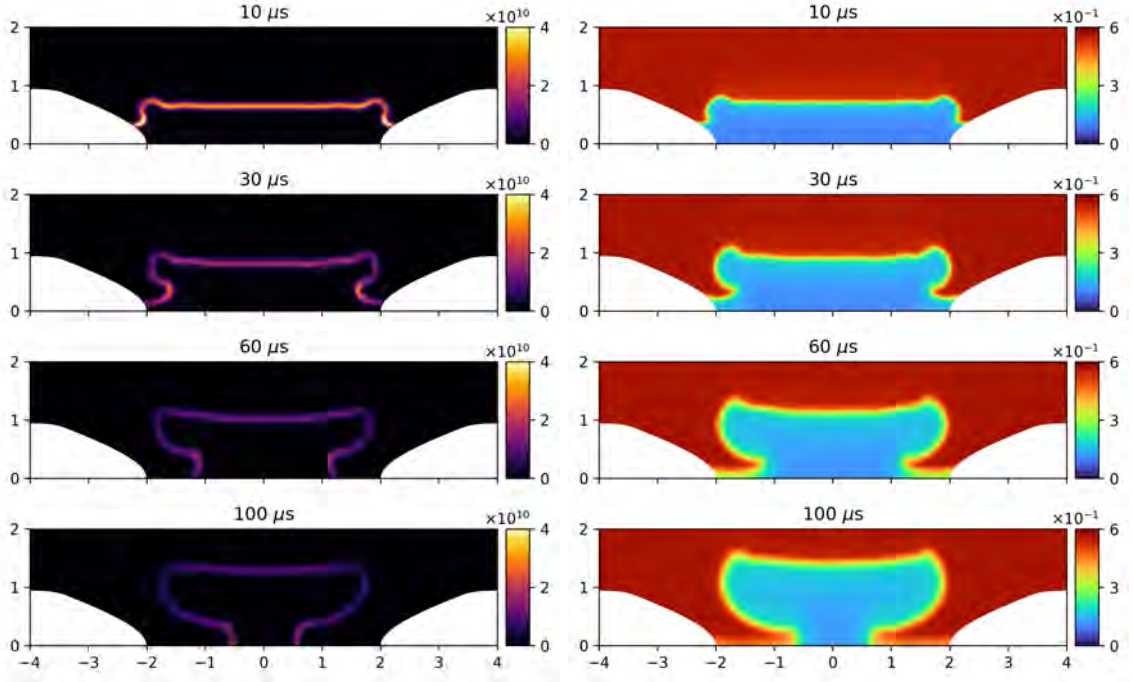


(c) Temperature [K]

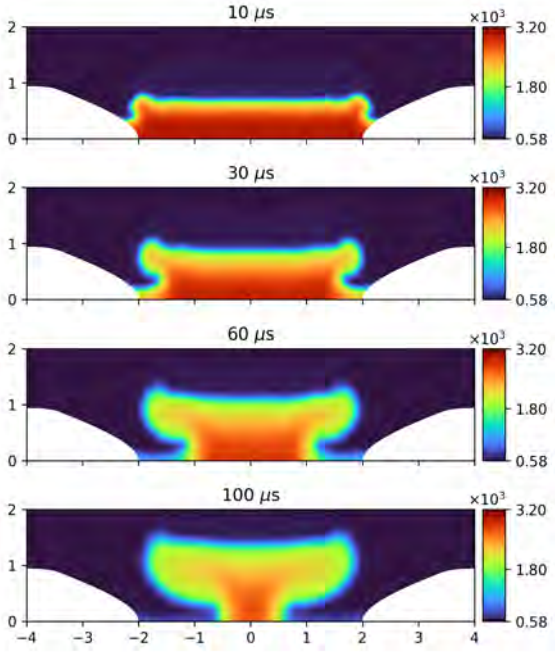
Figure 11.5: Snapshots of pressure, density and temperature during the first instants of the high-energy deposit case.

the ignition kernel radially outwards. Note that the temperature of the ignition kernel is significantly decreased due to this entry of fresh gases.

Finally in a third phase shown in Fig. 11.7, the flame lifts from the axis to get a toroïdal shape: the ignition kernel completely extends radially in a closed tore shape due to the flow recirculation induced by the initial shock wave. This case illustrates the extreme case of a flame completely lifted from the axis and should be avoided for efficient ignition. Similar kind of flames have already been observed in Gracio Bilro Castela [2016] and Dumitrache et al. [2019].

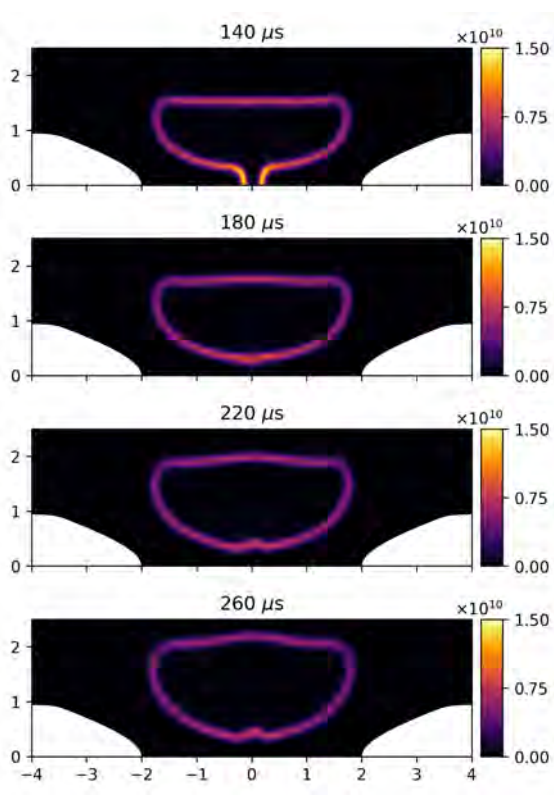


(a) Heat-release [W.m^{-3}] (b) Density [m^{-3}]

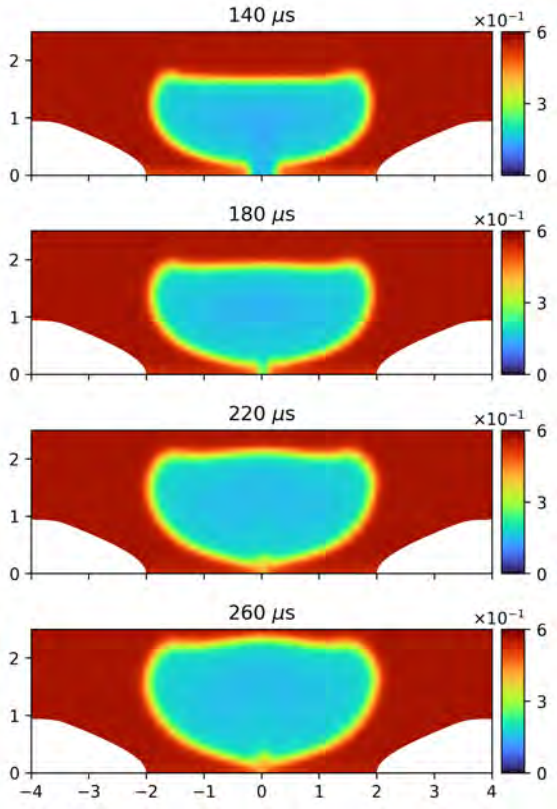


(c) Temperature [K]

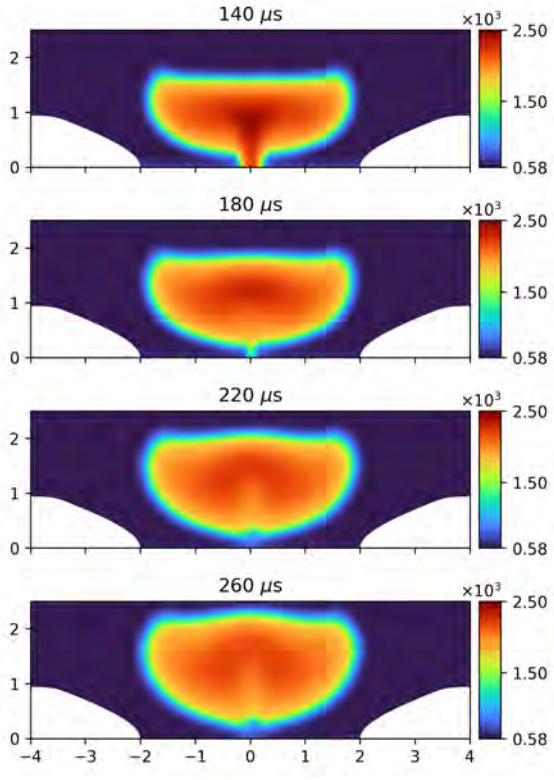
Figure 11.6: Snapshots of heat release, density and temperature during phase 2 of the high-energy deposit case.



(a) Heat-release [W.m^{-3}]



(b) Density [m^{-3}]



(c) Temperature [K]

Figure 11.7: Snapshots of heat release, density and temperature during phase 3 of the high-energy deposit case.

Chapter 12

Building a phenomenological model for PAC

Contents

In this chapter an improved model for Plasma Assisted Combustion, based on the model developed in Gracio Bilro Castela [2016], is discussed. Detailed modeling, as done in the previous Chap. 11, offers lots of insights about the plasma assisted combustion mechanisms but is very costly. The reduced cost of this simulation also relies on the axisymmetric configuration studied, which allows to simulate a three-dimensional domain on a two-dimensional grid. This is not possible on real configurations as turbulence is present and breaks this invariance. Hence a model for plasma assisted combustion is needed. The Castela model is first recalled in Section. 12.1. Then improvements of this model are discussed that could tackle the shortcomings of the Castela model. Lack of time did not allow to test the model on real configurations but could be the subject of future work.

12.1 The Castela model

12.1.1 Governing equations

In place of a detailed description of the plasma discharge which would leads to a high CPU cost, a phenomenological model has been proposed by Gracio Bilro Castela [2016]. In this model the impact of the discharge on the energy of the mixture is decomposed into three effects which are namely the fast gas heating ($\dot{E}^p_{\text{heat}}$), the chemical effect ($\dot{E}^p_{\text{chem}}$) and a slow gas heating due to the relaxation of the vibrationally excited N_2 ($\dot{E}^p_{\text{vib}}$). The discharge power can be written as the sum of these three contributions:

$$\dot{E}^p = \dot{E}^p_{\text{chem}} + \dot{E}^p_{\text{heat}} + \dot{E}^p_{\text{vib}} \tag{12.1}$$

The Navier-Stokes equations with additional plasma source terms then yield:

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \nabla \cdot [\rho \mathbf{u}] = 0 \\
&\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot [\rho \mathbf{u}\mathbf{u} + p\mathbf{I} - \boldsymbol{\tau}] = 0 \\
&\frac{\partial \rho e_t}{\partial t} + \nabla \cdot [(\rho e_t + p)\mathbf{u} - \boldsymbol{\tau} \cdot \mathbf{u} + \mathbf{q}] = \dot{E}^p_{\text{chem}} + \dot{E}^p_{\text{heat}} + \dot{R}^p_{VT} \\
&\frac{\partial \rho Y_k}{\partial t} + \nabla \cdot [\rho Y_k (\mathbf{u} + \mathbf{V}_k)] = \dot{\omega}^c_k + \dot{\omega}^p_k
\end{aligned} \tag{12.2}$$

where $\dot{\omega}^p_k$ corresponds to the mass production rate associated to the chemical effect of the discharge and $\dot{R}^p_{VT}$ corresponds to the vibrational-translational relaxation of N_2. Note that here e_t is the total energy which is different from the total non chemical energy equation solved in AVIP. As the typical timescale of vibrational-translational relaxation is much higher than the discharge duration, it is necessary to store it into a vibrational energy density e_{vib} which can be transported and relaxed according to:

$$\frac{\partial \rho e_{\text{vib}}}{\partial t} + \nabla \cdot [\rho e_{\text{vib}}\, \mathbf{u}] = \nabla \cdot [\rho \mathcal{D}_{N_2} \nabla e_{\text{vib}}] + \dot{E}^p_{\text{vib}} - \dot{R}^p_{VT} \tag{12.3}$$

where $\mathcal{D}_{N_2}$ is the diffusion coefficient of N_2 into the mixture, $\dot{E}^p_{\text{vib}}$ is the vibrational energy production term computed from plasma chemistry and $\dot{R}^p_{VT}$ is the vibrational to translational relaxation modeled using the Landau-Teller harmonic oscillator approach as expressed in Eq. 12.4. Relaxation times τ^k_{VT} are computed using experimental coefficients in Capitelli et al. [2000] for H_2O, CO_2 and in Millikan and White [1963] for N_2, O_2, O. The exchange term between the vibrational energy equation and mixture energy equation then reads:

$$\dot{R}^p_{VT} = \rho \frac{e_{\text{vib}} - e^{\text{eq}}_{\text{vib}}(T)}{\tau_{VT}} \;,\; \tau_{VT} = \left(\sum_k \frac{1}{\tau^k_{VT}} \right)^{-1} \tag{12.4}$$

where

$$e^{\text{eq}}_{\text{vib}}(T) = \frac{r\Theta_1}{e^{\Theta_1/T} - 1} \tag{12.5}$$

is the equilibrium energy of vibrational population of N_2 assuminga harmonic oscillator model with the characteristic vibrational temperature $\Theta_1 = 3386$ K and $r = R/W_{N_2}$.

12.1.2 Model closure

In order to close the model it is necessary to determine the fraction of discharge energy going into $\dot{E}^p_{\text{chem}}$, $\dot{E}^p_{\text{heat}}$ and $\dot{E}^p_{\text{vib}}$. This repartition depends on different parameters such as gas temperature, pressure and composition (T, P, X_i). Other parameters such as the pulse shape or the geometry of the electrodes can also play a role. By considering only the parameters (T, P, X_i) the fraction of the discharge energy going into $\dot{E}^p_{\text{chem}}$ and $\dot{E}^p_{\text{heat}}$ can be written as follows :

$$\dot{E}^p_{\text{heat}} = g_{\text{heat}}\left(X_{k=1,\dots,N_{sp}}, T, P\right)\dot{E}^p \tag{12.6}$$

$$\dot{E}^p_{\text{chem}} = g_{\text{chem}}\left(X_{k=1,\dots,N_{sp}}, T, P\right)\dot{E}^p \tag{12.7}$$

where g_{heat} and g_{chem} are the fractions of the discharge energy transferred into sensible and chemical energy, respectively. The chemical energy source term should compensate the change in sensible energy due to the production of neutral species from the plasma discharge:

$$\dot{E}^p_{\text{chem}} = \sum_k e_k W_k \dot{w}^p_k \tag{12.8}$$

A fraction of the discharge power can be associated to each species with the following relation:

$$e_k W_k \dot{w}^p_k = g^k_{\text{chem}} \dot{E}^p \tag{12.9}$$

and consequently:

$$g_{\text{chem}} = \sum_k g^k_{\text{chem}}. \tag{12.10}$$

The production rate of the k^{th} species during the pulse is given by :

$$\dot{w}^p_k = g^k_{\text{chem}} \frac{\dot{E}^p}{e_k W_k} \tag{12.11}$$

Finally the model states that the rest of the energy that is not transferred into $\dot{E}^p_{\text{chem}}$ and $\dot{E}^p_{\text{heat}}$ goes into vibration:

$$\dot{E}^p_{\text{vib}} = [1 - (g_{\text{heat}} + g_{\text{chem}})]\dot{E}^p \tag{12.12}$$

It is supposed that fast gas heating and chemical effect are mainly due to the dissociation of O_2 in electron impact reaction of quenching of electronically excited N_2. Following the experiments from Rusterholtz et al. [2013] and simulations made by Popov [2013] about 35 ± 5 % of the discharge energy goes into O_2 dissociation and 20 ± 5 % into ultra fast gas heating where this value of $g_{\text{heat}} = 0.2$ is obtained from the temperature increase induced by the discharge. Consequently in the model approximately 45 % of the energy is stored in vibrational modes of N_2.

It is assumed that the chemical source term is only due to the dissociation of O_2 and scaled with $Y_{O_2}/Y^f_{O_2}$ where $Y^f_{O_2}$ is the dioxygen mass fraction in the fresh gas. The fraction of energy η going to O_2 dissociation is taken equal to 35%. $\dot{E}^p_{\text{chem}}$ writes finally:

$$\dot{E}^p_{\text{chem}} = \eta \frac{Y_{O2}}{Y^f_{O2}} \left(1 - \frac{e_{O_2}}{e_O}\right) \dot{E}^p \tag{12.13}$$

In order to retrieve the energy available for ultra fast gas heating another assumption is made by assuming that the sum of $\dot{E}^p_{\text{heat}}$ and $\dot{E}^p_{\text{chem}}$ is equal to the energy transferred to the electronic states of N_2:

$$\dot{E}^p_{\text{heat}} + \dot{E}^p_{\text{chem}} = \alpha \dot{E}^p \tag{12.14}$$

with $\alpha = 0.55$. Finally the energy going into fast gas heating is retrieved as well as the vibrational energy which is assumed to be the remaining energy:

$$\dot{E}^p_{\text{heat}} = \left[\alpha - \eta \frac{Y_{O2}}{Y^f_{O2}} \left(1 - \frac{e_{O_2}}{e_O}\right)\right] \dot{E}^p \tag{12.15}$$

$$\dot{E}^p_{\text{vib}} = (1-\alpha)\dot{E}^p \tag{12.16}$$

The model closure can be summarized by the Eqs. (12.13), (12.15) and (12.16).

12.2 Generalization of the model

12.2.1 Improved chemical source term

In the model developed above only one chemical plasma process is considered which is the dissociation of O_2:

$$O_2 \longrightarrow 2\,O \tag{12.17}$$

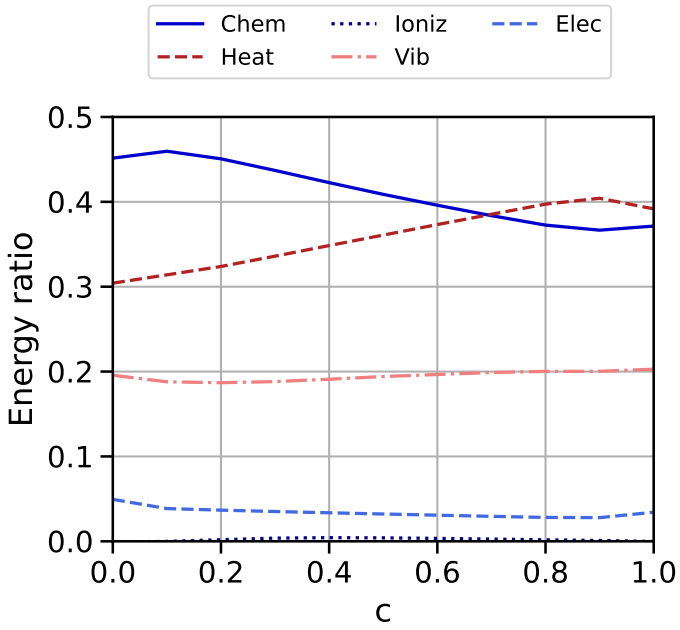


Figure 12.1: Ratios of energy of a single discharge in a stochiometric methane air-mixture at different compositions as a function of the progress variable $c = (Y_{\text{CO}} + Y_{\text{CO2}})/(Y^b_{\text{CO}} + Y^b_{\text{CO2}})$ where the superscript b correspond to burnt gases.

Note that this dissociation is due mainly to electron-impact reactions and quenching of nitrogen excited states and it has been compacted in one global reaction without the electrons

for this reason. This may be complemented thanks to the 0D computations performed in Chap. 9 showing that more processes coming from the plasma chemistry can have an impact on combustion. Moreover the parameters of the Castela model have only been validated in air and not in combustible mixtures: in burnt gases for example $Y_{O2} \simeq 0$ so that the chemical effect of O_2 dissociation would tend to zero in these zones. A burnt gases chemistry is currently under development but the first results of the energy paths as a function of the progress varible $c = (Y_{\mathrm{CO}} + Y_{\mathrm{CO2}})/(Y^b_{\mathrm{CO}} + Y^b_{\mathrm{CO2}})$ is shown in Fig. 12.1. The chemical source term in the burnt gases region is decreasing as c increases but is clearly non-zero. The chemical source term is thus generalized and we consider P plasma processes:

$$\dot{E}^p_{\mathrm{chem}} = \sum_{j=1}^{P} \dot{Q}_j \Delta e^m_j \tag{12.18}$$

where $\dot{Q}_j$ is the molar production rate of the plasma process j and Δe^m_j its corresponding change of internal energy. From the 0D computations the following list of plasma processes for the generalization of the model seem to synthesize the chemical processes of our chemistry (as before the dissociation may be due to quenching of excited states or electron-impact so that only the simplified process is written):

$$\mathrm{O_2} \longrightarrow 2\,\mathrm{O} \tag{12.19}$$

$$\mathrm{N_2} \longrightarrow 2\,\mathrm{N} \tag{12.20}$$

$$\frac{1}{2}\,\mathrm{N_2} + \mathrm{O_2} \longrightarrow \mathrm{NO} + \mathrm{O} \tag{12.21}$$

$$\mathrm{CH_4} \longrightarrow \mathrm{CH_3} + \mathrm{H} \tag{12.22}$$

$$\mathrm{CO_2} \longrightarrow \mathrm{CO} + \mathrm{O} \tag{12.23}$$

$$\mathrm{H_2O} \longrightarrow \mathrm{OH} + \mathrm{H} \tag{12.24}$$

$$\mathrm{CH_4} + \frac{1}{2}\,\mathrm{O_2} \longrightarrow \mathrm{CH_3} + \mathrm{OH} \tag{12.25}$$

$$\mathrm{CO_2} + \frac{1}{2}\,\mathrm{O_2} \longrightarrow \mathrm{CO} + \mathrm{O_2} \tag{12.26}$$

We leverage the zero-dimensional reactors to tabulate as a function of the progress variable c across a 1D flame the proportion of energy going into each channel:

$$\dot{E}^p_{\mathrm{chem}} = \sum_j \alpha_j \dot{E}^p \quad \dot{E}^p_{\mathrm{heat}} = \alpha_{\mathrm{heat}} \dot{E}^p \quad \dot{E}^p_{\mathrm{vib}} = \alpha_{\mathrm{vib}} \dot{E}^p \tag{12.27}$$

where α_j is the proportion of the energy going to the plasma process j, α_{heat} the proportion going to fast gas heating and finally α_{vib} the proportion going inside vibration and by conservation:

$$\sum_{j=1}^{P} \alpha_j + \alpha_{\mathrm{heat}} + \alpha_{\mathrm{vib}} = 1 \tag{12.28}$$

The molar production rate is retrieved from the proportion of energy going into process

j from the following relation

$$\dot{Q}_j = \alpha_j \frac{\dot{E}^p}{\Delta e_j^m} \tag{12.29}$$

from which the species plasma source term $\dot{\omega}_k^p$ are computed. The Castela model in this representation can be retrieved as particular instance of the model. To do so we consider only one process, $O_2 \longrightarrow 2\,O$, and the corresponding molar production rate is modeled by

$$\dot{Q}_{O_2 \to 2\,O} = \eta \frac{Y_{O_2}}{Y_{O_2}^f} \frac{\dot{E}^p}{2 W_O e_O} \tag{12.30}$$

The corresponding production rates of O and O_2 are deduced from this single reaction as

$$\dot{\omega}_O^p = 2 W_O \dot{Q}_{O_2 \to 2\,O} \tag{12.31}$$
$$\dot{\omega}_{O_2}^p = -W_{O_2} \dot{Q}_{O_2 \to 2\,O} \tag{12.32}$$

The fast gas heating and vibrational production terms are then computed as a result of this chemical source terms as

$$\dot{E}_{\text{heat}}^p = (\alpha \dot{E}^p - \dot{E}_{\text{chem}}^p) \tag{12.33}$$
$$\dot{E}_{\text{vib}}^p = (1-\alpha)\dot{E}^p \tag{12.34}$$

12.2.2 Vibrational relaxation

In the Castela model, the vibrational source term is given by the following source term:

$$\dot{R}_{VT}^p = \rho \frac{e_{\text{vib}} - e_{\text{vib}}^{\text{eq}}(T)}{\tau_{VT}} \ , \ \tau_{VT} = \left(\sum_k \frac{1}{\tau_{VT}^k} \right)^{-1} \tag{12.35}$$

where

$$e_{\text{vib}}^{\text{eq}}(T) = \frac{r \Theta_1}{e^{\Theta_1/T} - 1} \tag{12.36}$$

We claim here that the vibrational energy is only a non-equilibrium one and that the equilibrium part of this vibrational energy is already contained in the thermodynamics of the mixture as shown in Section. 4.1.3. Hence the relaxation source term should be corrected and not include this equilibrium energy leading to:

$$\dot{R}_{VT}^p = \rho \frac{e_{\text{vib}}}{\tau_{VT}} \ , \ \tau_{VT} = \left(\sum_k \frac{1}{\tau_{VT}^k} \right)^{-1} \tag{12.37}$$

12.2.3 Temporal and spatial distribution of the discharge

The different channels of a plasma discharge have been modeled in the previous section; the temporal and spatial shapes of the discharge source term $\dot{E}^p$ are now discussed. The NRP discharges are imposed with a temporal frequency of f_d and corresponding temporal period T_d. We impose as input of the model a discharge energy E_d that is applied only during τ_{pulse} over a discharge spatial profile $\mathcal{F}(\mathbf{r})$:

$$\dot{E}^p(\mathbf{r},t) = \begin{cases} \frac{\varepsilon_d}{\tau_{\text{pulse}}}\mathcal{F}(\mathbf{r}) & t \,(\text{mod}\, T_d) \leq \tau_{\text{pulse}} \\ 0 & t \,(\text{mod}\, T_d) > \tau_{\text{pulse}} \end{cases} \tag{12.38}$$

In the Castela model, the shape function $\mathcal{F}$ is taken as an analytical profile using erfc functions. Thanks to the multi-dimensional streamer computations, more realistic energy density profiles of streamers are now available.

Conclusion and perspectives on Part I, II and III

In this work, fully coupled plasma assisted combustion simulations are performed. To do so the plasma discharges features have been in AVIP, a low-temperature plasma and combustion code. To simulate plasma discharges, the drift-diffusion equations have been chosen for plasma species. In addition the Poisson equation must be solved for consistency to get the electric field which the plasma species react to. Third, solving combustion is done by integrating the reactive Navier-Stokes equations with the code AVBP.

Plasmas span a large range of amplitudes and are very stiff to solve numerically. Hence Total Variation Diminushing schemes, for both drift-diffusion equations and Navier-Stokes equations, have been developed and validated against canonical cases. This necessitated to change the schemes formulation from the original AVBP data structure and to develop Finite Volume Vertex-Centered Methods. Two schemes have been implemented for drift-diffusion equations: the improved Scharfetter Gummel scheme for quadrangular elements and the limited Lax-Wendroff scheme for triangular elements. The convective part of the Navier-Stokes equations, the Euler fluxes, are integrated using HLLC with a MUSCL reconstruction and two-stages time integration (RK2). The Poisson equation is also discretized in a Finite Volume Vertex-Centered formulation and validated against analytical solutions for convergence.

To perform multi-dimensional studies while still keeping the computational cost low, axisymmetric configurations have been considered in this work. This required solving the Poisson equation and transport equations in cylindrical formulation with an azimuthal invariance. The Navier-Stokes equations, in particular, included geometric source terms due to the cylindrical formulation. These have all been added in AVIP.

Finally streamer simulations have been performed, first using simple chemistries in canonical configurations for validating the code capabilities. We then turned to pin-pin configurations which are closer to reality. A study of NRP discharges for plasma assisted ignition of a methane-air mixture has been carried out and the benefits of NRP discharges have been demonstrated. We showed that, at the right frequency, the creation of a conducting channel eases the process of discharge propgation while creating a homogeneous temperature region that enhances combustion compared to a single pulse discharge.

These fully-coupled simulations pave the way for the development of precise models, based on those fully coupled simulations, that could be used in industrial configurations.

A major drawback in PAC simulations and discharge simulations is the computational cost, which is due in a large part to the resolution of the Poisson equation which can represent up to 80% of the total cost. The last part of this thesis is therefore devoted to the development of an alternative method to significantly accelerate the simulations.

Part IV

Deep Learning and the Poisson equation

Chapter 13

Deep Learning and Neural Networks

Contents

This chapter presents a brief introduction to deep learning with some general definitions and properties of multilayer perceptrons (MLPs). The basic building blocks of MLPs are recalled as well as the fundamental backpropagation algorithm. In a second part, the special case of Convolutional Neural Networks (CNNs), used extensively in this work, is presented.

13.1 Definitions and properties of neural networks

13.1.1 General definitions and introduction

Feedforward neural networks, also called **fully-connected layer neural networks** or **multilayer perceptrons** (MLPs) are the canonical deep learning models. Their goal is to approximate some unknown mapping $\mathbf{y} = \tilde{f}(\mathbf{x})$ between an input $\mathbf{x}$ and an output $\mathbf{y}$. MLPs are often used in image classification models where $\mathbf{x}$ is an image whereas $\mathbf{y}$ is a category, *i.e.* a dog, a cat or a plane. MLPs approximate the desired function $\tilde{f}$ with an approximate function $f(\mathbf{x}; \boldsymbol{\theta})$ that is dependent on a set of parameters $\boldsymbol{\theta}$. Note that the output of the network in this section will always be denoted $\mathbf{a} = f(\mathbf{x}; \boldsymbol{\theta})$.

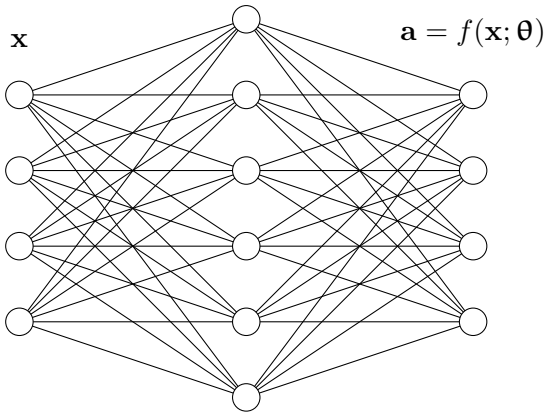


Figure 13.1: Sketch of a MLP network with one hidden layer

The feedforward property of these networks comes from the information propagation in MLPs which travels from the input to the output without any feedback loop. The approximate function is indeed composed of different intermediate functions to form a chain structure. These different intermediate functions, also called **layers**, work together as a **network** to give the desired output $\mathbf{y}$ where for three layers we would have: $f(\mathbf{x}) = f^1(f^2(f^3(\mathbf{x})))$. The number of these intermediate functions is called the **depth** and in practical architectures there are many hence the **deep learning** terminology. The last layer of the network is called the **output layer** and represents the approximation to the target value $\mathbf{y}$. All other intermediate layers are called **hidden layers** because the training data does not show the desired output for each of these layers: the hidden layers create abstract representation of the data to better predict it. A simple example of a feedforward network with one hidden layer is shown in Fig. 13.1.

Finally the term *neurons* comes from a loose connection with neuroscience. Instead of thinking of the network in terms of layers, each unit of the network can be seen as working in parallel to create abstract representations. However it is best to see feedforward neural networks as function approximators that are aimed to achieve statistical generalization, occasionally drawing insights from neuroscience [Goodfellow et al., 2016, Chap. 6].

13.1.2 Gradient Based learning

Designing a neural network is not much different from a classical machine learning model: we must choose an optimization procedure, a loss function and a model family (in this case feedforward neural networks).

MLPs learn from a set of input examples the best approximation $f(\cdot;\boldsymbol{\theta})$ of the unknown mapping $\tilde{f}$ where $\boldsymbol{\theta}$ is the vector of parameters of the model. A loss function $\mathcal{L} = \mathcal{L}(\cdot;\boldsymbol{\theta})$ is defined to measure how close the approximate mapping f is to the exact mapping $\tilde{f}$. At each step of the optimization procedure, $\boldsymbol{\theta}$ is adjusted to get closer to the exact mapping. A set of examples, the **training** set, is processed by the network and allows to adjust the parameters $\boldsymbol{\theta}$ to get closer to the exact mapping. A set of input examples never seen in the optimization procedure, called the **validation** set, is used in conjunction at each step to see how the network behaves on data it is has not seen. The validation set allows to *refine* the model

and prevents **overfitting**: when the training loss decreases but the validation loss stops decreasing as shown in Fig. 13.2 or even increases the model does not learn any generalizable features but only features that are specific to the training set. Hence the optimal model is often taken as the one for which the validation loss is minimal. Finally the **test** set is a set of data that is never used to design the model and it estimates how successful the model is on never-seen data.

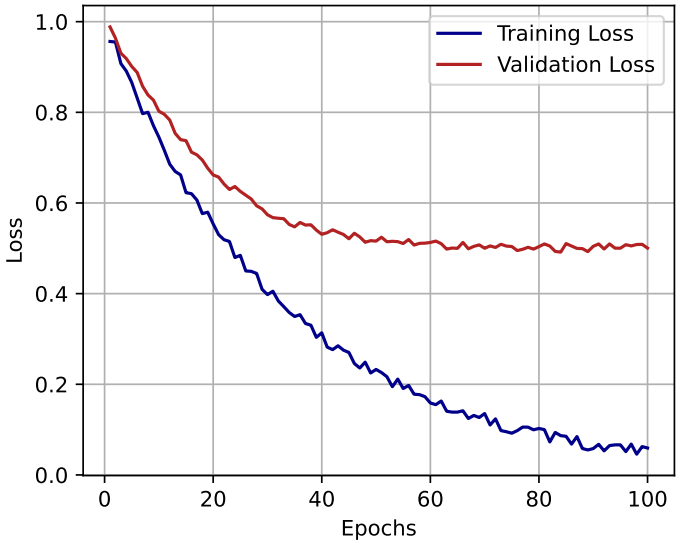


Figure 13.2: Illustration of overfitting.

The basic principles of gradient learning are now discussed. Let us take n input examples, x being one of the input and y the desired output. During each step of the training procedure we update the hyperparameters $\boldsymbol{\theta}$ as $\boldsymbol{\theta}' = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$ and the loss function changes as [Nielsen, 2016, Chap. 1]:

$$\delta\mathcal{L} = \nabla\mathcal{L} \cdot \delta\boldsymbol{\theta} \tag{13.1}$$

The hyperparameters are updated in the direction of the gradient descent to yield a decreasing loss function:

$$\delta\boldsymbol{\theta} = -\eta\nabla\mathcal{L} \implies \delta\mathcal{L} = -\eta||\nabla C||^2 \tag{13.2}$$

so that the cost will always decrease. As presented in the next sections, neural networks are non-linear and that causes most lost functions to become non-convex. Convex optimization converges for any set of initial parameters but this is not true anymore for non-convex optimization. Initialization is therefore critical in neural networks and parameters are usually initialized randomly with small values.

13.1.2.1 Stochastic Gradient Descent

From a training set of n pairs $(\mathbf{x}, \mathbf{y})$ of input/target, the loss function is defined on this set as:

$$\mathcal{L} = \frac{1}{n} \sum_x \mathcal{L}_x(\mathbf{a}, \mathbf{y}) \tag{13.3}$$

where $\mathcal{L}_x$ is the loss function for each input $\mathbf{x}$ and $\mathbf{a} = f(\mathbf{x}; \boldsymbol{\theta})$ is the output of the neural network. The gradient is thus

$$\nabla \mathcal{L} = \frac{1}{n} \sum_x \nabla \mathcal{L}_x \tag{13.4}$$

At each step of the gradient descent, the loss function must be computed on the whole training set. This step is costly as the training set size is usually huge. This is computationally expensive only to make a *single* step of gradient descent. Deep learning would be very slow if the loss functions were computed on the whole training set. **Stochastic Gradient Descent** (SGD) solves this issue by introducing mini-batches. The batch-size $m \ll n$ is defined to approximate the loss function using a portion of the whole training set so that computation is much faster:

$$\nabla \mathcal{L} \simeq \frac{1}{m} \sum_x \nabla \mathcal{L}_x \tag{13.5}$$

For a fixed model size, the cost of SGD does not depend on the size of the training set n: it provides a scalable way of training non-linear models on huge datasets.

13.1.3 Loss functions and units

We now consider a feedforward neural network of L layers that takes as input a vector $\mathbf{x}$. The feedforward neural network takes as input $\mathbf{a}^0 = \mathbf{x}$ and across each layer $l \in [\![1, L]\!]$ the following operations are performed:

$$\mathbf{z}^l = \mathbf{W}^l \, \mathbf{a}^{l-1} + \mathbf{b}^l \tag{13.6}$$

$$\mathbf{a}^l = \sigma(\mathbf{z}^l) \tag{13.7}$$

where σ is called the activation function, $\mathbf{W}^l \in \mathcal{M}_{n_l, n_{l-1}}(\mathbb{R})$ the weight matrix and $\mathbf{b}^l \in \mathbb{R}^{n_l}$ the bias. A sketch of a two hidden layer feedforward network is shown in Fig. 13.3 where the above notations are used.

13.1.3.1 Loss functions and output units

The choice of output unit is tightly coupled with the choice of the loss function: the output layer determines the way that we interpret and compute the result of the neural network. For example for classification problems the output layer should be composed of the number of classes we want to represent: to categorize numbers on images (the MNIST databse), the output layer should be of size 10, each output unit representing one number.

Loss functions must provide a distance of the computed solution $f(\mathbf{x}; \boldsymbol{\theta})$ to the actual target $\mathbf{y}$. The most straightforward choice of loss function is the quadratic loss function or mean-squared error (MSE) which is the just the L^2 distance between the predicted output value and the actual target:

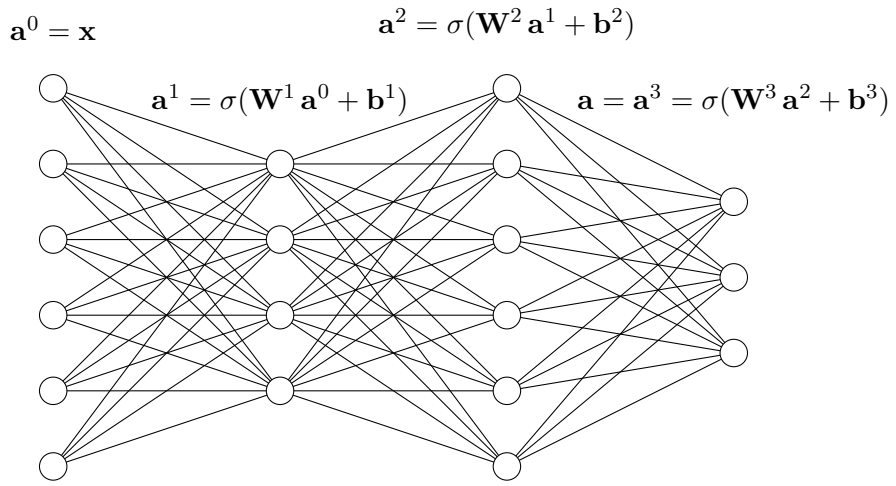


Figure 13.3: Sketch of a MLP network with two hidden layers.

$$\mathcal{L}_{\text{MSE}}(\boldsymbol{\theta}) = ||\mathbf{a} - \mathbf{y}||_2^2/2 \tag{13.8}$$

where $\mathbf{a}$ is the output of the network. This is a good measure in any case of the distance to the predicted value. Other losses are more suited depending on the case but the MSE is always a good starting point. The mean-absolute error (MAE) loss is also a possibility by switching to L^1 norm:

$$\mathcal{L}_{\text{MAE}}(\boldsymbol{\theta}) = ||\mathbf{a} - \mathbf{y}||_2^2/2 \tag{13.9}$$

For regression, an improvement of the MSE is the Huber loss which is less sensitive to outliers than the MSE loss while still keeping quadratic slopes near the bottom of the loss:

$$\mathcal{L}_{\delta,\text{Huber}}(\boldsymbol{\theta}) = \begin{cases} \frac{1}{2}||\mathbf{y} - \mathbf{a}||_2^2 & \text{if} \quad ||\mathbf{y} - \mathbf{a}|| < \delta \\ \delta(||\mathbf{y} - \mathbf{a}||_2 - \frac{1}{2}\delta) & \text{otherwise} \end{cases} \tag{13.10}$$

In classification problems the cross-entropy function is often used and allows to fix the learning slowdown problem encountered with sigmoid layers and the MSE loss [Nielsen, 2016]. For M classes the cross-entropy loss reads

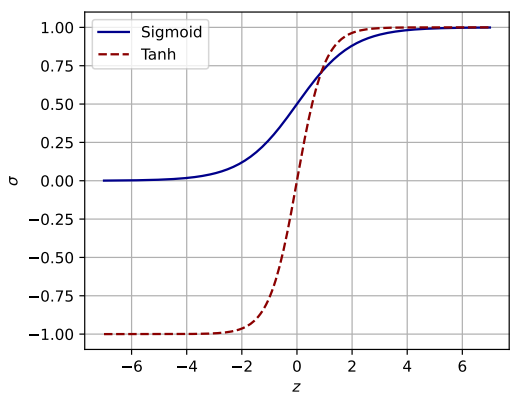
$$\mathcal{L}_{\text{CE}}(\boldsymbol{\theta}) = -\sum_{c=1}^{M} y_c \log(p_c) \tag{13.11}$$

where y_c is a binary indicator of whether or not c is the correct class for the input and p_c is the predicted probability that the input is of class c.

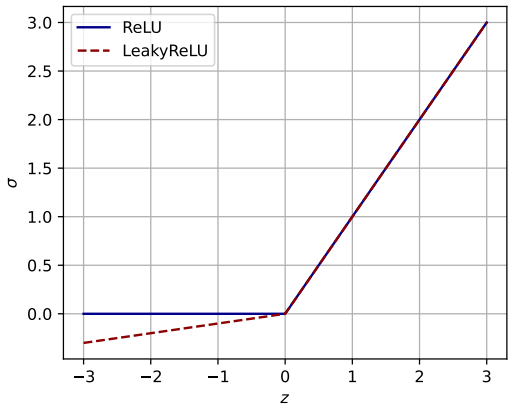
13.1.3.2 Hidden Units

The choice of activation functions in hidden layers is now discussed. Since deep learning uses a lot of hidden layers, this choice is critical for the design of neural networks. However there are not definite guiding principles and we more rely on experience for the design of hidden

units.



(a) Sigmoid and tanh functions



(b) ReLU and LeakyReLU functions

Figure 13.4: Typical activation functions used in neural networks.

Rectified Linear Units Rectified linear units use the following activation function: $g(z) = \max(0, z)$ shown in Fig. 13.4a. It is very smilar to linear units and non-linearity is introduced in the simplest possible way. The derivative remains large whenever the unit is activate which makes it easily interpretable as there are no second-order effects. However rectified linear units cannot learn when the activation is zero as the gradient is not always non-zero since it has a different value for the left and right derivatives.

Various generalizations of rectified linear units have been derived to solve this issue and all start from the general form of activation function

$$\sigma(z) = \max(0, z) + \alpha \min(0, z) \tag{13.12}$$

where α is now a parameter controling the slope of the function in the negative region. **Absolute value rectification** [Jarrett et al., 2009] imposes $\alpha = -1$ leading to an absolute value activation function: it is used for object recognition where it makes sense to search for features that are invariant under reversed polarity [Chap. 6][Goodfellow et al., 2016]. **LeakyReLU** [Maas et al., 2013] imposes a constant small positive value, for example $\alpha = 0.1$ shown in Fig. 13.4b whereas **Parametrized ReLU** [He et al., 2015] makes the network learn the parameter α.

Sigmoid units Before the introduction of ReLU units, the most used activation functions were the sigmoid and hyperbolic tangent functions which are shown in Fig. 13.4a. Sigmoid and hyperbolic tangent units, unlike piecewise linear units, suffer from saturation at low and high values of the unit which makes learning difficult for these kind of units. Because of that, their use in hidden units is now discouraged [Goodfellow et al., 2016].

13.1.4 The universality theorem

A linear model, mapping from input to output through matrix multiplication, can only learn linear functions. This would be the case of the feedforward neural network without non-linear activation functions presented earlier. These are easy to train because loss functions associated to linear models are often convex which ensures convergence. On the other hand we want the feedforward networks to learn non-linear functions.

Fortunately feedforward neural networks can compute any function. Cybenko [1989] and Hornik et al. [1989] demonstrated that standard multilayer feedforward networks with as few as one hidden layer using arbitrary squashing functions are capable of approximating any Borel measurable function provided sufficiently many hidden units. The concept of Borel measurability is outside the scope of the thesis and it suffices here to say that any continuous function on a closed and bounded set of $\mathbb{R}^n$ can be approximated by feedforward networks. In this sense neural networks are **universal approximators**.

13.1.5 The backpropagation algorithm

The backpropagation algorithm, which enables neural networks to learn fast by a easily computing gradients, is given in this section. Notations introduced above are kept and the sketch of a four layer MLP network following those notations is found in Fig. 13.3. The backpropagation algorithm proceeds in 4 steps:

1. **Feedforward**: the network processes the input $\mathbf{x}$ by starting with $\mathbf{a}^0 = \mathbf{x}$:

for $l \in [\![1, L]\!]$ **do**

$$\mathbf{z}^l = \mathbf{W}^l \mathbf{a}^{l-1} + \mathbf{b}^l \tag{13.13}$$

$$\mathbf{a}^l = \sigma(\mathbf{z}^l) \tag{13.14}$$

end

where the different parameters of the network have been described earlier. We also define the error of neurons in layer l:

$$\delta^l = \frac{\partial \mathcal{L}}{\partial \mathbf{z}^l} \tag{13.15}$$

2. **Output error**: the output error is the starting point of the backpropagation and can be computed as [Goodfellow et al., 2016, Nielsen, 2016]:

$$\delta^{x,L} = \nabla_a \mathcal{L} \odot \sigma'(\mathbf{z}^L) \tag{13.16}$$

3. **Backpropagate**: from the output error all the other errors of each layer are computed backwardly so that

4. **Gradient descent**: finally the derivative of the cost function with respect to each parameter of the model can be computed from the errors backpropagated:

for $l \in [\![L-1, 1]\!]$ **do**

$$\delta^l = ((\mathbf{W}^{l+1})^T \delta^{l+1}) \odot \sigma'(\mathbf{z}^l) \tag{13.17}$$

end

for $l \in [\![1, L]\!]$ **do**

$$\frac{\partial \mathcal{L}}{\partial w^l_{jk}} = a^{l-1}_k \delta^l_j \qquad \frac{\partial \mathcal{L}}{\partial b^l_j} = \delta^l_j \tag{13.18}$$

$$\mathbf{W}^l \to \mathbf{W}^l - \frac{\eta}{m} \sum_x \delta^l (\mathbf{a}^{l-1})^T \tag{13.19}$$

$$\mathbf{b}^l \to \mathbf{b}^l - \frac{\eta}{m} \sum_x \delta^l \tag{13.20}$$

end

13.1.6 Training a feedforward neural network

Over the course of the past decade, problems encountered in the training of neural networks have given rise to several techniques to circumvent those issues and some of these are presented in this section.

13.1.6.1 Overfitting and regularization

When accuracy stops improving the validation data, *i.e.* the validation loss stops decreasing, the model is learning peculiarities about the training set. When classification accuracy reaches a plateau what the network learns afterwards no longer generalizes to the test data and the network is overfitting.

In general the best way of reducing overfitting is to increase the size of the training dataset. The training data can be artificially expanded when there is not enough ressource to produce more training data. To do so the data is expanded by applying operations that reflect real world variations: as an example for image recognition we can apply rotations, translations, distorsions and skewing.

Modifying the loss function can also circumvent the overfitting issue. A term is added in the loss function to make the network learn small weights (in absolute value):

1. L_1 regularization:
$$\mathcal{L} = \mathcal{L}_0 + \frac{\lambda}{2n} \sum_i w_i^2 \tag{13.21}$$

2. L_2 regularization:
$$\mathcal{L} = \mathcal{L}_0 + \frac{\lambda}{n} \sum_i |w_i| \tag{13.22}$$

where w_i represent all the weights flattened to one dimension.

Dropout can also be used, we randomly delete (temporarily) a portion of the hidden neurons in the network for each batch. The dropout is essentially averaging the effects of a very large number of different networks.: those different networks may overfit in different ways and averaging may help eliminate that kind of overfitting.

13.1.6.2 Weight initialization

As we are dealing with non-convex optimization when training neural networks, any set of initial parameters does not necessarily converge to the solution of the problem. Therefore a proper initialization of the parameters is essential for the network to learn efficiently. The most basic initialization of weights and biases would be to take them from a centered Gaussian distribution with 1-standard deviation:

$$w \sim \mathcal{N}(0, 1) \tag{13.23}$$
$$b \sim \mathcal{N}(0, 1) \tag{13.24}$$

This causes the same saturation problem as for the output layer but this time for the hidden layers because the resulting output will have a broad dispersion and thus high values. A better choice is:

$$w \sim \mathcal{N}(0, 1/\sqrt{n_{\text{in}}}) \tag{13.25}$$
$$b \sim \mathcal{N}(0, 1/\sqrt{n_{\text{in}}}) \tag{13.26}$$

13.1.6.3 Choosing a neural network's hyper parameters

The first challenge is to get any non trivial learning, *i.e.* for the network to achieve results better than by chance. To do so a good idea is to start reducing the size of training and validation data and play with the learning rate. When there is a signal for η, find the larger η for which the training loss function doesn't oscillate.

A learning rate schedule is also possible in which η decreases gradually. The regularization parameter is to be determined after the learning rate.

13.1.6.4 The vanishing gradient problem

We expect networks with more hidden layers to be more powerful. Such networks could use the intermediate layers to build up multiple layers of abstraction.

However there is an issue that makes deep networks hard to train: the different layers are learning at vastly different speeds. In particular, when latter layers in the network are learning well, early layers often get stuck during training, learning almost nothing at all. This is the vanishing gradient problem.

13.2 Convolutional neural network

A feedforward network architecture presented in the first section of the chapter does not take into account the spatial structures of the inputs. It treats input pixels which are far apart and close together on exactly the same footing. CNNs on the other hand take advantage of the spatial structure of the inputs.

First uses of CNNs date as far back as 1997 when LeCun et al. [1997] used CNNs for character recognition tasks. However their current widespread use is due to the successful application of a CNN in Krizhevsky et al. [2012] to beat state-of-the-art classification in the ImageNet challenge.

13.2.1 Discrete convolution

Convolutional neural networks (CNNs) are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers. Convolution is a commutative mathematical operation between two functions

$$s(t) = (x \star w)(t) \equiv \int_{\mathbb{R}} x(a) w(t-a)\,\mathrm{d}a = (w \star x)(t) \tag{13.27}$$

When the integration variable is discretized, the discrete convolution is defined as an infinite discrete sum following

$$s(t) = \sum_{a=-\infty}^{+\infty} w(a) x(t-a) \tag{13.28}$$

In convolutional networks the data are defined on a grid so that in the end the operation reads for 2D convolution

$$S(i,j) = (K \star I)(i,j) = \sum_m \sum_n I(i-m, j-n) K(m,n) \tag{13.29}$$

where K is called the **convolution kernel**. The minus sign is only present to preserve the commutative property. In most deep learning frameworks what is actually implemented is the discrete cross-correlation [Goodfellow et al., 2016, Chap. 9]:

$$S(i,j) = (K \star I)(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n) \tag{13.30}$$

In this text, discrete convolution is still used abusively to describe the discrete cross-correlation.

Convolution is interesting because it keeps the intrinsic structure of the data. Deep learning tasks involve images, sound clips and many other kind of data for which, at least on one axis, *ordering matters*. These spatial or temporal structures are completely lost in the classical feedforward neural network presented in the first section. The basic definitions regarding discrete convolutions for neural networks is recalled here and the interested reader can refer to [Goodfellow et al., 2016, Chap. 9] for a more thorough material. Figures of 2D convolutions presented next are adapted from the github repository `https://github.com/vdumoulin/conv_arithmetic` compagnion of the Dumoulin and Visin [2018] paper.

A simple 2D convolution is shown in Fig. 13.5 where a kernel of size 3×3 is applied on a 5×5 input. The convolution kernel slides across the input producing an output of size 3×3. The same kernel is applied all across the input so that a similar treatment is applied all over the input: this allows to preserve locality. From this simple example we can see that the size of the output is tightly linked with the size of the input *and* the kernel size. The **stride**, *i.e.* the distance between two consecutive positions of the kernel, and the **padding**, *i.e.* the number of zeros concatenated at the beginning and end of each axis, also play a role in the size of the output layer.

An example of a 2D convolution kernel of size 3×3 applied on an input of size 5×5 with 1-padding and 2-strides is shown in Fig. 13.6 resulting in an output of size 3×3.

The examples here only show one input and one output per layer but multiple images can be used in each layer. The width of each layer defined will be the size of the third dimension of the kernel k that is the only one that is fixed.

Only two-dimensional convolutions are shown in the figures for simplicity but convolution can be applied in N-dimensions. The convolution kernel will be a generalized N-dimensional cube that will slides across the corresponding N-dimensional input.

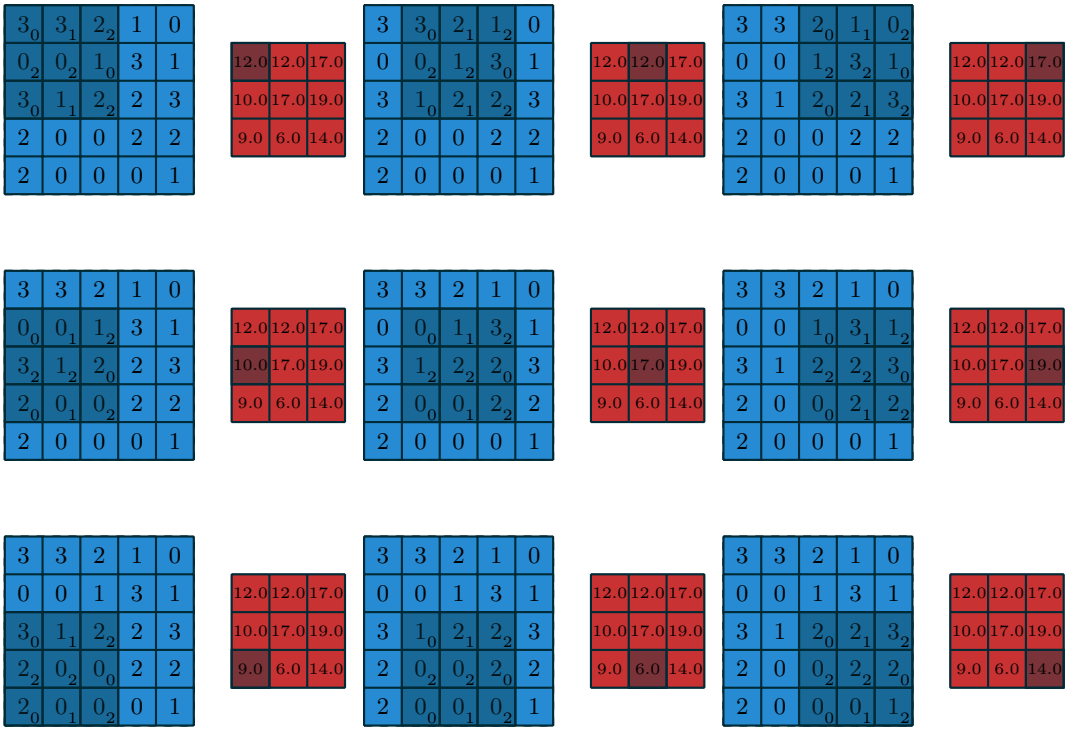


Figure 13.5: Simple 2D convolution with no padding, no strides with input size of 5 and output size of 3.

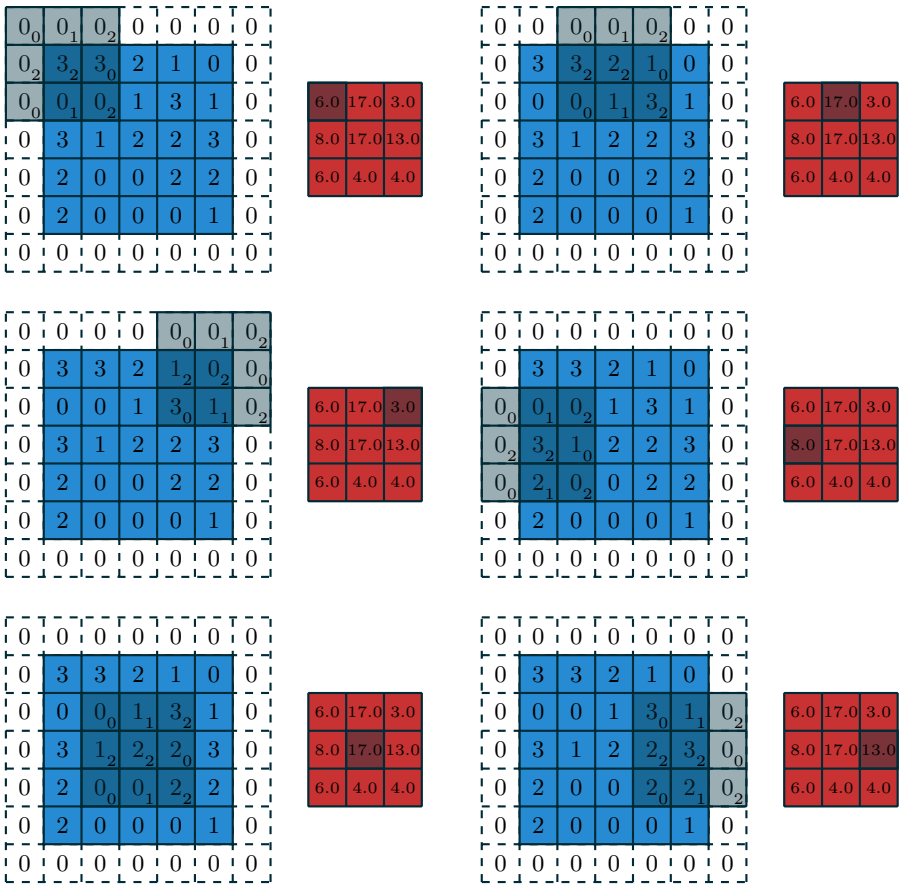


Figure 13.6: 2D convolution with 1-padding, 2-strides with input size of 5 and output size of 3.

13.2.2 Convolution arithmetic

A few basics of convolution arithmetic is recalled here adapted from Dumoulin and Visin [2018]. This section assumes:

- 2-D discrete convolutions ($N = 2$)
- square inputs ($i_1 = i_2 = i$)
- square kernel size ($k_1 = k_2 = k$)
- same strides along both axes ($s_1 = s_2 = s$)
- same zero padding along both axes ($p_1 = p_2 = p$)

The general formula relating the input size i to the output size o is:

$$o = \left\lfloor \frac{i + 2p - k}{s} \right\rfloor + 1 \tag{13.31}$$

13.2.3 Pooling layers

CNNs also contain pooling layers to simplify the information in the output from the convolutional layer. A pooling layer takes each feature map output from the convolutional layer and prepares a condensed feature map: the size of feature maps are reduced using average, maximum or minimum operations to *summarize* regions. An example of a pooling operation in an input of size 5×5 is shown in Fig. 13.7 where the average is performed from the input to the output.

13.2.4 Transposed convolution arithmetic

A transposed convolution (or deconvolution) is a transformation going in the opposite direction of a normal convolution, from something that has the shape of the output of some convolution to something that has the shape of its input while maintaining a connectivity pattern that is compatible with said convolution.

Let us take a convolution with $i = 4, k = 3, s = 1, p = 0$. This gives an output size of $o = 2$. If the input and output are unrolled into vectors from left to right, top to bottom, the convolution can be represented as a sparse matrix $\mathbf{C}$:

$$\begin{bmatrix} w_{0,0} & w_{0,1} & w_{0,2} & 0 & w_{1,0} & w_{1,1} & w_{1,2} & 0 & w_{2,0} & w_{2,1} & w_{2,2} & 0 & 0 & 0 & 0 & 0 \\ 0 & w_{0,0} & w_{0,1} & w_{0,2} & 0 & w_{1,0} & w_{1,1} & w_{1,2} & 0 & w_{2,0} & w_{2,1} & w_{2,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_{0,0} & w_{0,1} & w_{0,2} & 0 & w_{1,0} & w_{1,1} & w_{1,2} & 0 & w_{2,0} & w_{2,1} & w_{2,2} & 0 \\ 0 & 0 & 0 & 0 & 0 & w_{0,0} & w_{0,1} & w_{0,2} & 0 & w_{1,0} & w_{1,1} & w_{1,2} & 0 & w_{2,0} & w_{2,1} & w_{2,2} \end{bmatrix}$$

Using this representation, the deconvolution is obtained by transposing $\mathbf{C}$; in other words, the error is backpropagated by multiplying the loss with $\mathbf{C}^T$.

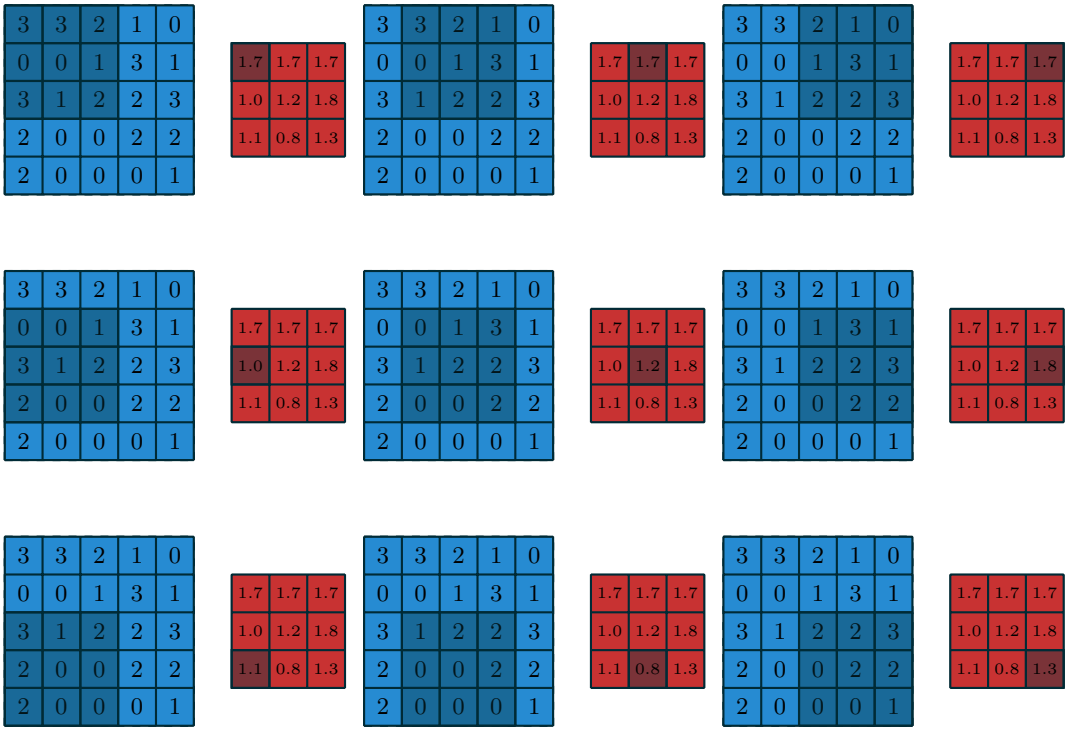


Figure 13.7: Average pooling from input size 5 to output size 3.

13.2.5 CNN backpropagation

The CNN backpropagation algorithm proceeds in exactly the same way as the classical backpropagation algorith by replacing the weight matrix $\mathbf{W}$ by the kernel convolution K and the matrix multiplication by the convolution operation $\star$.

13.2.5.1 Example of a MNIST network

A typical example of a 2D convolutional neural network is shown in Fig. 13.8. From the 28×28 resolution images of the MNIST database, 2D convolution with a pooling layer are applied two times. Then MLP layers are applied to *flatten* the output to yield 10 neurons each representing the probability of the image to be a certain number.

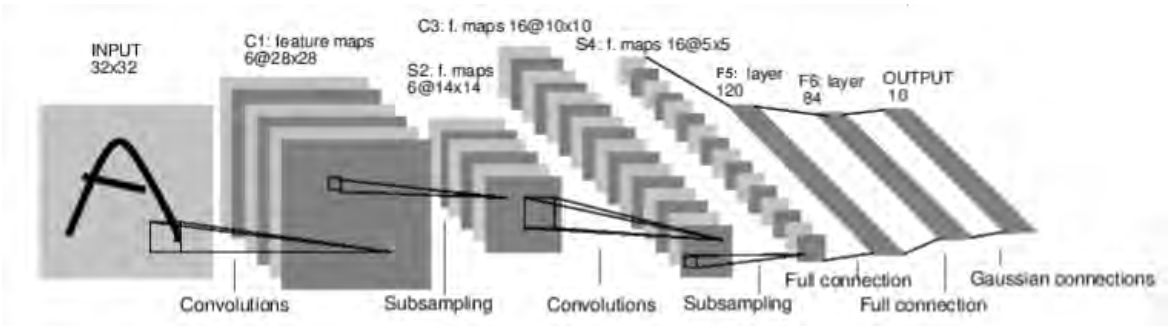


Figure 13.8: MNIST network taken from PyTorch tutorials.

Chapter 14

Elliptic PDEs and Neural Networks

14.1 Introduction to papers

From the earlier chapters, solving the Poisson equation is a complicated task that involves solving linear systems. In this thesis, attempt to solve the Poisson equation using CNNs has been carried out in structured grids. The first paper discusses the central role of the receptive field to correctly predict the potential and electric field from a charge density. We try to understand how multiple scale architectures, especially the UNet [Ronneberger et al., 2015], interpret the Poisson equation. Coupling with streamer test cases similar to the ones described Chap. 10 has been implemented and shows good propagation results where the speed and energy of the streamer discharges are preserved.

In a second paper, accepted at NeurIPS Workshop Machine Learning and the Physical Sciences (`https://ml4physicalsciences.github.io/2021/`), we apply the methodology discussed above to a more general set of elliptic differential equations, the set of Helmholtz equations of the photoionization source term presented in Sections. 2.4.4 and 6.5. The global network parameters found for the Poisson equation also hold true for this more general type of equations and interaction with transport equations also yield satisfying results.

As recalled in the papers, the PlasmaNet library (`https://gitlab.com/cerfacs/plasmanet`) has been written to perform all those studies.

USING NEURAL NETWORKS TO SOLVE THE 2D POISSON EQUATION FOR ELECTRIC FIELD COMPUTATION IN PLASMA FLUID SIMULATIONS

Lionel Cheng
CERFACS
42 Avenue Gaspard Coriolis
31057 Toulouse, France
cheng@cerfacs.fr

Ekhi Ajuria Illarramendi
ISAE-SUPAERO / CERFACS
Université de Toulouse
Toulouse, France
ajuria@cerfacs.fr

Guillaume Bogopolsky
SAFRAN AIRCRAFT ENGINES / CERFACS
Campus de l'Espace, 1 avenue Hubert Curien,
27207 Vernon, France
bogopolsky@cerfacs.fr

Michaël Bauerheim
ISAE-SUPAERO
Université de Toulouse
Toulouse, France
michael.bauerheim@isae-supaero.fr

Bénédicte Cuenot
CERFACS
42 Avenue Gaspard Coriolis
31057 Toulouse, France
cuenot@cerfacs.fr

ABSTRACT

The Poisson equation is critical to get a self-consistent solution in plasma fluid simulations used for Hall effect thrusters and streamer discharges, since the Poisson solution appears as a source term of the unsteady nonlinear flow equations. Two types of plasma fluid simulations are considered in this work: the canonical electron plasma oscillation and a double headed streamer discharge. As a first step, solving the 2D Poisson equation with zero Dirichlet boundary conditions using a deep neural network is investigated using multiple-scale architectures, defined in terms of number of branches, depth and receptive field [1]. One key objective is to better understand how neural networks learn the Poisson solutions and provide guidelines to achieve optimal network configurations, especially when coupled to the time-varying Euler equations with plasma source terms. Here, the Receptive Field is found critical to correctly capture large topological structures of the field. The investigation of multiple architectures, losses, and hyperparameters provides an optimal network to solve accurately the steady Poisson problem. The performance of the optimal neural network solver, called PlasmaNet, is then monitored on meshes with increasing number of nodes, and compared with classical parallel linear solvers. Next, PlasmaNet is coupled with an unsteady Euler plasma fluid equations solver in the context of the electron plasma oscillation test case. In this time-evolving problem, a physical loss is necessary to produce a stable simulation. PlasmaNet is finally tested on a more complex case of discharge propagation involving chemistry and advection. The guidelines established in previous sections are applied to build the CNN to solve the same Poisson equation in cylindrical coordinates with different boundary conditions. Results reveal good CNN predictions and pave the way to new computational strategies using modern GPU-based hardware to predict unsteady problems involving a Poisson equation, including configurations with coupled multiphysics interactions such as in plasma flows.

Keywords: Convolutional neural network, Poisson equation, plasma oscillation, streamer discharge, plasma fluid simulations

[1] All the code in this work is available at https://gitlab.com/cerfacs/plasmanet

1 Introduction

The Poisson equation is a well-known equation encountered in many fields of physics, from gravitation to incompressible flows, as well as in plasmas. In the context of plasma numerical simulations, the resolution of the Poisson equation for the electric field, which appears as source terms coupled to the nonlinear flow equations, is critical to properly describe the coupling with the charged particles' density fields. In this context, the Poisson equation relates the electromagnetic potential ϕ to the charge distribution ρ_q:

$$\nabla^2 \phi = -\frac{\rho_q}{\varepsilon_0} \tag{1}$$

where ε_0 is the magnetic permeability in vacuum.

The potential gives access to the electric field as $\mathbf{E} = -\boldsymbol{\nabla}\phi$. Plasma fluids are made of transport charged species i of density n_i, velocity $\mathbf{v}_i$ and specific energy E_i governed by the Euler transport equations in which the electric field appears as a source term. The classical way to solve the Poisson equation is to discretize the Laplace operator on a mesh so that Eq. (1) reduces to a linear system $A\phi = R$, where A is the Laplace operator matrix and R is the discretized version of the charge density. Direct or iterative methods can be used to solve such linear systems [1]. The computational cost of solving Eq. (1) increases with the mesh size. Following the idea of FluidNet [2, 3], this work introduces data-driven methods to accelerate the resolution of the Poisson equation, and investigate their behavior when coupled to unsteady Euler equations.

The resolution of Partial Differential Equations (PDE) using Machine Learning techniques was first developed in the mid-1990s, with the introduction of MultiLayer Perceptrons (MLP) [4] to solve a 2D Poisson equation with Dirichlet boundary conditions [5]. Although showing promising results, these first attempts were quickly limited by the available computational resources. With today's computational power as well as the recent development of user-friendly Machine Learning frameworks, the interest in the resolution of PDEs with Machine Learning has significantly grown. One of the main milestones in the field was the introduction of *Physics informed neural networks* (PINN) [6] which incorporates physical knowledge into the neural networks. For example, training the network to minimize the residual of physical equations imposes a physical constraint on the network. Such networks employ automatic differentiation [7] resulting in mesh-free methods. Yet, PINNs have to be trained specifically for each boundary or initial condition, which limits their practical use, in particular in Computational Fluid Dynamics (CFD).

The capability of neural networks to approximate the solution of PDEs has thus led to the development of various surrogate models for fluid mechanics simulation. For example, Convolutional Neural Networks (CNN) trained with physical loss functions have been used to substitute incompressible fluid solvers [8, 2, 3]), obtaining stable simulations with considerable speed-up. An *a posteriori* physical correction of the network predictions have also been reported successful in time-evolving problems [9], avoiding error accumulation in time. For a complete review of the use of Machine Learning in fluid dynamics, the reader is referred to the review of Brunton et al. [10].

Instead of completely substituting a CFD solver, this work focuses on one particular step of the resolution process: the Poisson equation. Initial attempts using MLPs [11] were quickly followed by the introduction of CNNs [12, 13], which were better suited to map the input and output in 2 or 3 dimensions. Nevertheless, they still treated the network as a separate instance to the fluid solver, as they were trained *outside the box* and were not coupled to a CFD solver to validate the methodology on steady or unsteady simulations.

Recent works have used CNNs [2] to solve the Poisson equation, coupled to the fluid solver, embedding the concept of a simulation into the neural network. Additionally, the hybridization of such embedded networks with standard Poisson solvers provided fast and robust CFD solvers, especially on configurations with new physics (for example a network trained with constant density flow, but tested on variable density simulations [3]). Similarly, differentiable fluid solvers [14] have also recently been introduced, which enabled to encode flow dynamics during the training process.

These works are extended here by training a deep CNN to solve the Poisson equation in the context of plasma flows, where it is used to obtain the electric field from a charge distribution. A first objective is to better understand how to design the neural architecture and its associated hyperparameters to achieve stable and accurate plasma flow simulations. A second objective is therefore to couple the data-driven Poisson solver to a multiphysics Euler plasma unsteady solver, and evaluate the resulting accuracy and performance (accuracy referring to the precision of the network prediction and performance to the network inference time). In such time-evolving multiphysics problems, it is found critical (1) to train the network using a physical-based loss function, and (2) to design optimal network architectures, for which the receptive field is found essential. In section 2, a first analytical test problem is presented. The methodology based on CNN is described in section 3 with different loss functions and architectures. In section 4, the datasets used for training and validation are presented. The accuracy and performance of the neural network solver is assessed in section 5. In section 6 the coupling of the neural network solver with an Euler plasma flow solver is shown in a canonical test case.

Finally, in section 7, a more complex test case of plasma discharge propagation in a cylindrical geometry is tackled using the experience gained from the previous study in cartesian geometry.

2 Problem configuration and solution

The objective is to test the network-based Poisson solver on an academic plasma-fluid problem where analytical solutions exist. To do so, a simple unsteady problem on a rectangular domain is investigated, with boundary conditions and geometry allowing analytical resolution. Note however that the present methodology, as FluidNet [2] for incompressible flows, is not restricted to rectangular domains and simple geometries.

Plasmas are composed of charged species that can be modeled in a fluid formulation [15]. In its simplest form, each charged species i has its own set of Euler equations with electromagnetic source terms (reduced only to the electric field here) and no collision source terms (supposed to be negligible here by assuming a low-temperature as done in [15, Chap. 11.1]):

$$\frac{\partial \mathbf{U}_i}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{F}_i = \mathbf{S}_i \quad \text{for all species} \quad i \tag{2}$$

$$\nabla \cdot \mathbf{E} = \frac{\rho_q}{\varepsilon_0} \tag{3}$$

where

$$\mathbf{U}_i = \begin{bmatrix} \rho_i \\ \rho_i \mathbf{u}_i \\ \rho_i E_i \end{bmatrix} \quad \mathbf{F}_i = \begin{bmatrix} \rho_i \mathbf{u}_i \\ \rho_i \mathbf{u}_i \otimes \mathbf{u}_i + p_i \mathbf{I} \\ (\rho_i E_i + p_i)\mathbf{u}_i \end{bmatrix} \quad \text{and} \quad \mathbf{S}_i = \begin{bmatrix} 0 \\ q_i\, n_i\, \mathbf{E} \\ q_i\, n_i\, \mathbf{E} \cdot \mathbf{u}_i \end{bmatrix} \tag{4}$$

where ρ_i is the mass density, n_i the number density, $\mathbf{u}_i$ the speed, E_i the total energy per mass unit, p_i the pressure of species i. Finally $\rho_q = \sum q_i\, n_i$ is the charge density and $\mathbf{E}$ the electric field. From now on, $R = \rho_q/\varepsilon_0$ is set to alleviate notations. It will be abusively referred to as the charge density althrough strictly speaking the charge density is ρ_q.

In this set of equations, the charged species interact with one another to yield an electric field through the Maxwell-Gauss equation (3). Without magnetic field, the electric field can be written as $\mathbf{E} = -\nabla\phi$ and so Eq. (3) becomes a Poisson equation (1). When discretizing Eq. (1), the resulting linear system requires iterative methods such as Jacobi or Conjugate gradient [1] to be solved. In most plasma-fluid simulations of real applications, this resolution represents up to 80% of the total CPU cost and limits the computation capability.

Eq. (1) must be completed with proper boundary conditions, taken here as zero-Dirichlet conditions, (*i.e.*, imposing $\phi = 0$ on all the boundaries). The equation is computed on a square domain with uniform spacing Δ in the x and y directions. Thus the problem can be recast as

$$\begin{cases} \nabla^2\phi = -R \quad \text{in} \quad \mathring{\Omega} & (5) \\ \phi = 0 \quad \text{on} \quad \partial\Omega & (6) \end{cases}$$

where $\mathring{\Omega}$ refers to the internal computational domain and $\partial\Omega$ to its boundary. Applying Dirichlet boundary conditions on a domain of size (L_x, L_y) allows the derivation of analytical solutions, written in terms of spatial Fourier series (A) as

$$\phi(x,y) = \sum_{n=1}^{+\infty}\sum_{m=1}^{+\infty} \phi_{nm} \sin\left(\frac{n\pi x}{L_x}\right)\sin\left(\frac{m\pi y}{L_y}\right) \tag{7}$$

$$\phi_{nm} = \frac{R_{nm}}{\left[\left(\frac{n\pi}{L_x}\right)^2 + \left(\frac{m\pi}{L_y}\right)^2\right]} \tag{8}$$

$$R_{nm} = \frac{4}{L_x L_y}\int_{x',y'} \sin\left(\frac{n\pi x'}{L_x}\right)\sin\left(\frac{m\pi y'}{L_y}\right) R(x',y')\,\mathrm{d}x'\,\mathrm{d}y' \tag{9}$$

where the profile of charge density $R(x', y')$ may be of any kind.

To compute the analytical solution, spatial Fourier modes R_{nm} of the charge density are first computed Eq. (9). Then the potential Fourier coefficients ϕ_{nm} are derived from R_{nm} with Eq. (8). Thus the resulting potential has a diffused

shape compared to the charge density. An example is given in Fig. 1, showing the potential (ϕ, left) and the electric field ($\mathbf{E}$, middle), obtained with a charge distribution ($R = -\nabla^2\phi$, right) consisting of two Gaussian functions. The two Gaussian peaks, clearly visible on the charge density field, are totally diffused and merged in the resulting potential, highlighting the low-pass filter behavior of the inverse Laplacian operator. Note that this solution is of little practical use: for N (resp. M) modes along the x (resp. y), $N \times M$ integral evaluations Eq. (7) are necessary to compute the potential. For a low-frequency charge density profile such as the two gaussians, $N = M = 10$ modes in each direction are required to get below 1% of error on the 1-norm of the electric field. For high-frequency profiles such as the one depicted in Fig. 6 $N = M = 50$ modes in each direction is necessary so that 2500 domain integrals need to be computed. Therefore this solution can help us understand the structure of the solution and can serve as a reference solution for comparison with neural network predictions, but it is too computationally expensive to be used in a simulation.

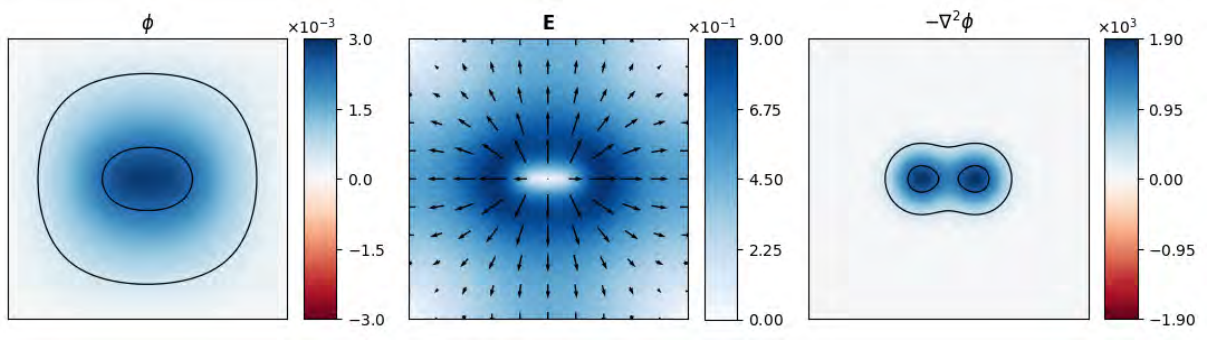


Figure 1: Potential [V] and electric field [V.m^{-1}] (norm in color and direction with arrows) associated to a two-Gaussians charge density [V.m^{-2}] in a 1 cm^2 domain.

3 Neural networks and methods

Deep neural networks are composed of multiple tunable neurons that can learn complex functions. To do so, a non-convex optimization procedure is performed to update the neuron weights by minimizing a cost function (the loss function). This loss function is crucial, and several metrics are presented and tested in this work. Neural networks are denoted by f such that

$$\phi_{\text{out}} = f(R_{\text{in}}) \tag{10}$$

The target solution (potential obtained with a linear solver) is on the other hand denoted by ϕ_{target}.

3.1 Loss functions

Two kinds of loss functions are used: one for the interior points corresponding to $\mathring{\Omega}$ and the other one for the boundary points $\partial\Omega$, as introduced in Eqs. (5)-(6).

In deep neural networks on supervised tasks, pixel-to-pixel distances are often used as loss functions for training and are defined as `InsideLoss` Eq. (12) and `DirichletLoss` Eq. (11). For the `DirichletLoss`, the target value of the boundary points is known from the problem definition (0 in the chosen configuration here). However for the `InsideLoss` a pre-computed target dataset to which the network prediction is compared is needed. An alternative is the `LaplacianLoss` which uses the residual of Eq. (5) and therefore avoids to solve the Poisson equation with linear solvers. Two combinations of losses will be thus tested: `DirichletLoss` - `InsideLoss` and `DirichletLoss` - `LaplacianLoss`.

- `DirichletLoss`:

$$\mathcal{L}_D(\phi_{\text{out}}) = \frac{1}{b_s(2n_x + 2n_y - 4)} \sum_{b,j,i} (\phi_{\text{out}}^{b,j,i})^2 \tag{11}$$

- `InsideLoss`:

$$\mathcal{L}_I(\phi_{\text{out}}, \phi_{\text{target}}) = \frac{1}{b_s(n_x - 1)(n_y - 1)} \sum_{b,j,i} \left(\phi_{\text{out}}^{b,j,i} - \phi_{\text{target}}^{b,j,i}\right)^2 \tag{12}$$

- `LaplacianLoss`:

$$\mathcal{L}_L(\phi_{\text{out}}) = \frac{L_x^2 L_y^2}{b_s(n_x-1)(n_y-1)} \sum_{b,j,i} \left(\nabla^2 \phi_{\text{out}}^{b,j,i} + R_{\text{in}}^{b,j,i}\right)^2 \tag{13}$$

where b_s, n_x, n_y refer to the batch size (number of training examples used in one iteration of learning), number of nodes in x and y directions, b, j, i refer to the indices of the batch size, the y and x directions respectively.

3.2 Network architectures

The analytical derivation Eq. (7) highlighted the low-pass behavior of the inverse Laplace operator. Spatial scales are therefore crucial when solving a Poisson equation, which therefore drives the choice of the neural network: the Multi-Scale (denoted MSNet) architecture [16] and UNet architecture [17] embed the notion of spatial scales, with dedicated treatment of the various scales contained in the input.

The MSNet has been designed for video predictions [16] for which classical deep networks failed to capture accurately the largest scales of the inputs. This architecture has already been used for different flow applications, such as the super-resolution of turbulent flows [18] and the propagation of acoustic waves [19]. Generally speaking this network consists of n_s scale channels (a sketch of a $n_s = 3$ network is presented in Fig. 2). In each channel i, the initial images are downsampled from n_p pixels to $\lfloor n_p/2^i \rfloor$ pixels per direction. From now on the image resolution is either defined in terms of spatial spacing Δ or number of pixels (n_p). A series of convolutional layers is applied and the output of channel i goes to channel $i-1$ except for the $i = 0$ scale channel.

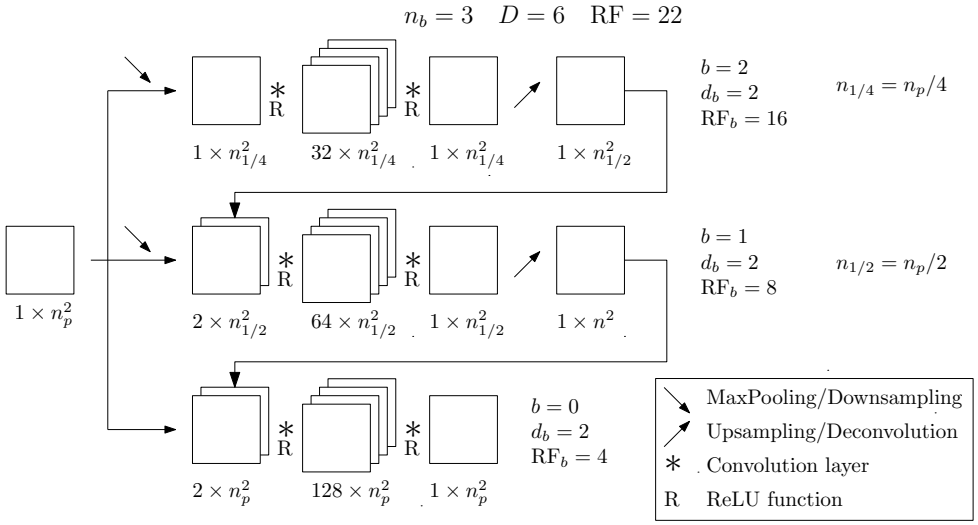


Figure 2: Sketch of MSNet with $n_s = 3$.

The UNet was first introduced for biomedical segmentation [17], and it has been widely used by the ML-CFD community [20, 21]. A sketch is presented in Fig. 3. A series of encoding layers are applied, each time decreasing the number of pixels by a power of 2 in each direction as in the MSNet. What differentiates the UNet from a simple encoder-decoder network is the skip connection which links every enconding layer to its decoding counterpart.

In both MSNet and UNet, the local branch b is defined as the power of 2 by which the initial resolution is divided by, in a specific scale of the network. The number of branches n_b of the network is the number of scales involved ($n_b = n_s$).

The local depth d_b is defined as the number of successive convolutional layers in branch b. The global depth D of the network is then the sum of the local depths, assuming that D corresponds to the longest path from input to output. The higher the depth of the network, the more likely vanishing or exploding gradients can appear.

$$D = \sum_{b=0}^{n_b-1} d_b \tag{14}$$

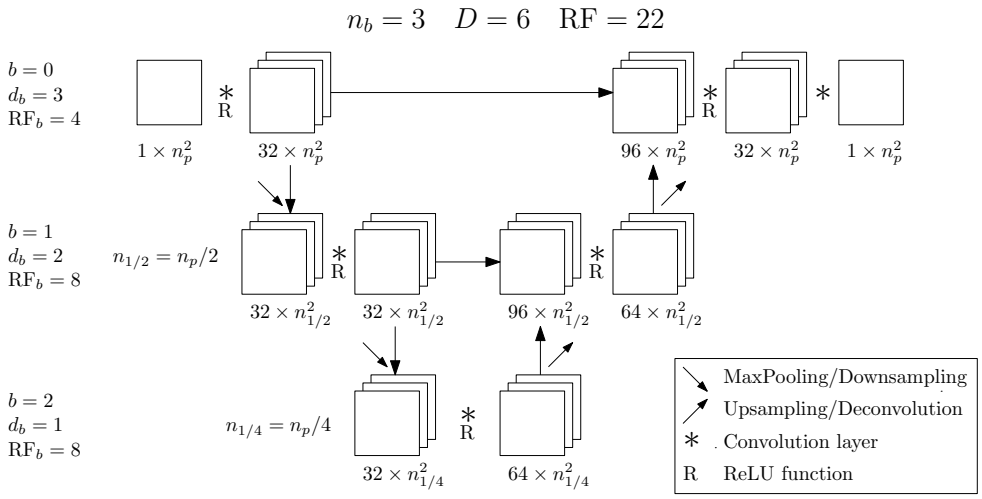
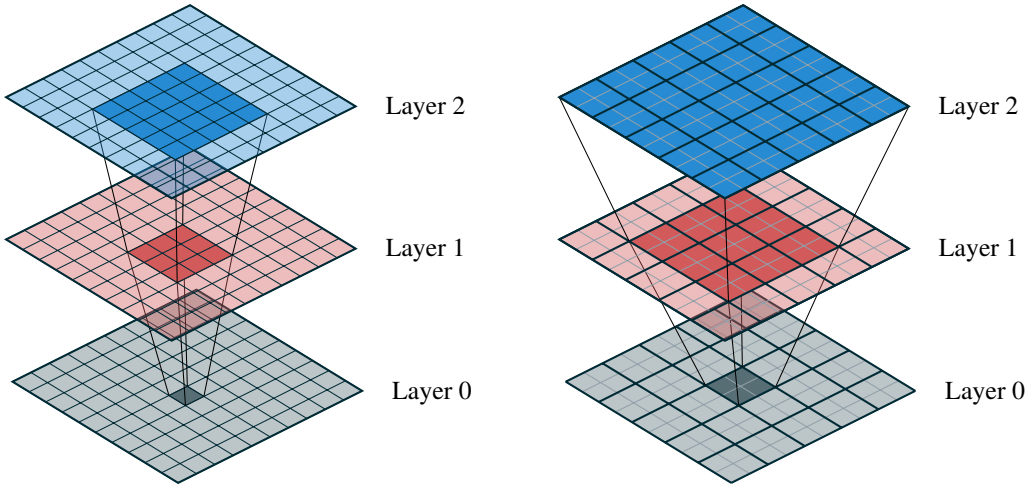


Figure 3: Sketch of UNet with $n_s = 3$.

3.2.1 Receptive field

In a convolutional neural network (CNN), from one layer to the next, information propagates according to the kernel size k_s of the convolution layer. Typically, one layer with $k_s = 3$ gathers information from a neighboring of 3×3 pixels only, and therefore cannot capture large structures. However, when several layers of convolution are applied, points that are far away from each other in the input image can interact with each other, which allows larger structures to be detected. As an illustration, the domain of influence of the center point of an image with two convolutional layers at different branches b is depicted in Fig. 4. To help capturing the largest scales, MSNet and UNet employ downscaled branches: by downscaling the input image by a factor 2, the second branch in Fig. 4b tackles flow structures twice as big as the first branch in Fig. 4a.



(a) $b = 0 \quad d_b = 2 \quad k_s = 3$ (b) $b = 1 \quad d_b = 2 \quad k_s = 3$

Figure 4: Domain of influence of the center point across two convolutional layers with a kernel size of 3.

To quantify the information propagation the global receptive field RF is defined as the size of the domain of influence of the input center point in number of points in the original scale n_p. The receptive field can be splitted into local receptive fields per branch RF_b so that

$$\mathrm{RF} = \sum_{b=0}^{n_b-1} \mathrm{RF}_b \tag{15}$$

with

$$\mathrm{RF}_b = \begin{cases} 1 + d_b(k_s - 1)2^b & \text{if } b = 0 \\ d_b(k_s - 1)2^b & \text{otherwise} \end{cases} \tag{16}$$

where the branch $b = 0$ needs to take into account the original input pixel (+1) and k_s is the kernel size of all the convolutional layers of the network (supposed to be equal). In Fig. 4a, $b = 0$, $k_s = 3$ so that every layer extends the receptive field by $(3 - 1)2^0 = 2$ and the branch receptive field is $\mathrm{RF}_0 = 1 + 4 = 5$. In Fig. 4b, $b = 1$, $k_s = 3$, the initial RF from previous branches is equal to 2 ($\mathrm{RF}_0 = 1$) and the grid that is actually processed by the convolutional kernel is 2 times coarser (black thick line) compared to the orignial grid (thin gray line). Hence every layer extends the receptive field by $(3 - 1)2^1 = 4$ which is 2 times more than the previous branch yielding a receptive field of $\mathrm{RF}_1 = 8$ for that branch and the total receptive field in the end is $\mathrm{RF} = \mathrm{RF}_0 + \mathrm{RF}_1 = 10$ which corresponds to the size of layer 2.

In [22], a *theoretical receptive field* is defined as the size of the input domain of influence on the output center point. Tests carried on the studied UNets and MSNets show that this definition matches Eq. (15) so that both formulations are equivalent. However, the importance of the domain influence is not uniform, as points closer to the studied pixel will have more paths to influence the output, resulting in a *gaussian-like* distribution [23]. The distribution helps to understand the behavior of the CNN, as it will especially focus on a smaller centered region, known as *effective receptive field*, while still being influenced by information located on the boundaries of the *receptive field*.

One convolutional layer in branch b contributes two times more to the receptive field than a convolutional layer in branch $b - 1$. The local and global properties defined in this section are reported for the simple MSNet3 and UNet3 in Fig. 3.

The receptive field analysis is particularly of interest in view of the elliptic nature of the Poisson equation, which does not load to characteristic lines and propagates information only spatially, every point of the domain influencing the whole domain. This suggests that the neural network should see the whole input image to predict correctly the solution.

3.3 Normalization

A neural network learns well from inputs and outputs that span approximately the same range of values [24]. For example in image prediction, the network needs to output a field of values in the interval [0 ,1] from values that are also in [0, 1] [25]. In the present case, the potential maximum value is not known *a priori*, so that only the density charge is rescaled as

$$\phi_{\mathrm{out}} = f(\tilde{R_{\mathrm{in}}}) \quad \text{where} \quad \tilde{R_{\mathrm{in}}} = R_{\mathrm{in}} \times \left| \frac{\phi_{\mathrm{out}}}{R_{\mathrm{in}}} \right|_{\mathrm{max}} \tag{17}$$

which ensures maximum values of the input and output of the same order.

A reasonable value for the ratio of the potential over the charge density needs to be found. From the solution of the potential in terms of Fourier series, the following normalization is chosen:

$$\left| \frac{\phi_{\mathrm{out}}}{R_{\mathrm{in}}} \right|_{\mathrm{max}} = \frac{\alpha}{\left(\frac{\pi^2}{4}\right)^2 \left(\frac{1}{L_x^2} + \frac{1}{L_y^2}\right)} \tag{18}$$

where $\alpha = 0.1$ is used. More details on the derivation of this relation can be found in B. This value is actually correlated with the FWHM (Full Width Half Maximum) of the charge density R. The higher the value of the FWHM, the higher the maximum value of the potential ϕ.

Besides bringing the values of the input and output of the networks to the same orders of magnitude, this normalization also brings similarity in domain-length: the normalized solution in a square box of length L_x and resolution Δ is similar to the one in a square box of length αL_x and resolution $\alpha\Delta$.

3.4 Resolution scaling

After the training of the network in a square box of length L_x and resolution Δ_1 the problem of the applicability of this network to a square box of length L_x and resolution Δ_2 is discussed in this section.

The Laplacian operator with a resolution Δ can be nondimensionalized:

$$\nabla^2_\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} = \frac{1}{\Delta^2}\left(\frac{\partial^2}{\partial \bar{x}^2} + \frac{\partial^2}{\partial \bar{y}^2}\right) \tag{19}$$

where the overbar indicates nondimensionalized physical values. The nondimensionalized operator should be the conserved quantity between two resolutions. Denoting by Δ_{sim} the resolution of the simulation and by Δ_{NN} the resolution at which the neural network was trained, the following relationship holds:

$$\nabla^2_{\Delta_{\text{sim}}} = \frac{\Delta^2_{\text{NN}}}{\Delta^2_{\text{sim}}} \nabla^2_{\Delta_{\text{NN}}}. \tag{20}$$

What the network is emulating is in fact the inverse of the Laplacian, hence:

$$(\nabla^2_{\Delta_{\text{sim}}})^{-1} = \frac{\Delta^2_{\text{sim}}}{\Delta^2_{\text{NN}}} (\nabla^2_{\Delta_{\text{NN}}})^{-1} \tag{21}$$

so that the initial neural network guess needs to be multiplied by a ratio of resolutions to be applied to other resolutions.

4 Datasets

In the present work, two types of datasets are proposed, where the spatial scales can be controlled. The objective is to better understand how the neural network can learn, and then predict accurately the various scales of the solution in the context of the Poisson equation. To do so, two types of datasets will be generated: (i) a random dataset, and (ii) a random Fourier dataset.

4.1 Random dataset

First proposed by Ozbay [12], a random distribution of values in the range $[-1, 1]$ is generated in a coarse grid of size $n_{\text{coarse}} = \lfloor n_p/c \rfloor$, n_p being the number of points in each direction and c a chosen filter size. Then bicubic interpolation generates a random field with controlled structure size on the target grid. The minimum structure is of size c pixels. This procedure is illustrated for $c = 16$ in Fig. 5. From now on, `random_c` will be used to denote c-random datasets.

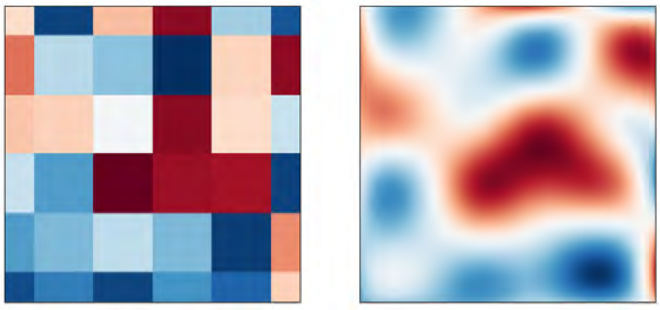


Figure 5: Random values taken in a 6×6 coarse grid (left) and interpolated in a 101×101 fine grid (right) where $c = 16$.

An example of $(\phi, \mathbf{E}, R = -\nabla^2\phi)$ is shown in Fig. 6 for a random dataset filetered with $c = 8$.

4.2 Random Fourier dataset

Also proposed by Ozbay [12], a random Fourier dataset is obtained by setting randomly the coefficients R_{nm} of Eq. (9), and introducing maximum frequencies (N and M) in the sums:

$$A(x, y) = \sum_{n=1}^{N} \sum_{n=1}^{M} A_{nm} \sin\left(\frac{n\pi x}{L_x}\right) \sin\left(\frac{m\pi y}{L_y}\right) \quad \text{for} \quad A \in \{\phi, R\} \tag{22}$$

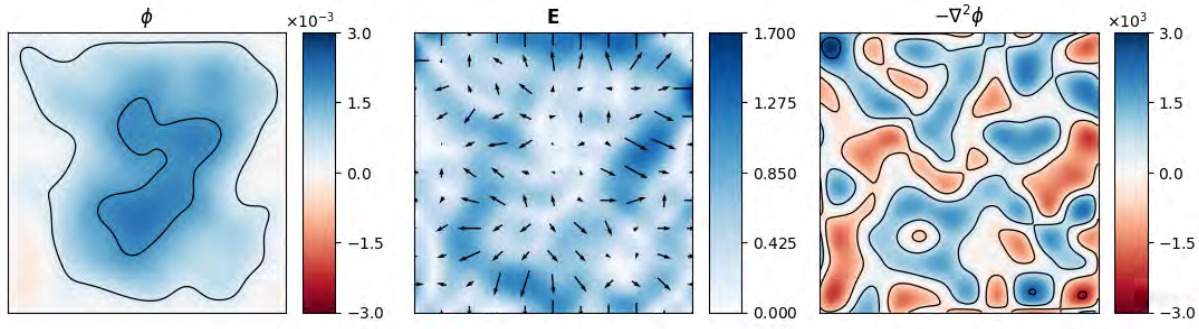


Figure 6: Example of `random_8` source term input in a 1 cm^2 domain.

where ϕ_{nm} is deduced from R_{nm} thanks to Eq. (8).

The value of R_{nm} are taken randomly following a power p decreasing law to mimic the high-frequency damping of physical solutions:

$$R_{nm}(p) \sim \frac{1}{n^p + m^p} \frac{en_0}{\varepsilon_0} \mathcal{U}(-1, 1). \tag{23}$$

where $\mathcal{U}(-1, 1)$ corresponds to a uniform distribution over the range $[-1, 1]$.

One example of a Fourier dataset is shown in Fig. 7. Note that in this case, the low number of modes of the dataset ($N = M = 3$) allows a clear correlation between the potential and the charge distribution contrary to Fig. 6. This dataset allows therefore to understand the frequency response of the network by selecting particular frequencies. From now on, `fourier_N_p` will be used to denote (N, p)-Fourier datasets.

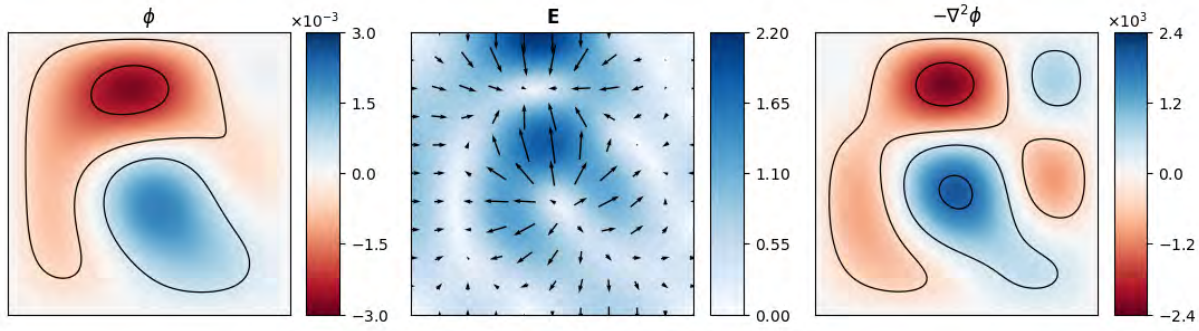


Figure 7: Example of Fourier dataset item with $N = M = 3$ modes and $p = 0$ in a 1 cm^2 domain.

5 Single frame inference of the potential and electric field

This section focuses on the training and prediction of the single Poisson solution for the potential and electric field, *i.e.*, without coupling with the unsteady plasma Euler equations. The behavior of the networks when changing the architecture, loss function and dataset is evaluated. Training was typically performed over 300 epochs with 101×101-resolution datasets containing 10 000 snapshots (8000 for training and 2000 for validation) in a 1 cm^2 domain. All the networks are constructed with around 100 000 parameters, where the number of filters per layer is changed to approximately match this number when the number of branches is changed. All evaluations are performed at epoch 300. A summary of the parametric study is given in Tab. 1. All computations were carried out using in-house Nvidia Tesla V100 SXM2 32 Gb GPUs.

5.1 Metrics

To monitor the accuracy of the networks the 1-norm and infinity norm residuals are used:

Architecture	UNet, MSNet
Number of branches	3, 4, 5
Receptive field	50, 75, 100, 150, 200
Number of parameters	100 000
Training snapshots	10 000
Training resolution	101×101

Table 1: Overview of the parametric study

$$\left\|u_{\text{out}} - u_{\text{target}}\right\|_1 = \frac{1}{n}\sum_i \left|u^i_{\text{out}} - u^i_{\text{target}}\right| \tag{24}$$

$$\left\|u_{\text{out}} - u_{\text{target}}\right\|_\infty = \max_i \left|u^i_{\text{out}} - u^i_{\text{target}}\right| \tag{25}$$

for $u \in \{\phi, \mathbf{E}\}$ and where the index i spans all the relevant sizes of batch, dimension and directions x and y. The networks are evaluated on `random` and `fourier` datasets where each evaluation dataset contains 1000 snapshots. One network trained on `random_8` snapshots and evaluated on the batch of datasets is shown in Fig. 8. The accuracy of the network on Fourier datasets slightly decreases with an increasing number of modes N and is also observed when c increases for the random datasets. Overall both metrics give similar levels on all the datasets and are on the same order of magnitude not showing overfitting on the training dataset. From now on datasets are not expanded and a combined dataset evaluation is implied, *i.e.*, the accuracy of the network is evaluated on the concatenation of the 6 datasets show in Fig. 8.

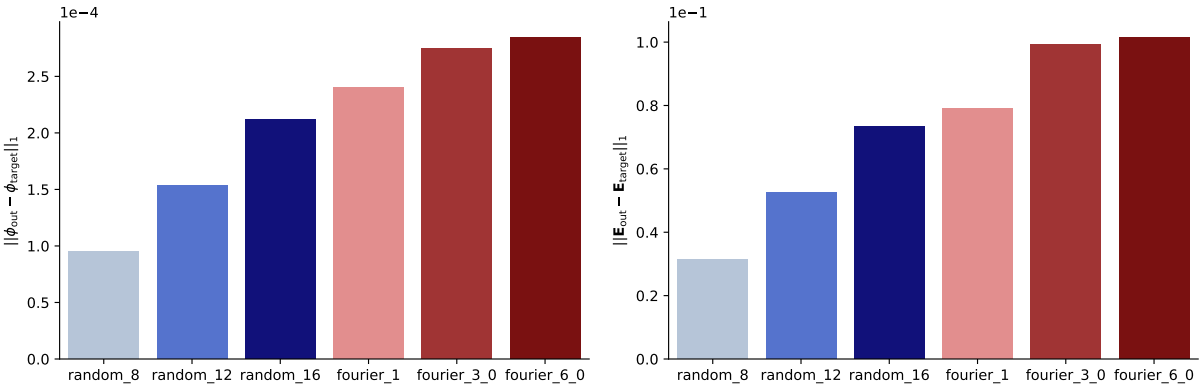


Figure 8: Datasets evaluation for 1-norm error of the potential (left) and electric field (right) with a UNet of $n_b = 3, k_s = 3, \text{RF} = 100$.

Trainings have been carried out with both `random` and `fourier` datasets separately. `random`-trained networks are unambiguously better with both metrics and losses than `fourier`-trained networks. `random` datasets seem to contain more information about the Poisson equation and have therefore been preferred for training in all the following discussions while `fourier` datasets have been retained for *a posteriori* analysis.

5.2 Physical loss

The `LaplacianLoss` depends only on the input and output of the network and does not need any target value. It is supplemented by the `DirichletLoss` to give a reference for the potential.

In a numerical simulation, the quantity of interest is the electric field $\mathbf{E}$ which as a derivative of the potential requires sufficient smoothness of the potential solution. Moreover, the electric field appears as a source term in the species i momentum equation, which in turn impacts the species i mass density (first term of $\mathbf{F}_i$). As a consequence, second-order smoothness on the potential must be ensured for the simulation to be stable.

Comparing the application of the `InsideLoss` and `LaplacianLoss` in training in Fig. 9 shows a better accuracy on the 1-norm and infinity-norm (not shown here but similar) on the potential with `InsideLoss`. On the electric field

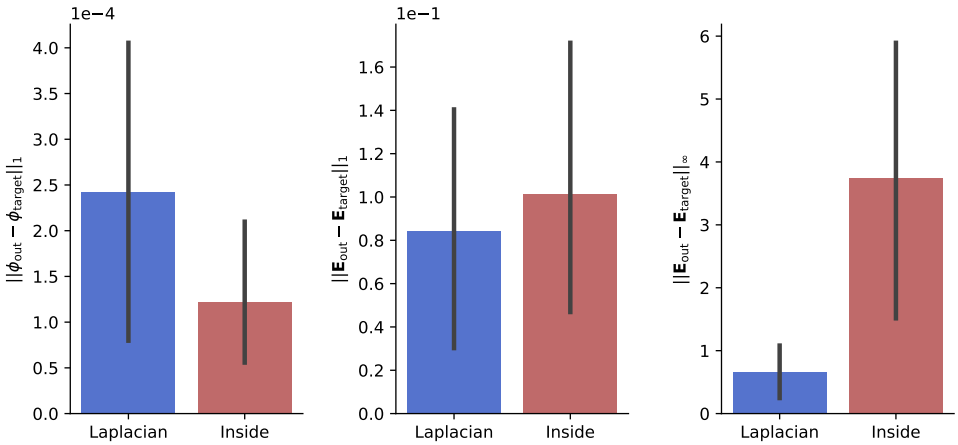


Figure 9: Comparison of `InsideLoss` and `LaplacianLoss` across `random_8`-trained UNets with number of branches from 3 to 5 and RFs from 50 to 200. 1-norm of the potential (left), 1-norm (center) and infinity-norm (right) of the electric field.

1-norm, both losses are comparable with slightly better accuracy for the `LaplacianLoss`. However the infinity-norm of $\mathbf{E}$ obtained with the `InsideLoss` is one order of magnitude greater than with the `LaplacianLoss`. This indicates few points where the electric field is unphysical and cannot be tolerated in the numerical simulation of the Euler plasma equations.

The inference of the same network, either trained with `LaplacianLoss` or with `InsideLoss` highlights the smoothing effect of the `LaplacianLoss`, while the Laplacian of the infered potential for an `InsideLoss`-trained network is completely unphysical. Comparing Fig. 10(top) with Fig. 1 there is a very good agreement between the `LaplacianLoss` and linear solver solution. Consequently `LaplacianLoss` with `DirichletLoss` are chosen from now for all subsequent cases.

5.3 UNet vs MSNet

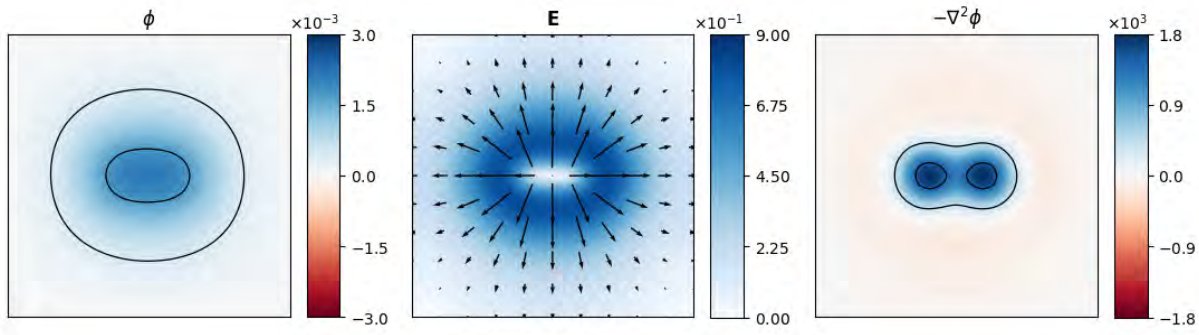
UNet and MSNet architectures are compared in this section. The same number of parameters (around 100 000) and the same receptive fields and numbers of branches are used for both architectures. Comparison is shown in Fig. 11. From these results, it appears that UNet architectures are better suited than MSNet architectures for the problem at hand over all parameters used. Thus only UNet architecture will be considered in the following.

One way to explain this difference may be found in the way each architecture goes from one scale to the other. MSNet has been designed to make video prediction [16] from one frame to the other and understand local movement, so only local information propagates from one snapshot to the other. Looking at Fig. 2, MSNets compact the information of one scale in one feature map before inserting it to the next scale. On the other hand, UNets in Fig. 3 apply a skip connection as well as an upsampling at the end of every intermediate scale, always keeping all relevant information.

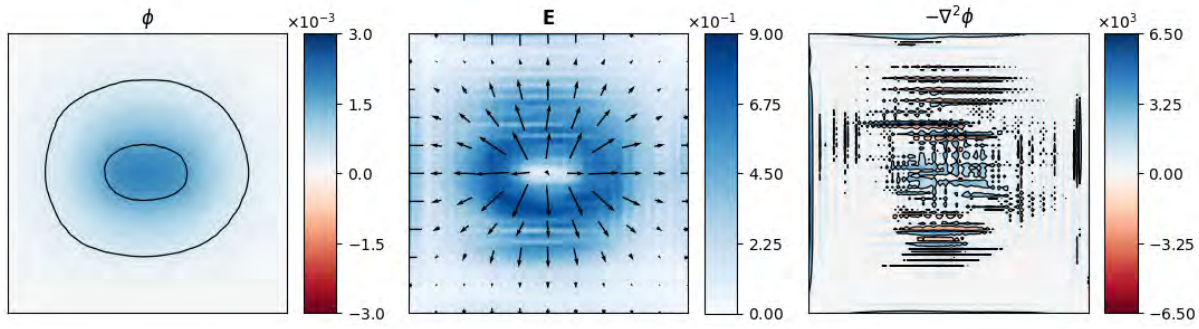
5.4 Receptive field

Figure 12 shows the evaluation of UNet on the same resolution as the trained resolution, containing 3 to 5 branches, where the receptive fields vary from 50 to 200. Due to the definition of the receptive field, RF = 100 corresponds to the situation where the middle point of the input images can influence the whole output solution. The boundary pixels, however, do not influence the whole domain yet but only one quarter. Only when the receptive field reaches 200, any point of the input image influences the whole output domain, including the boundaries.

A first look at the results shows that networks with the same receptive have a similar behavior whatever the number of branches and depth, which therefore do not influence by themselves the network performance. This can be understood by looking at the structure of the Poisson equation: elliptic differential equation solutions need the information of the whole domain at every point. This elliptic nature is highlighted in the analytical solution of the problem where domain integrals are present to compute the Fourier coefficients of the charge density R_{nm} Eq. (9).



(a) `LaplacianLoss`



(b) `InsideLoss`

Figure 10: Potential [V] and electric field [V.m^{-1}] (norm in color and direction with arrows) associated to a two-Gaussians charge density [V.m^{-2}] in a 1 cm^2 domain for UNet with $n_b = 3$, RF = 100. Top: `LaplacianLoss`. Bottom: `InsideLoss`.

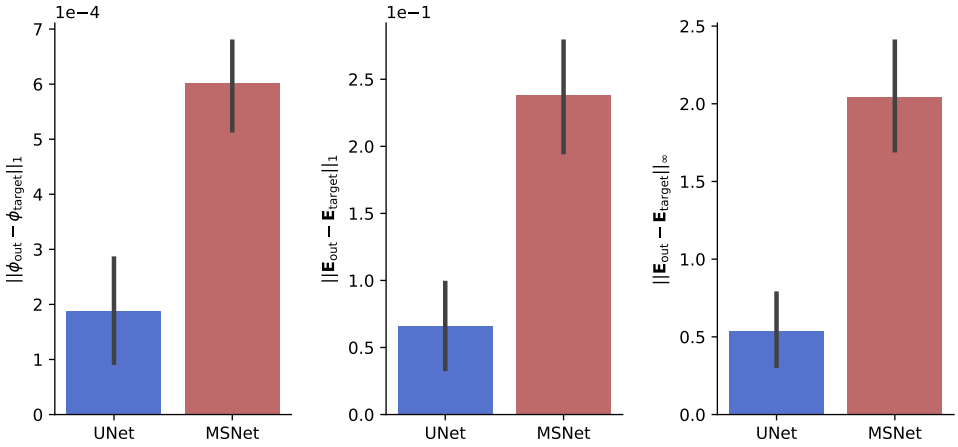


Figure 11: Comparison of MSNet and UNet architectures for different receptive fields [50-200] and numbers of branches [3-5]. 1-norm of the potential (left), 1-norm (center) and infinity-norm (right) of the electric field.

Moreover, due to the potential damping Eq. (8) in n^2 and m^2, the low frequencies have the highest amplitudes. So it is critical for the network to be able to capture the whole domain when going through convolutions hence the importance of the receptive field. Thus accuracy improves when the receptive field increases because the network is able to capture the dominant longer wavelength content.

Accuracy is similar for different number of branches when keeping the receptive field constant. There is however a performance gain in higher number of branches networks. The convolutions are applied on lower resolution images, decreasing substantially the inference time at fixed number of parameters.

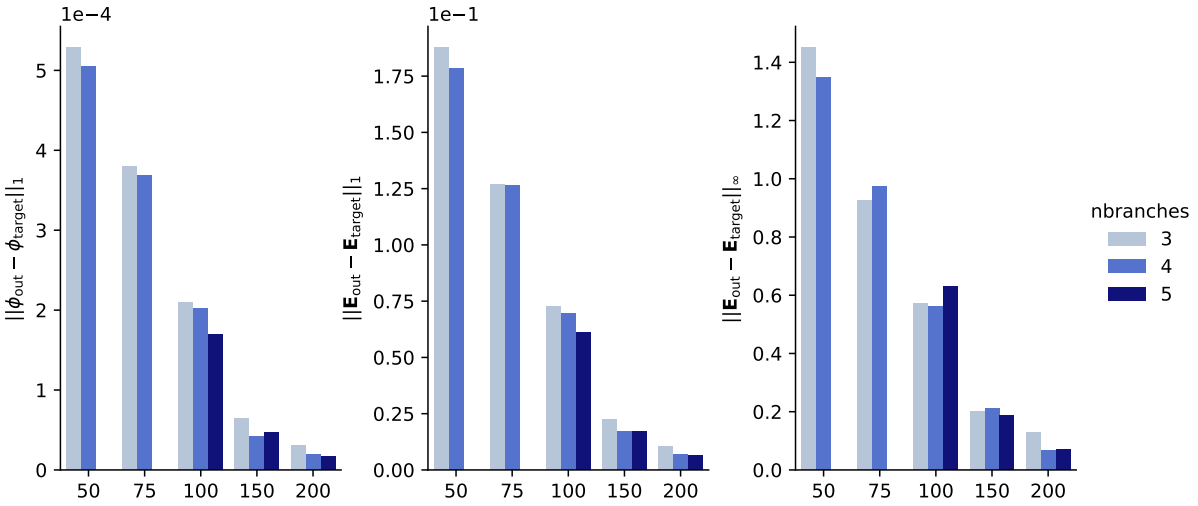


Figure 12: Error metrics of UNet for different receptive fields and numbers of branches from 3 to 5 branches and receptive fields from 50 to 200. 1-norm for the potential (left), 1-norm (center) and infinity-norm (right) for the electric field.

A Fourier decomposition of the network outputs and target potential field is carried out to confirm this interpretation. The amplitudes of the first two modes are shown in Fig. 13. The amplitude of the fundamental mode ϕ_{11} follows the same trend as the other metrics and is at least an order of magnitude greater than the other first modes ϕ_{12}, ϕ_{21}, ϕ_{22} indicating that it drives the errors. Having a high receptive field should mostly impact all modes with $n = 1$ or $m = 1$, *i.e.*, modes that contain a wavelength equal to the length of the domain.

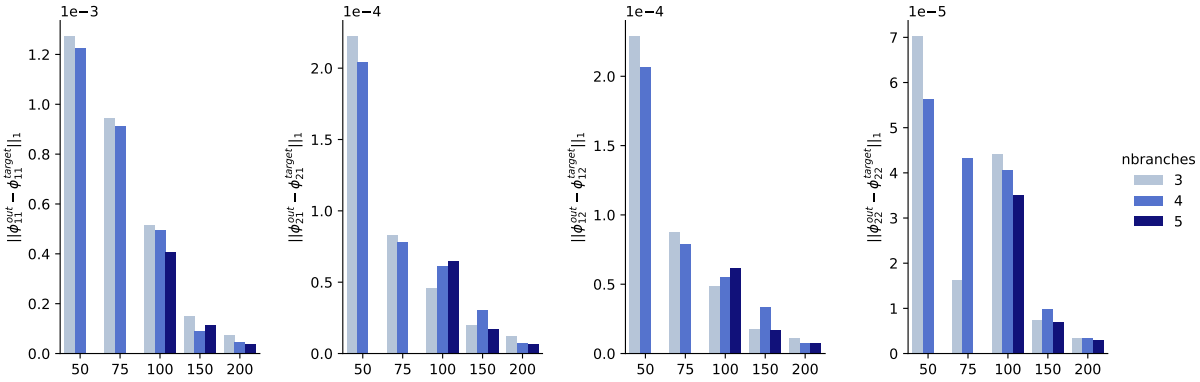


Figure 13: Error of amplitude of the first two modes of UNet solution for different receptive fields and numbers of branches.

As explained in the beginning of the subsection, the monotonic decrease of the residuals from RF = 50 to RF = 200 is due to saturation of the domain of influence of every point. No significant increase of accuracy should be observed for higher receptive fields as the fundamental mode would not be better captured. This is shown in Fig. 14 for UNet5 from RF = 200 to RF = 400.

As discussed previously networks with higher number of branches have faster resolution times due to faster convolutions when the number of points is lowered in the downscaled branches. There is however a limit to the number of branches due to the lowest resolution branch. The size of the lowest resolution branch images ($n_p/2^{n_b-1}$) should be greater than the kernel size (k_s) of the convolutional layers of that branch otherwise no information propagation is taking place (i.e. the most downscaled input image is smaller than the kernel size). For the 101×101 resolution images this means that UNet5 is an optimum because $\lfloor 101/2^4 \rfloor = 6 > k_s = 3$ whereas for UNet6 $\lfloor 101/2^5 \rfloor = 3 = k_s$. Thus adding a new branch at fixed number of network parameters is detrimental for the accuracy of the network. In the case of the UNet6, weights and biases used in the last branch ($b = 5$) are useless because no relevant information can be extracted from this scale. This is demonstrated in Fig. 14 where the accuracy of the UNet6 is significantly less than UNet5 at the same receptive field. A gain of accuracy is observed for UNet6 when the receptive field increases whereas it is constant for UNet5. Since the weights of branch $b = 5$ are meaningless, the UNet6 acts as a reduced UNet5 with less parameters and a lower receptive field. The contribution to the receptive field of branch 5 RF_5 of the three UNet6-RF200/300/400 are respectively: 64, 128 and 192. The resulting effective receptive fields $\mathrm{RF}_{\mathrm{eff}} = \mathrm{RF} - \mathrm{RF}_5$ are thus 136, 172 and 208 and the increased accuracy can be explained following the same reasoning as the beginning of the section.

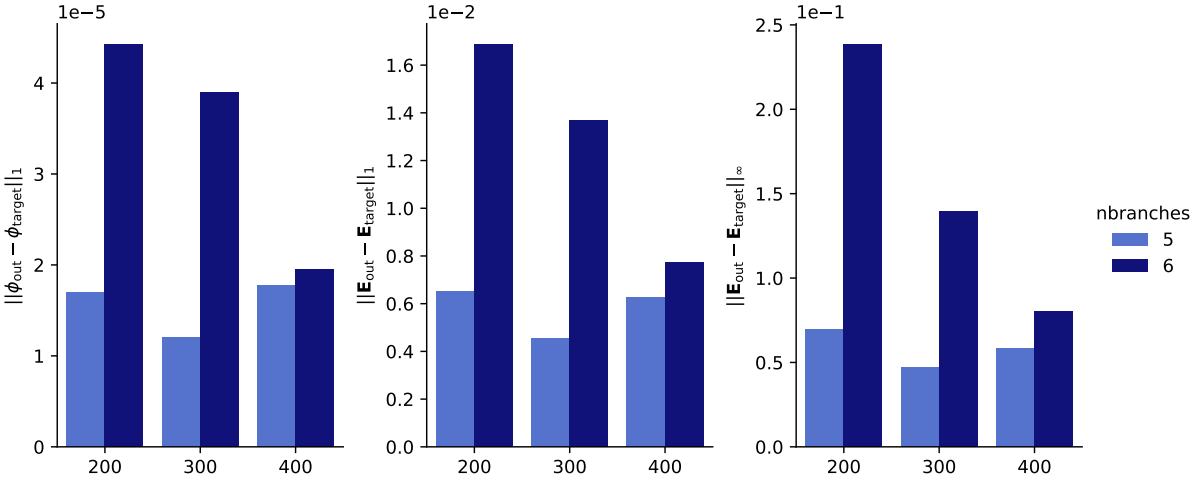


Figure 14: Error metrics of UNet for different receptive fields with high number of branches.

To sum up, for maximizing accuracy the receptive field should be chosen to saturate the domain of influence of any input point. That way the network correctly captures the low spatial frequencies which are dominant in most of the real-engineering physical fields. For performance optimization, the number of branches should be maximized as long as there is meaningful information in the downsampled branches (domain size bigger than the kernel size). Hence the optimal global parameters of the network for a given number of pixels n_p should be:

$$\mathrm{RF} = 2n_p \tag{26}$$

$$n_b = \max\{b \in \mathbb{N} | \lfloor n_p/2^b \rfloor > k_s\} + 1 \tag{27}$$

5.5 Resolution invariance and spectral analysis

Thanks to the resolution scaling in Section 3.4, the network is able to work on resolutions different from the training resolution. Spectral analysis on different resolutions from a network trained on a single resolution is conducted in this section, where the optimum UNet5 with RF = 200 trained on 101×101 resolution images is used.

The sine modes Eq. (22) for which the exact solutions are known are used to conduct the spectral analysis. Each mode is studied separately so that only one term A_{nm} of the double sum in Eq. (22) is taken. The 1-norm residuals of the neural network potential and electric field of the UNet5-RF200 are shown in Fig. 15 for different values of (n, m) as functions of the domain resolution.

The minimum residual is found at the trained resolution of 101 for both metrics. At that resolution, the $(n, m) = (1, 1)$ mode error is for the potential and electric field more than one order of magnitude higher than the $(n, m) = (10, 10)$ mode error. This tendency remains true at other resolutions so that the longer the wavelength the harder it is for the

network to capture it correctly. Moreover, the frequency response of the network when increasing the resolution is not the same: the loss of accuracy for shorter wavelengths (red curves) is lower than higher wavelengths (blue curves).

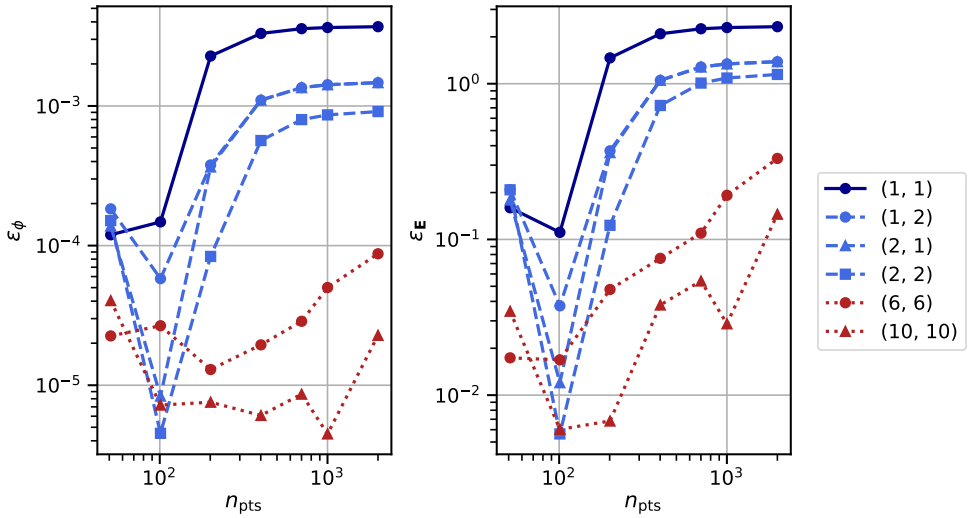


Figure 15: 1-norm of the potential (left) and electric field (right) residuals for different modes (n, m) and different resolutions with UNet5, RF = 200.

The optimal network can thus work on resolutions that are different from the trained resolution, but errors grow as the tested resolution differs from the training one. One needs to keep in mind the loss of accuracy which is higher for longer wavelengths than shorter wavelengths. This loss of accuracy for resolutions that differ from the training resolution could be compensated either by interpolating the inference domain to the training resolution, by introducing a hybrid strategy [26] which combines the network prediction with traditional iterative solvers to ensure a user-defined accuracy level or by training the network on multiple resolutions. However, these methods count with their own drawbacks that limit their use. On the one hand, interpolating the domain to the training resolution is a computationally expensive process, which considerably increments the time taken to complete the simulation. Moreover, even if the network accuracy increases on the interpolated domain, the interpolation introduces high frequency oscilations which are amplified by the network resulting in more unstable simulations. On the other hand, a hybrid strategy is suited to mitigate high frequency errors, as the charge field is locally difussed. However, iterative Jacobi solvers are not suited to cope with errors associated with long wavelengths, especially on high resolution domains. To correct the error related to low frequency modes, the number of needed Jacobi solver iterations is too high, considerably increasing the simulation time.

Note that finding strategies to make CNN work on multiple resolutions is still an open topic, which requires effort [12, 27] and is out of the scope of the present paper. However, the guidelines obtained in this study, highlighting the key role of the receptive field and number of downscaled branches, are a first attempt to better understand how the CNN architecture learns the spatial distribution of the outputs. These guidelines could be reused to build efective CNN methods able to generalize on variable resolutions.

5.6 Neural network performance

Neural networks run best on GPUs whereas linear system solvers have been historically run on CPUs. To assess the neural network performance against classical linear solvers, CPU and GPU performances need to be compared. The methodology applied here is as follows: given a computational node containing CPUs and GPUs, the speedup when activating or not the GPUs compared to the use of all the CPUs available in one node is assessed. This indicates the potential speedup that the neural network can provide compared to a classical linear system solver running on the same CPUs. Code to run all the benchmarking presented in the following is available at `https://gitlab.com/cerfacs/plasmanet`.

Two configurations have been used in our local cluster: `config_1` is a bi-socket Intel node with 2 x 18 core Xeon Gold 6140 (2.3 Ghz clock speed and 96 Gb memory) interacting with 4 NVIDIA V100 32 Gb GPUs where only one of the four GPUs is used in this study. The second configuration `config_2` is a bi-socket AMD node with 2 x 64 core EPYC Rome 7702 (2 Ghz clock speed and 512 Gb of memory) interacting with a single NVIDIA A100 40 Gb GPU.

Therefore following the methodology, we use all the cores available in one computational node (36 for `config_1` and 128 for `config_2`) to assess the minimal resolution time of linear Poisson solvers using PETSc [28].

The Dirichlet boundary conditions Poisson problem on the 2D square of 1 cm^2 with two Gaussian charge density is used to compare the linear system solver and neural network solver performances. In the case of the linear system solver, the matrix has been symmetrized so that it is positive symmetric definite. Various linear solvers have been tested and results are presented in C. The Conjugate Gradient (CG) method [29, Chap. 6.7] as iterative solver and HYPRE BoomerAMG [30] preconditioner is the highest performing option in this case. To get closer to the accuracy of the neural network solver, the relative tolerance of the iterative solvers has been raised to 10^{-3} where a 4 times speed up is observed compared to a 10^{-12} relative tolerance as shown in Fig. 16 for `config_1`. All the execution times shown with PETSc or the neural network solver are averages taken over 20 resolutions.

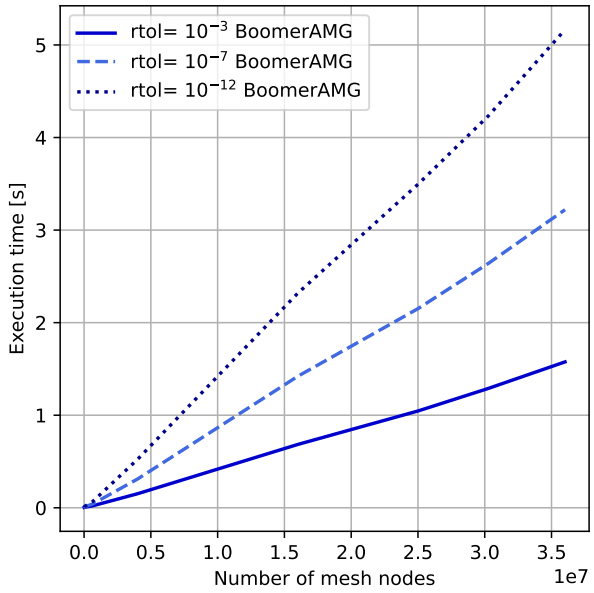


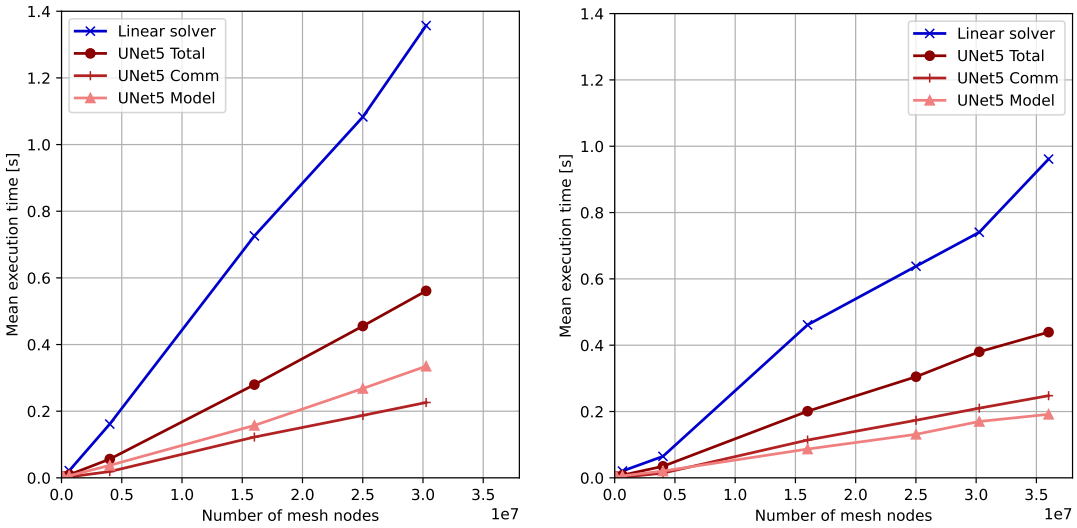
Figure 16: Performance of linear solver when raising the relative tolerance for different AMG solvers on `config_1`.

The neural network performance compared to PETSc linear system solver using CG-BoomerAMG is shown in Fig. 17a on `config_1` where the total execution times, model inference times and communication times of the network are shown. At high number of mesh nodes, the linear solver run time on 36 cores is higher than the neural network total run time by a factor of 2.5 for 30×10^6 mesh nodes. Concerning the GPU performance, the communication time increases with the number of nodes and becomes a significant part of the execution time, as expected. Note that only one of the four GPUs available on the computational node has been used as inference on multi-GPUs is not implemented for the neural network. The maximum resolution of 5501×5501 corresponds to the maximum memory of the GPU at hand (32 Gb) and depends on the hardware available. This is a clear limitation of the neural network solver as it is much more memory consuming than the classical linear system solver: the UNet5 network architecture used at 5501×5501 resolution use up around 30 Gb whereas a single float 64 array of 5501×5501 is around 200 Mb.

Results for `config_2` are shown in Fig. 17b. The GPU memory is higher (40 Gb against 32 Gb for V100 GPU) and the number of cores available as well (128 against 36 for `config_1`). The higher memory allowed the inference of a finer resolution at 6001×6001 (the point at 3.6×10^7 number of nodes in Fig. 17b). For this configuration, the resolution time of the neural network running on the A100 GPU is about a factor 2 lower than the linear system solver on 128 cores, making it a viable option in terms of performance.

Comparing the V100 and A100 GPUs, a similar communication time, *i.e.* the time taken by the CPUs to send the data to the GPU, is observed. However the model time, *i.e.* the application of the neural network on the GPU, is about two times faster for the A100 GPU compared to the V100 GPU.

Thus, deep neural networks are shown to effectively accelerate simulations. This study paves the way for future ones to further analyze and improve inference times and memory cost of the neural networks.



(a) `config_1` (b) `config_2`

Figure 17: Performance of linear solver (CG-BoomerAMG) against neural network solver.

6 Neural networks and canonical plasma fluid simulation

From the previous section, at a resolution of $n_p = 101$ pixels in each direction, a 100 000 parameter UNet architecture trained on a `random_8` dataset, with receptive field RF $= 5$ and number of branches $n_b = 5$ using a combined `LaplacianLoss-DirichletLoss` is chosen for best network accuracy and performance. The target test case corresponds to the plasma oscillations in a square domain, in order to have analytical solutions of this space-time evolving plasma problem to assess both accuracy and performance of the proposed method. The network solves the Poisson equation in place of the linear system solver coupled to the plasma Euler equations. The performance of both solvers is fully analyzed.

6.1 2D plasma oscillation test case

One of the fundamental properties of plasmas is to maintain electric charge neutrality at a macroscopic scale under equilibrium conditions. When this macroscopic charge neutrality is disturbed, large Coulomb forces come into play and tend to restore the macroscopic charge neutrality [15, Chap. 11.1].

Electrons and positive ions with charge e are considered. Ion motion is neglected since its mass is way larger than that of the electrons. A very small electron density perturbation n_e is initialized such that:

$$n_{\text{electron}}(\mathbf{r}, t) = n_0 + n_e(\mathbf{r}, t) \tag{28}$$

$$n_{\text{ion}}(\mathbf{r}, t) = n_0 \tag{29}$$

where n_0 is a constant number density and $|n_e| \ll n_0$. Linearization of the momentum equation, combined with the mass equation and the Maxwell-Gauss equation [15, Chap. 11.1] yields:

$$\frac{\partial^2 n_e}{\partial t^2} + \omega_p^2 n_e = 0 \quad \text{where} \quad \omega_p = \sqrt{\frac{n_e e^2}{m_e \varepsilon_0}} \tag{30}$$

The electron density varies harmonically in time at the electron plasma frequency $f_p = \omega_p / 2\pi$, or oscillation period $T_p = 1/f_p$ and it can be shown that the electric field does as well.

Note that the initial electron perturbation $n_e(x, y, t = 0)$ can be chosen arbitrarily, in the end electron density and electric field profiles vary harmonically at pulsation ω_p:

$$n_e(x, y, t) = n_e(x, y, t = 0)\ \cos\left(\omega_p t\right) \tag{31}$$

$$\mathbf{E}(x, y, t) = \mathbf{E}(x, y, t = 0)\ \cos\left(\omega_p t\right) \tag{32}$$

This plasma oscillation can be simulated by discretizing the 2D plasma Euler equations in a cell-vertex formulation with a classical Law-Wendroff scheme (second order in time and space) [31, Chap. 4].

Taking a typical value for the background density [32], $n_0 = 10^{16}\,\mathrm{m}^3$ is used. This value gives an oscillation period of $T_p = 1.11$ ns. A perturbation amplitude around $n_e = 10^{11}\,\mathrm{m}^3$ is used. This value is not critical and only needs to satisfy $n_e \ll n_0$. In the cases presented the electron density field is initially perturbed with a two-Gaussians shape function.

6.2 Neural network Poisson equation solver in plasma oscillation simulation

The use of the neural network to solve the Poisson equation coupled with the unsteady Euler equations to simulate the 2D electron plasma oscillation is carried out using the different architectures and losses presented in the previous section. As for the test cases of the previous section, optimum results are obtained with UNet5, RF = 200 when coupled with transport equations. The choice of `LaplacianLoss` over `InsideLoss` is critical to get a stable simulation as already shown in Fig. 10. A high enough receptive field is also necessary to get an accurate solution.

Examples of plasma oscillation simulations where the Poisson equation has been solved by networks with different receptive fields are shown in Fig. 18. Two UNet5 networks with $n_b = 5$ and RF = 100, 200 are used. As shown in Fig. 12, a gain of accuracy with increased RF is observed with a factor of around 4 for both 1 and infinity norms of the electric field from RF = 100 to RF = 200. This accuracy gain has a real impact on the solution as seen in Fig. 18a where the contours of electron density are not smooth anymore. On the other hand, the RF = 200 network produces very satisfactory results in Fig. 18b. Finally the quantity of interest of the simulation which is the plasma oscillation period T_p, is not well captured in Fig. 18a whereas it is perfectly retrieved in Fig. 18b.

7 Double headed streamer

The previous test case of plasma oscillation, although representative of the interaction between electromagnetic field and plasma species, does not include any chemistry or numerical stiffness as the perturbation electron density at the origin of the plasma oscillation is five orders of magnitude smaller than the background density. A more complex and stiffer case is proposed here to validate the whole methodology developed in the previous sections: the double headed streamer introduced in [32]. Streamer discharges are relevant in plasma assisted combustion [33] and material processing [34].

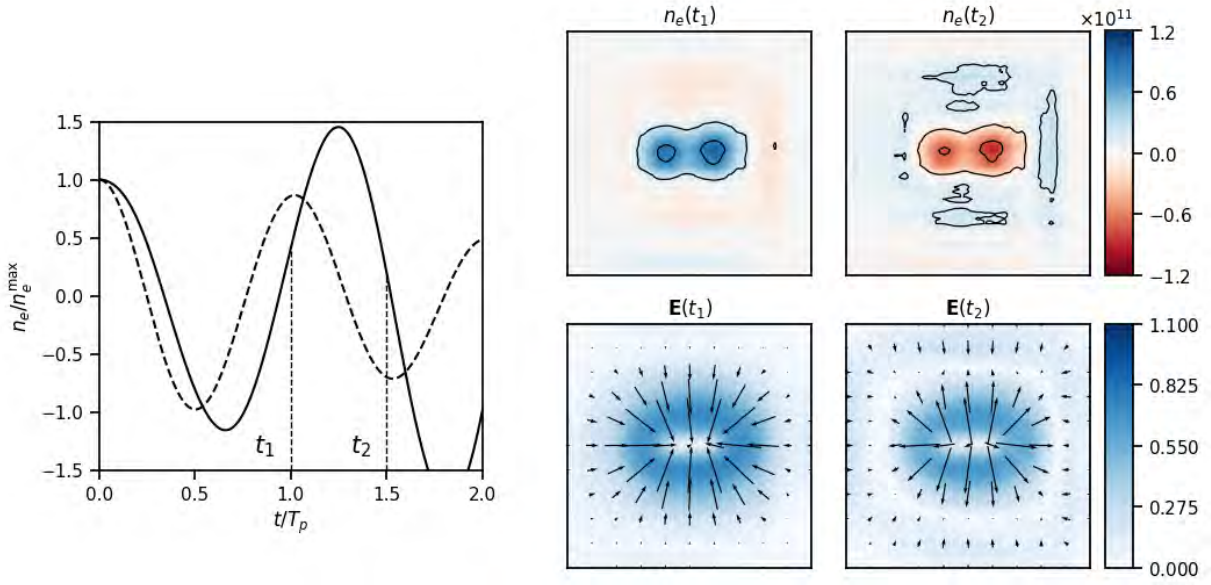
Transport and chemistry kinetic coefficients are detailed in [35]. This chemistry was used to model atmospheric plasma discharges in air in Celestin [32] and Tholin [36] among others. It is composed of three species: electrons (n_e), positive ions (n_p) and negative ions (n_n). Those three species are modelled in a drift-diffusion approximation so that only densities need to be monitored, which is a reasonably well approximation in plasma discharges [32]. The electrons are much faster than the ions due to the mass ratios, so that during the time of the discharge propagation, the ions (n_p and n_n) can be considered not moving: no transport for them is required, and they are therefore only affected by chemistry, which depends on the magnitude of the electric field $E = |\mathbf{E}|$. Thus, the system of equations reads:

$$\frac{\partial n_e}{\partial t} + \nabla \cdot (n_e \mathbf{W_e} - D_e \nabla n_e) = n_e \alpha |W_e| - n_e \eta |W_e| - n_e n_p \beta \tag{33}$$

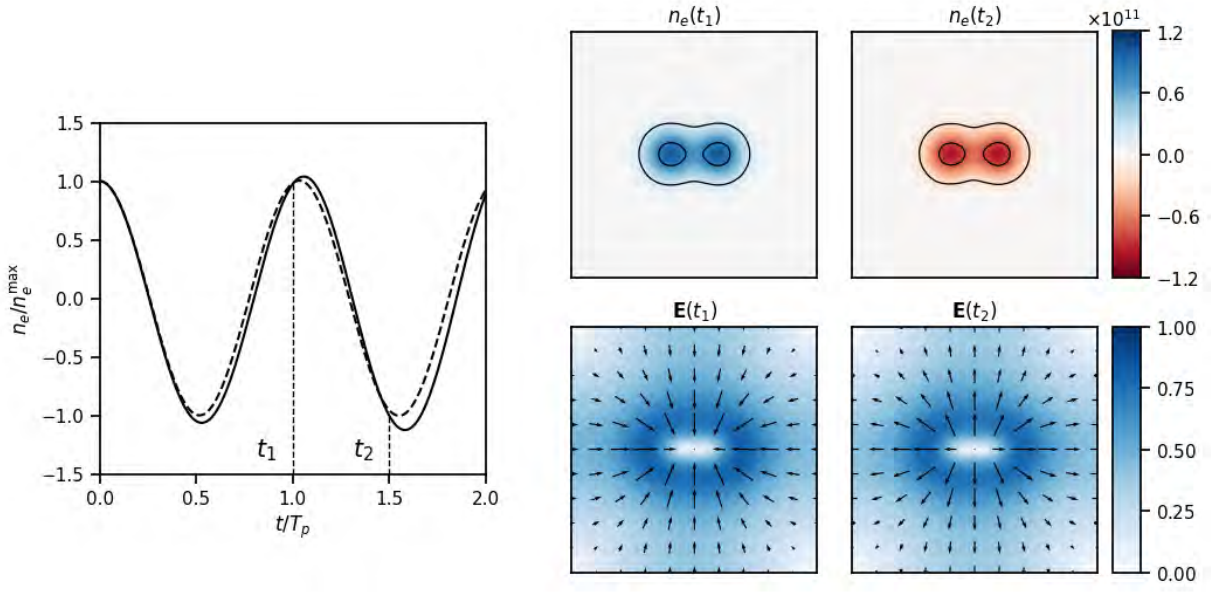
$$\frac{\partial n_p}{\partial t} = n_e \alpha |W_e| - n_e n_p \beta - n_n n_p \beta \tag{34}$$

$$\frac{\partial n_n}{\partial t} = n_e \eta |W_e| - n_n n_p \beta \tag{35}$$

where $\alpha = \alpha(E/N)$ is the ionization coefficient, $\eta = \eta(E/N)$ the attachment coefficient, N the neutral gas density, β the recombination rate, $\mathbf{W}_e = -\mu_e \mathbf{E}$ the drift-velocity of the electrons and $\mu_e = \mu_e(E/N)$ the electron mobility. The electric field $\mathbf{E}$ is critical as it controls both transport for electrons and chemistry for all species. Analogously to the plasma Euler equations, the electric field is computed from the potential given by the Poisson equation:



(a) RF = 100



(b) RF = 200

Figure 18: Plasma oscillation simulation with the Poisson equation solved with UNet5 at different receptive fields. Temporal evolution of the mean (solid) and high absolute values (dashed - points inside the contours of electron density in Fig. 1) on the left and snapshots of n_e and $\mathbf{E}$ at $t_1 = T_p$ and $t_2 = 1.5\,T_p$ on the right.

$$\nabla^2\phi = -\frac{\rho}{\epsilon} \quad \text{where} \quad \rho = e(n_p - n_e - n_n) \tag{36}$$

$$\mathbf{E} = -\nabla\phi \tag{37}$$

The double headed streamer is initialized with a neutral Gaussian profile at $x = 2$ mm and $r = 0$ mm with a background density in a rectangular domain of $L_x \times L_r = 4 \times 1$ mm^2, corresponding to an azimuthal cut of the cylindrical geometry, so that

$$n_e = n_p = n_0 \exp\left[-\left(\frac{x - x_0}{\sigma_x}\right)^2 - \left(\frac{r}{\sigma_r}\right)^2\right] + n_{\text{back}} \tag{38}$$

with $n_0 = 10^{19}\,\text{m}^{-3}$, $n_{\text{back}} = 10^{14}\,\text{m}^{-3}$ and a strong constant electric field of $E_x = 4.8 \times 10^6\,\text{V m}^{-1}$ is applied at the boundary conditions.

A robust upwind scheme has been adopted for the advection part of the electron density with central differencing for the diffusion flux and Euler time integration is performed with a timestep of $\Delta t = 10^{-12}$ s.

The transport equations and the Poisson equation are solved in cylindrical coordinates. An axisymmetric formulation is used so that the 2D domain corresponds to a uniform grid of coordinates (x, r). Because of the cylindrical coordinates, solving the Poisson problem is different compared with the previous 2D cartesian problem (Sections 5 and 6):

$$\begin{cases} \nabla^2\phi = \dfrac{1}{r}\dfrac{\partial}{\partial r}\left(\dfrac{1}{r}\dfrac{\partial\phi}{\partial r}\right) + \dfrac{\partial^2\phi}{\partial x^2} = -R \quad \text{in} \quad \mathring{\Omega} & (39) \\ \phi = -E_x x \quad \text{on} \quad \partial\Omega_D & (40) \\ \nabla\phi \cdot \mathbf{n} = 0 \quad \text{on} \quad \partial\Omega_N & (41) \end{cases}$$

where Dirichlet boundary conditions are applied at $x = 0$, $x = L_x$, $r = L_r$ and Neumann boundary conditions are applied at the axis $r = 0$. A loss function `NeumannLoss` has been introduced to take into account this new boundary condition:

$$\mathcal{L}_N(\phi_{\text{out}}) = \frac{1}{b_s(n_x - 2)} \sum_{b,i} (\nabla\phi_{\text{out}}^{b,0,i} \cdot \mathbf{e}_r)^2 \tag{42}$$

so that three losses are used in this case: `NeumannLoss`, `DirichletLoss` and `LaplacianLoss`. Note that a constant background electric field E_x is applied at the boundary conditions. To stay close to the previous study the problem has been split in two: the neural network deals with zero Dirichlet boundary conditions and charge density. The rest of the problem only yields a constant electric field $E_x\mathbf{e}_x$. From the superposition principle the total electric field is the sum of the neural network solution and the constant electric field:

$$\mathbf{E} = \mathbf{E}_{\text{NN}} + E_x\mathbf{e}_x \tag{43}$$

Training with random profiles as described previously has been done, where a sample of the dataset is shown in Fig. 19. Unlike the cartesian geometry case, the potential is not uniformly distributed but it is amplified at the axis $r = 0$ corresponding to the bottom of the 2D domain in Fig. 19.

The network architecture has been chosen accordingly to the prescribed optimal parameters of Eqs.(26) and (27). These have been adapted to the present case as the geometry is now rectangular and not squared so that a receptive field in each direction can be defined. To achieve these guidelines on the 401×101 mesh, receptive fields of $\text{RF}_x = 800$ and $\text{RF}_y = 200$ have been chosen with $n_b = 5$ branches and around 100 000 parameters.

The strong background electric field imposed in the whole domain allows ionization of air through collisions and the propagation of two streamers, one going to the left (negative streamer) and the other to the right (positive streamer). The UNet5-RF_x800-RF_y200 and linear system Poisson solver results are compared in Figs. 20 and 21. At the beginning of the propagation, the neural network and the linear system yield similar fields (Fig. 20). After a while the absolute values of maximum of electric field and electron density are underestimated by the neural network, where the electric field and electron density profiles are slightly diffused by the network (Fig. 21). Overall, a good agreement is found to

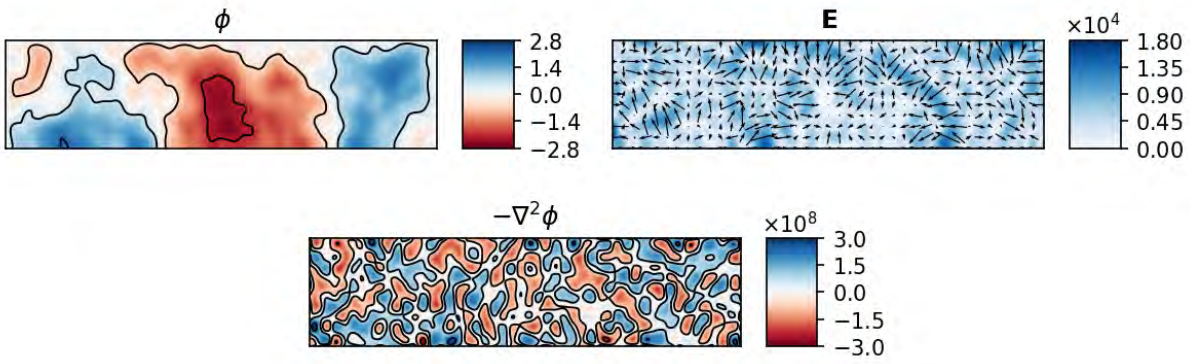
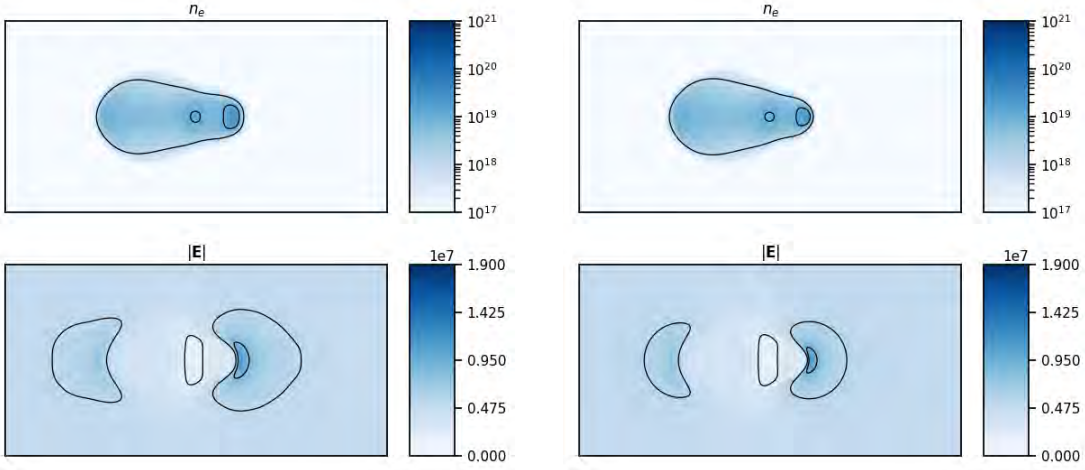


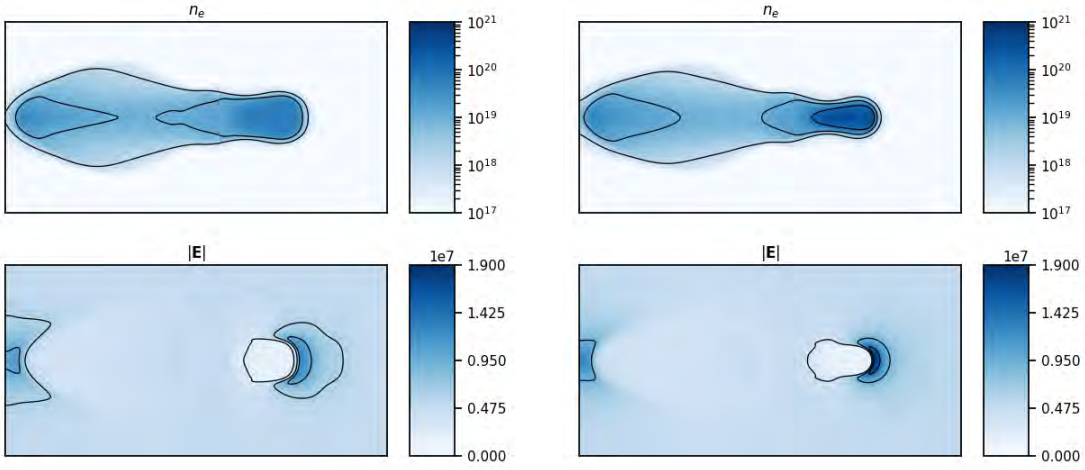
Figure 19: Example of `random_8` source term input in a 4×1 mm^2 cylindrical domain.



(a) Neural network

(b) Linear system

Figure 20: Comparison of electron density and electric field norm at 1.6 ns for neural network and linear system Poisson solver. The computational domain has been mirrored from the central axis.



(a) Neural network

(b) Linear system

Figure 21: Comparison of electron density and electric field norm at 2.8 ns for neural network and linear system Poisson solver. The computational domain has been mirrored from the central axis.

be satisfactory as the UNet5 manages to predict correctly the electric field $\mathbf{E}$, which then drives the propagation of the two streamers.

Two important global properties of these streamers are of interest and need to be well reproduced by the simulations: the speed of the negative and positive streamers as well as the discharge energy. The position of the negative and positive streamers at the axis $r = 0$ can be evaluated by the location x of the maximum of the norm of the electric field. The discharge energy E_d is given by [32]

$$E_d(t) = \int_0^t \int_V \mathbf{J} \cdot \mathbf{E} \, \mathrm{d}V \, \mathrm{d}t \tag{44}$$

where $\mathbf{J}$ is the total current and the space integration is performed on the entire simulation domain. Since only electrons are moving in this case $\mathbf{J} = en_e\mu_e\mathbf{E}$.

These quantities are compared in Fig. 22. Although the network does not reproduce exactly the magnitudes of both the electric field and electron density, it can be observed that those global properties are well reproduced by the neural network Poisson solver.

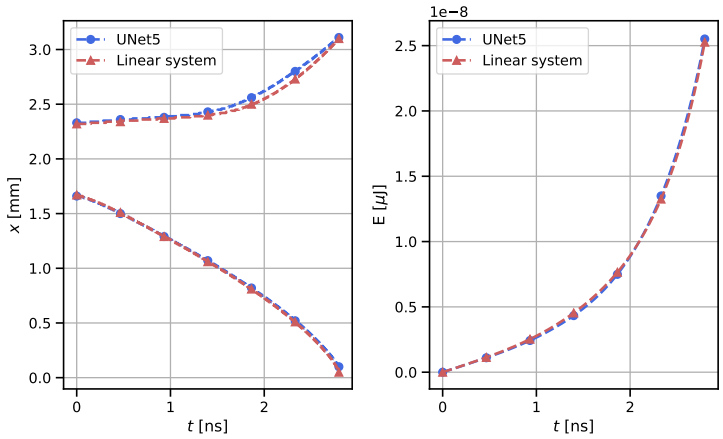


Figure 22: Comparison of positive (above 2 mm) and negative (below 2 mm) streamers propagation and energy for linear system and network runs.

8 Conclusion

CNNs have been used to solve the Poisson equation in a 2D-plasma cartesian geometry simulation. Two types of well-known multiple-scale architectures have been used to predict the potential field from the charge density field: UNet and MSNet. These architectures have been chosen in accordance with the analytical solution of the problem at hand which exposes multiple scales.

Across all the hyperparameters varied, the UNet architecture outperforms the MSNet architecture. This has been attributed to the way MSNets encode information from one scale to the other.

The choice of the loss is critical to get a stable simulation. Although the potential is the solved quantity, the real physical value of interest with regard to simulations is the electric field. From the transport equations, C^2 regularity must be ensured and so the `LaplacianLoss` has been chosen with `DirichletLoss` to have a reference of potential.

Due to the elliptic nature of the Poisson equation, information propagation across the convolutional layers is critical to get an accurate solution. Increasing the receptive field of the network yields better performance because the fundamental mode is better captured that way. For a fixed receptive field, networks with high number of branches should be preferred as they are faster. However, above a certain number of branches, convolutional layers applied on images with not enough pixels are not understood by the network so that there is an optimum number of branches per resolution.

Generalization of the information from one resolution and one domain length is extended to different ones thanks to scaling laws. Loss of accuracy when applying the network to higher resolutions than the trained one is observed and is amplified for low frequencies.

The performance of neural networks solvers has been assessed and is comparable to that of linear system solvers on the hardware configurations studied. It opens the path to further studies to improve AI-based simulations in an HPC context, for instance to reduce the memory load of deep networks.

The coupling of the neural network in place of a linear solver with plasma transport equations has been tested. The best network found from the inference study is also the most stable one inside the simulation.

Lastly the optimal parameters found in this simple cartesian geometry in terms of receptive field and losses have been adapted to a rectangular domain representing an azimuthal cut of the cylindrical geometry. The methodology adopted for the cartesian, full Dirichlet Poisson problem is shown to be valid also for cylindrical, mixed Dirichlet-Neumann boundary conditions showing a good generalization of the method.

Future works could be dedicated to solve the Poisson equation on unstructured meshes, either for plasma-fluid, or incompressible simulations, for instance to take into account the presence of obstacles (*e.g.* the anode and cathode). Additionnally, this work can be extended to other elliptic equations, such as the screened Poisson equation, which governs photoionisation in plasma-fluid simulations. For all these cases, the insight gained by the present study on the architecture, receptive fields, and performances of neural networks in the context of unsteady simulations could be transferable to these future challenges.

Acknowledgments

This work was supported by the ANR project GECCO (ANR-17-CE06-0019).

A Derivation of the analytical solution

The analytical solution of the Poisson equation with boundary conditions depends on the Green function G of the chosen configuration [37, Chap. 1.10]:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0}\int \rho(\mathbf{x}')G(\mathbf{x},\mathbf{x}')\,\mathrm{d}V' + \frac{1}{4\pi}\int\left(G\frac{\partial\phi}{\partial n'} - \phi\frac{\partial G}{\partial n'}\right)\mathrm{d}S' \tag{45}$$

The Green function of a square domain L^2 with zero potential at the four boundaries is [37, Chap. 3.12]:

$$G(x,y,x',y') = \frac{16}{\pi L_x L_y}\sum_{n=1}^{+\infty}\sum_{m=1}^{+\infty}\frac{\sin\left(\frac{n\pi x}{L_x}\right)\sin\left(\frac{n\pi x'}{L_x}\right)\sin\left(\frac{m\pi y}{L_y}\right)\sin\left(\frac{m\pi y'}{L_y}\right)}{n^2/L_x^2 + m^2/L_y^2}. \tag{46}$$

In our case $\phi = 0$ on the boundaries and $G = 0$ on the boundaries has to be satisfied. Hence:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0}\int \rho(\mathbf{x}')G(\mathbf{x},\mathbf{x}')\,\mathrm{d}V'\,. \tag{47}$$

Substituting the Green function, the following solution is obtained:

$$\begin{aligned}\phi(x,y) = \sum_{n=1}^{+\infty}\sum_{m=1}^{+\infty}\Bigg[\frac{4}{L_x L_y}\int_{x',y'}\sin\left(\frac{n\pi x'}{L_x}\right)\sin\left(\frac{m\pi y'}{L_y}\right)R(x',y')\,\mathrm{d}x'\,\mathrm{d}y'\Bigg] \\ \times\frac{\sin\left(\frac{n\pi x}{L_x}\right)\sin\left(\frac{m\pi y}{L_y}\right)}{\pi^2(n^2/L_x^2 + m^2/L_y^2)}.\end{aligned} \tag{48}$$

B Normalization of inputs

A reasonable value for the value of the ratio of the potential over the charge density needs to be found. From the solution of the potential in terms of Fourier series Eq. (48), assuming a constant potential and taking only the term in $n = 1, m = 1$ in the summation reduces to:

$$\phi(x,y) = \frac{4}{L_x L_y} \int_{x',y'} \sin\left(\frac{n\pi x'}{L_x}\right) \sin\left(\frac{m\pi y'}{L_y}\right) R(x',y')\,\mathrm{d}x'\,\mathrm{d}y' \times \frac{\sin\left(\frac{\pi x}{L_x}\right)\sin\left(\frac{\pi y}{L_y}\right)}{\pi^2/L_x^2 + \pi^2/L_y^2}. \tag{49}$$

Taking the absolute value, the following inequality holds:

$$\left|\frac{\phi}{R}\right|_{\max} \leq \frac{1}{\left(\frac{\pi^2}{4}\right)^2 \left(\frac{1}{L_x^2} + \frac{1}{L_y^2}\right)}. \tag{50}$$

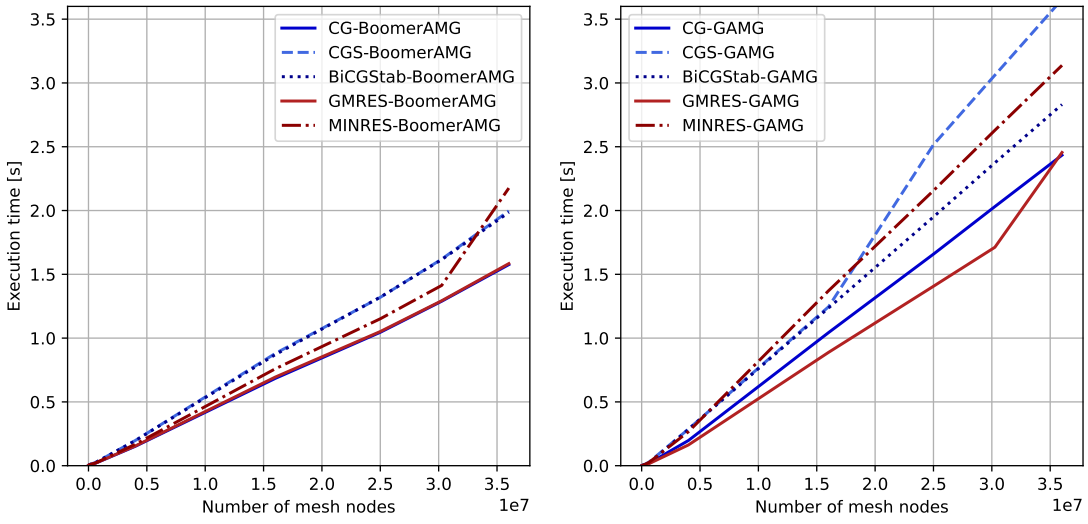
Therefore:

$$\left|\frac{\phi}{R}\right|_{\max} = \frac{\alpha}{\left(\frac{\pi^2}{4}\right)^2 \left(\frac{1}{L_x^2} + \frac{1}{L_y^2}\right)} \tag{51}$$

with $\alpha \leq 1$.

C Linear system solvers benchmark using PETSc

A varierty of solvers have been benchmarked in PETSc and are shown in Fig. 23. The most popular and used iterative solvers have been tested: Conjugate Gradient (CG), Conjugate Gradient Squared (CGS), Stabilized Biconjugate Gradient (BiCGStab), Minimal Residual (MINRES) and Generalized Minimal Residual (GMRES). The preconditioner is critical to get good performance and since the problem is elliptic, multigrid preconditioners are best suited for them [38]. The native PETSc GAMG [28] and Hypre BoomerAMG [30] resolution times in combination with the aforementioned iterative solvers are shown in Figs. 23a and 23b, respectively. Other preconditioners have been tested but due to huge performance gap compared to the multi-grid preconditioners they are not shown. BoomerAMG is shown to outperform the native PETSc GAMG for every iterative solver used. When using BoomerAMG, GMRES and CG yield very similar results with a small edge for CG which has been retained in Fig. 17.



(a) BoomerAMG

(b) PETSc GAMG

Figure 23: Performance of different solvers using AMG preconditioning on `config_1`.

References

[1] Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. *Numerical Mathematics*, volume 37. 01 2007.

[2] Jonathan Tompson, Kristofer Schlachter, Pablo Sprechmann, and Ken Perlin. Accelerating eulerian fluid simulation with convolutional networks. In *International Conference on Machine Learning*, pages 3424–3433. PMLR, 2017.

[3] Ekhi Ajuria Illarramendi, Antonio Alguacil, Michaël Bauerheim, Antony Misdariis, Benedicte Cuenot, and Emmanuel Benazera. Towards an hybrid computational strategy based on deep learning for incompressible flows. In *AIAA AVIATION 2020 FORUM*, page 3058, 2020.

[4] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[5] MWMG Dissanayake and Nhan Phan-Thien. Neural-network-based approximations for solving partial differential equations. *communications in Numerical Methods in Engineering*, 10(3):195–201, 1994.

[6] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. *arXiv preprint arXiv:1711.10561*, 2017.

[7] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of machine learning research*, 18, 2018.

[8] Nils Wandel, Michael Weinmann, and Reinhard Klein. Unsupervised deep learning of incompressible fluid dynamics. *arXiv preprint arXiv:2006.08762*, 2020.

[9] Antonio Alguacil, Michael Bauerheim, Marc C Jacob, and Stéphane Moreau. Predicting the propagation of acoustic waves using deep convolutional neural networks. *Journal of Sound and Vibration*, page 116285, 2021.

[10] Steven L Brunton, Bernd R Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52:477–508, 2020.

[11] Cheng Yang, Xubo Yang, and Xiangyun Xiao. Data-driven projection method in fluid simulation. *Computer Animation and Virtual Worlds*, 27(3-4):415–424, 2016.

[12] Ali Girayhan Özbay, Arash Hamzehloo, Sylvain Laizet, Panagiotis Tzirakis, Georgios Rizos, and Björn Schuller. Poisson cnn: Convolutional neural networks for the solution of the poisson equation on a cartesian mesh. *Data-Centric Engineering*, 2, 2021.

[13] Tao Shan, Wei Tang, Xunwang Dang, Maokun Li, Fan Yang, Shenheng Xu, and Ji Wu. Study on a fast solver for poisson's equation based on deep learning technique. *IEEE Transactions on Antennas and Propagation*, 68(9):6725–6733, 2020.

[14] Kiwon Um, Philipp Holl, Robert Brand, Nils Thuerey, et al. Solver-in-the-loop: Learning from differentiable physics to interact with iterative pde-solvers. *arXiv preprint arXiv:2007.00016*, 2020.

[15] J.A. Bittencourt. *Fundamentals of Plasma Physics*. Springer, third edition, 2004.

[16] Michael Mathieu, Camille Couprie, and Yann Lecun. Deep multi-scale video prediction beyond mean square error. *Proceedings of the ICLR*, 2016.

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[18] Kai Fukami, Koji Fukagata, and Kunihiko Taira. Super-resolution reconstruction of turbulent flows with machine learning. *Journal of Fluid Mechanics*, 870:106–120, 2019.

[19] Antonio Alguacil, Michaël Bauerheim, Marc C Jacob, and Stéphane Moreau. Predicting the propagation of acoustic waves using deep convolutional neural networks. In *AIAA AVIATION 2020 FORUM*, page 2513, 2020.

[20] Corentin J Lapeyre, Antony Misdariis, Nicolas Cazard, Denis Veynante, and Thierry Poinsot. Training convolutional neural networks to estimate turbulent sub-grid scale reaction rates. *Combustion and Flame*, 203:255–264, 2019.

[21] Nils Thuerey, Konstantin Weißenow, Lukas Prantl, and Xiangyu Hu. Deep learning methods for reynolds-averaged navier–stokes simulations of airfoil flows. *AIAA Journal*, 58(1):25–36, 2020.

[22] Leon O Chua and Tamas Roska. The cnn paradigm. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 40(3):147–156, 1993.

[23] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 4905–4913, 2016.

[24] Christopher M Bishop et al. *Neural networks for pattern recognition*. Oxford university press, 1995.

[25] Francois Chollet. *Deep learning with Python*. Simon and Schuster, 2017.

[26] Ekhi Ajuria Illarramendi, Michaël Bauerheim, and Bénédicte Cuenot. Performance and accuracy assessments of an incompressible fluid solver coupled with a deep convolutional neural network, 2021.

[27] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *International Conference on Learning Representations (ICLR)*, 2021.

[28] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Steven Benson, Jed Brown, Peter Brune, Kris Buschelman, Emil M. Constantinescu, Lisandro Dalcin, Alp Dener, Victor Eijkhout, William D. Gropp, Václav Hapla, Tobin Isaac, Pierre Jolivet, Dmitry Karpeev, Dinesh Kaushik, Matthew G. Knepley, Fande Kong, Scott Kruger, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Lawrence Mitchell, Todd Munson, Jose E. Roman, Karl Rupp, Patrick Sanan, Jason Sarich, Barry F. Smith, Stefano Zampini, Hong Zhang, Hong Zhang, and Junchao Zhang. PETSc Web page. `https://petsc.org/`, 2021.

[29] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. 01 2003.

[30] Robert D. Falgout and Ulrike Meier Yang. hypre: A library of high performance preconditioners. In Peter M. A. Sloot, Alfons G. Hoekstra, C. J. Kenneth Tan, and Jack J. Dongarra, editors, *Computational Science — ICCS 2002*, pages 632–641, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[31] N. Lamarque. *Numerical schemes and boundary conditions for the LES of two-phase flows in helicopter chambers*. PhD thesis, CERFACS, 2007.

[32] Sébastien Célestin. *Study of the dynamics of streamers in air at atmospheric pressure*. PhD thesis, 2008. Thèse de doctorat dirigée par Bourdon, Anne et Rousseau, Antoine Physique Châtenay-Malabry, Ecole centrale de Paris 2008.

[33] Yiguang Ju, Joseph K. Lefkowitz, Christopher B. Reuter, Sang Hee Won, Xueliang Yang, Suo Yang, Wenting Sun, Zonglin Jiang, and Qi Chen. Plasma Assisted Low Temperature Combustion. *Plasma Chemistry and Plasma Processing*, 36(1):85–105, 2016.

[34] Michael A. Lieberman and Allan J. Lichtenberg. *Principles of Plasma Discharges and Material Processings*. John Wiley and Sons, second edition, 2005.

[35] R Morrow and J J Lowke. Streamer propagation in air. *Journal of Physics D: Applied Physics*, 30(4):614–627, February 1997.

[36] Fabien Tholin. *Numerical simulation of nanosecond repetitively pulsed discharges in air at atmospheric pressure : Application to plasma-assisted combustion*. PhD thesis, 2012. Thèse de doctorat dirigée par Bourdon, Anne Physique Châtenay-Malabry, Ecole centrale de Paris 2012.

[37] John David Jackson. *Classical Electrodynamics*. John Wiley & Sons, 1999.

[38] Jinchao Xu and Ludmil T Zikatanov. Algebraic multigrid methods. 2016.

PlasmaNet: a framework to study and solve elliptic differential equations using neural networks in plasma fluid simulations

Lionel Cheng
CERFACS
cheng@cerfacs.fr

Ekhi Ajuria Illarramendi
ISAE-SUPAERO /CERFACS
ekhi.ajuria@cerfacs.fr

Michaël Bauerheim
ISAE-SUPAERO
bauerheim@isae-supaero.fr

Bénédicte Cuenot
CERFACS
cuenot@cerfacs.fr

1 Introduction

Elliptic partial differential equations (PDEs) are common in many areas of physics, from the Poisson equation in plasmas and incompressible flows to the Helmholtz equation in electromagnetism. Their numerical solution requires to solve a linear system and many libraries have been developed for this task [1]. Solving a linear system efficiently requires preconditioning the system which is a difficult task. It can become a bottleneck for performance when the number of nodes increases.

The rise of computational power and inherent speed of GPUs offers exciting opportunities to solve PDEs by recasting them in terms of optimization problems. Since the major introduction of *Physics informed neural networks* (PINN) [2], other architectures [3] and frameworks [4] have been created.

In plasma fluid simulations, the Poisson equation is solved, coupled to the charged species transport equations [5]. A pioneer work [6] has shown significant speedup using neural networks to solve the Poisson equation compared to classical linear system solvers on this problem. Coupling the neural network Poisson solver to plasma transport equations has shown promising results and the neural network can be considered as a viable option in terms of accuracy.

This work extends [6] and introduces *PlasmaNet* (`https://gitlab.com/cerfacs/plasmanet`), an open-source library written to study neural networks in plasma simulations. The optimal network parameters are first recalled and metrics to help design appropriate network architectures for solving elliptic differential equations are presented. We then attempt to solve a new class of elliptic differential equations, the screened Poisson equations using neural networks. These equations are used to infer the photoionization source term from the ionization rate in streamer discharges [7]. Finally a simulation running with three neural networks, coupled to plasma transport equations, to solve both the Poisson and the photoionization equations, is performed to assess the accuracy of neural networks predictions.

2 Network architectures for elliptic differential equations

Laplace and Poisson equations form the basis of elliptic PDEs. Studying them can give insights on how to solve all elliptic PDEs. From a given charge density ρ_q the Poisson equation yields the electromagnetic potential ϕ from which we can compute the electric field $\mathbf{E} = -\nabla\phi$ so that in the

Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021).

end $\mathbf{E} = f(\rho_q)$. In numerical simulations the physical domain is finite so that the Poisson equation is supplemented by Dirichlet and Neumann boundary conditions:

$$\nabla^2\phi = -\frac{\rho_q}{\varepsilon_0} \quad \text{in} \quad \mathring{\Omega} \quad \text{with} \quad \begin{cases} \phi = 0 & \text{on} \quad \partial\Omega_D \\ \mathbf{E}\cdot\mathbf{n} = E_N & \text{on} \quad \partial\Omega_N \end{cases} \tag{1}$$

The analytical solution of the Poisson equation with boundary conditions depends on the Green function G of the chosen configuration [8, Chap. 1.10]:

$$\phi(\mathbf{x}) = \frac{1}{4\pi\epsilon_0}\int \rho(\mathbf{x}')G(\mathbf{x},\mathbf{x}')\,\mathrm{d}V' + \frac{1}{4\pi}\int\left(G\frac{\partial\phi}{\partial n'} - \phi\frac{\partial G}{\partial n'}\right)\mathrm{d}S' \tag{2}$$

Expressions of the Green functions [8, Chap. 3] in series for cartesian, cylindrical and spherical coordinates highlight the importance of multiple scales in the Poisson equation which must be incorporated in the neural network architecture. The UNet architecture [9] has proved to work best to infer $\mathbf{E} = \mathrm{NN}(\rho_q)$ in [6] compared to the Multi-Scale architecture [10]. In UNets, the local branch b is defined as the power of 2 by which the initial resolution is divided by and the number of branches n_b is the number of scales involved. Local d_b and global D depths are defined as the number of successive convolutional layers in branch b and across the whole network, respectively.

Concerning losses, the use of a physical loss in the form of a `LaplacianLoss` inspired from PINNs [2] is critical to yield stable trainings and stable simulations when coupled to transport equations [6].

Elliptic PDEs, having no real characteristic curves, need the information of the whole domain at every point. This is highlighted by the analytical solution Eq. (2) which incorporates a domain integral. To quantify the information propagation across the neural network the global receptive field RF is defined as the size of the domain of influence of the input center point in number of points in the original scale n_p. The receptive field can be splitted into local receptive fields per branch RF_b so that

$$\mathrm{RF} = \sum_{b=0}^{n_b-1}\mathrm{RF}_b \quad \text{with} \quad \mathrm{RF}_b = \begin{cases} 1 + d_b(k_s-1)2^b & \text{if } b = 0 \\ d_b(k_s-1)2^b & \text{otherwise} \end{cases} \tag{3}$$

where k_s is the kernel size assumed constant in the network.

In [11], a *theoretical receptive field* is defined as the size of the input domain of influence on the output center point. Tests carried in *PlasmaNet* on the studied UNets show that this definition matches Eq. (3) so that both formulations are equivalent. However, the importance of the domain influence is not uniform, as points closer to the studied pixel will have more paths to influence the output, resulting in a *gaussian-like* distribution [12]. A parametric study in [6, Sec. 5] across multiple UNets showed that the optimal global parameters of the network for a given number of pixels n_p should be:

$$\mathrm{RF} = 2n_p \quad \text{and} \quad n_b = \max\{b \in \mathbb{N} | \lfloor n_p/2^b \rfloor > k_s\} + 1 \tag{4}$$

With these parameters the receptive field fills the entire computational domain for all the input points and convolution is relevant in the downscaled branches [6, Sec. 5].

3 Photoionization in plasma discharges

We model plasma discharges in air using the chemistry from [13]. It consists of electrons (n_e), positive ions (n_p) and negative ions (n_n). Those three species are modeled in a drift-diffusion approximation where the ions are considered not moving [7]:

$$\frac{\partial n_e}{\partial t} + \nabla\cdot(n_e\mathbf{W_e} - D_e\nabla n_e) = n_e\alpha|W_e| - n_e\eta|W_e| - n_e n_p\beta + S_{ph} \tag{5}$$

$$\frac{\partial n_p}{\partial t} = n_e\alpha|W_e| - n_e n_p\beta - n_n n_p\beta + S_{ph} \quad \frac{\partial n_n}{\partial t} = n_e\eta|W_e| - n_n n_p\beta \tag{6}$$

where S_{ph} is the photoionization source term, $\alpha = \alpha(E/N)$ is the ionization coefficient, $\eta = \eta(E/N)$ the attachment coefficient, E the electric field magnitude, N the neutral gas density, β the recombination rate, $\mathbf{W}_e = -\mu_e \mathbf{E}$ the drift-velocity of the electrons and $\mu_e = \mu_e(E/N)$ the electron mobility. These transport equations are coupled to the Poisson equation as $\rho_q = e(n_p - n_e - n_n)$. Such a coupling with the Poisson neural network solver has already been studied in [6]. However a rather high background density of charged species was used so that the photoionization source term S_{ph} could be neglected in previous work. Here, similar cases where photoionization cannot be neglected are considered, highlighting the benefits of the proposed open-source framework *PlasmaNet* to incorporate new physics for CNN-based fluid plasma simulations.

The plasma discharge is a double headed streamer [7], initialized with a neutral Gaussian profile at $x = 2$ mm and $r = 0$ mm with a background density in a rectangular domain of $L_x \times L_r = 4 \times 1$ mm^2, corresponding to an azimuthal cut of the cylindrical geometry.

In plasma air discharges, the electron-impact reactions produce excited states of N_2. Radiative relaxations of these states are absorbed by O_2 causing ionization [14]. Integral models have been developed to model this photoionization source term but are very costly in numerical simulations [14]. Simplifiying assumptions allow to recast this photoionization source term in terms of screened Poisson equations [7]:

$$S_{ph}(\mathbf{r}) = \sum_{j=1}^{j_m} S_{ph}^j(\mathbf{r}) \quad \forall j \in [\![1, j_m]\!] \; \nabla^2 S_{ph}^j(\mathbf{r}) - (\lambda_j p_{\mathrm{O2}})^2 S_{ph}^j(\mathbf{r}) = -A_j p_{\mathrm{O2}}^2 I(\mathbf{r}) \tag{7}$$

where $I(\mathbf{r}) = f(E/p) n_e \alpha |W_e|$ is an effective ionization term, p_{O2} the oxygen pressure, λ_j, A_j are fitting parameters and $j_m = 2$ or 3 depending on the level of precision wanted. Note that j_m resolutions of linear systems must be performed inside a numerical time iteration when solving S_{ph} so that the computational cost of photoionization is high. Here only $j_m = 2$ is considered for simplicity. Each component of the photoionoization source term obeys a screened Poisson equation:

$$\nabla^2 \phi - \lambda^2 \phi = -R \tag{8}$$

where λ controls the amount of diffusion of the solution, the higher the value of λ, the lower the diffusion ($\lambda = 0$ is the maximum diffusion and reduces to the Poisson equation). We apply a neural network to solve each component of the photoionization source term using the optimal parameters in Eq. (4). The same network architecure as the Poisson neural network solver (UNet5-RF_x800-RF_y200) is thus applied for each component of the photoionization source term. We introduce a physical loss associated to the screened Poisson equation called `PhotoLoss`:

$$\mathcal{L}_P(\phi_{\mathrm{out}}; \lambda) = \frac{L_x^2 L_y^2}{b_s (n_x - 1)(n_y - 1)} \sum_{b,j,i} \left[\nabla^2 \phi_{\mathrm{out}}^{b,j,i} - \lambda^2 \phi_{\mathrm{out}}^{b,j,i} + R_{\mathrm{in}}^{b,j,i} \right]^2 \tag{9}$$

Training is done using random datasets of 10 000 snapshots introduced in [15] and already used in [6, Sec. 4]. A snapshot of such a dataset is shown in Fig. 1. It can be seen that S_{ph}^2 is less diffusive than S_{ph}^1 which behaves closer to the Poisson equation due to $\lambda_1 \ll \lambda_2$.

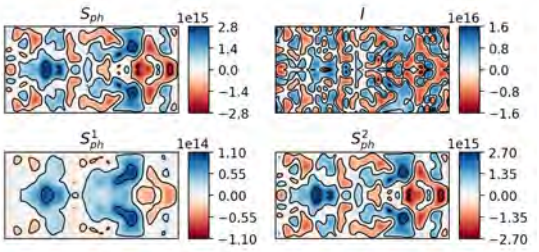


Figure 1: Example of `random_12` source term input in a 4×1 mm^2 cylindrical domain.

4 Results

First, snapshots of the propagation of the double headed streamer without photoionization using a linear system solver and a neural network solver for the Poisson equation are shown in Fig 2. Electron density and electric field profiles tend to be more diffusive when the Poisson equation is solved by the neural network. However the speed of propagation is well-captured as well as the discharge energy [6, Sec. 7].

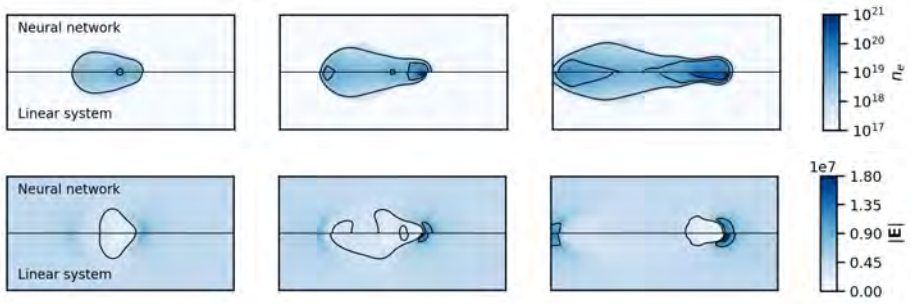


Figure 2: Electron density n_e and norm of electric field $|\mathbf{E}|$ of a double headed streamer using UNet5-RFx800-RFy200 (top-half) and a linear system solver (bottom-half) at 1.2, 2.0 and 2.8 ns.

$$\mathbf{E} = \mathrm{NN}_1(\rho_q) \\ S_{ph} = \mathrm{NN}_2(I) + \mathrm{NN}_3(I) \quad \rightarrow \quad \mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t[\mathbf{R}_t(\mathbf{E}) + \mathbf{R}_c(\mathbf{E}, S_{ph})]$$

Figure 3: Interaction of the neural network with plasma transport equations at each iteration. $\mathbf{U} = (n_e, n_p, n_n)$, $\mathbf{R}_t$ and $\mathbf{R}_c$ are transport and chemistry residuals.

Second, simulations with photoionization have been performed with two networks inferring S_{ph}^1 and S_{ph}^2. Another network is used to infer the electric field $\mathbf{E}$ (as in [6] and results from Fig. 2) so that three networks are coupled to the plasma transport equations to replace linear system solvers. Results are shown in Fig. 4 with a sketch of coupling in Fig. 3. The interaction of multiple neural networks solutions seem to yield promising results as the photoionization source term and electric field are correctly predicted. The right-propagating positive streamer ia slightly more diffuse when looking at the electron density than the reference solution whereas the left-propagating negative streamer is better captured by the neural networks. We note however negative values in the prediction of S_{ph} by the neural network (white regions in the snapshots due to clipping), which strictly speaking should not be allowed as the ionization source term is positive. These values have a relative amplitude of 10^{-3} and have been clipped to get good streamer propagation.

To prevent the rise of these negative values, a positive-valued dataset can be used so that the network learns only to infer positive-values. A similar dataset as the one shown in Fig. 1 but without negative values for the ionization rate I is used to train the networks and results are shown in Fig. 5. The negative values are indeed removed but the overall shape of the electron density is more diffused especially for the right-propagating positive streamer: the network struggles to predict very low values of S_{ph}.

5 Conclusion

We have introduced *PlasmaNet* and shown its ability to couple neural network solvers to plasma transport equations. The range of applicability of the method developed in [6] has been extended to the more general screened Poisson equations, which highlights the flexibility of the present framework *PlasmaNet* to incorporate new complex physics simulated by neural networks. Future work will try to integrate the screening length λ directly inside the network so that one network and not three will be necessary to solve both the Poisson equation and the photoionization source term. Dedicated regularized terms, such as penalty on negative S_{ph} could also be tested. Interaction of the neural network with other plasma test cases such as Hall effect thrusters are also planned.

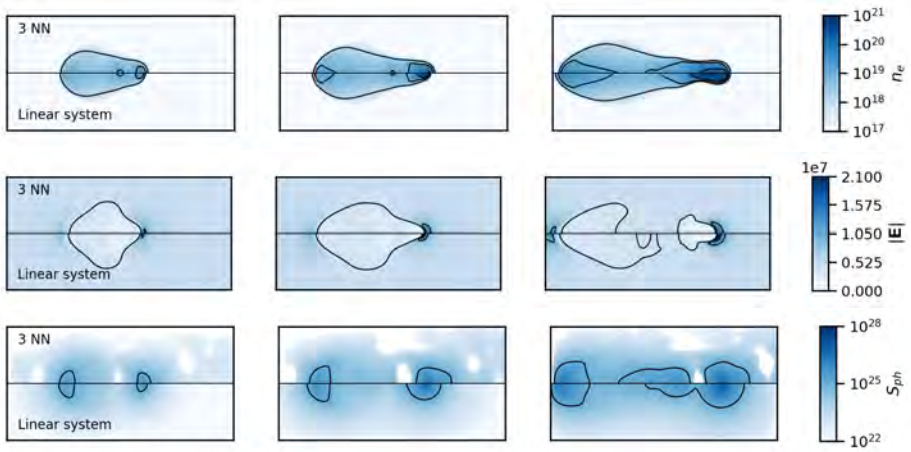


Figure 4: Electron density n_e, norm of electric field $|\mathbf{E}|$ an photoionization source term S_{ph} of a double headed streamer using three neural networks (top-half) and a linear system solver (bottom-half) at 1.6, 2.2 and 2.8 ns.

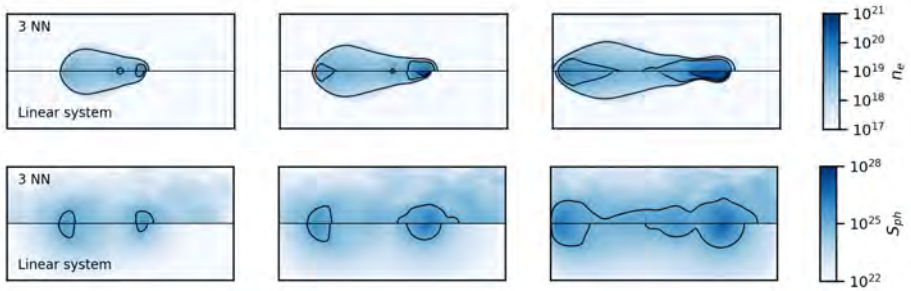


Figure 5: Electron density n_e and photoionization source term S_{ph} of a double headed streamer using three neural networks (top-half) and a linear system solver (bottom-half) at 1.6, 2.2 and 2.8 ns using a positive-valued training dataset.

References

[1] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin, Alp Dener, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Todd Munson, Karl Rupp, Patrick Sanan, Barry F. Smith, Stefano Zampini, Hong Zhang, and Hong Zhang. PETSc Web page, 2021. https://petsc.org/.

[2] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations. *arXiv preprint arXiv:1711.10561*, 2017.

[3] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

[4] Philipp Holl, Vladlen Koltun, Kiwon Um, and Nils Thuerey. phiflow: A differentiable pde solving framework for deep learning via physical simulations. *DiffCVGP workshop at NeurIPS 2020*, 2020.

[5] J.A. Bittencourt. *Fundamentals of Plasma Physics*. Springer, third edition, 2004.

[6] Lionel Cheng, Ekhi Ajuria Illarramendi, Guillaume Bogopolsky, Michael Bauerheim, and Benedicte Cuenot. Using neural networks to solve the 2d poisson equation for electric field computation in plasma fluid simulations. *arXiv preprint arXiv:2109.13076*, 2021.

[7] Sébastien Célestin. *Study of the dynamics of streamers in air at atmospheric pressure*. PhD thesis, 2008. Thèse de doctorat dirigée par Bourdon, Anne et Rousseau, Antoine Physique Châtenay-Malabry, Ecole centrale de Paris 2008.

[8] John David Jackson. *Classical Electrodynamics*. John Wiley & Sons, 1999.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[10] Michael Mathieu, Camille Couprie, and Yann Lecun. Deep multi-scale video prediction beyond mean square error. *Proceedings of the ICLR*, 2016.

[11] Leon O Chua and Tamas Roska. The cnn paradigm. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 40(3):147–156, 1993.

[12] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 4905–4913, 2016.

[13] R Morrow and J J Lowke. Streamer propagation in air. *Journal of Physics D: Applied Physics*, 30(4):614–627, February 1997.

[14] N. Liu. *Dynamics of positive and negative streamers in sprites*. PhD thesis, The Pennsylvania State University, 2006.

[15] Ali Girayhan Özbay, Arash Hamzehloo, Sylvain Laizet, Panagiotis Tzirakis, Georgios Rizos, and Björn Schuller. Poisson cnn: Convolutional neural networks for the solution of the poisson equation on a cartesian mesh. *Data-Centric Engineering*, 2, 2021.

14.2 Conclusion

The capability of neural networks to solve elliptic differential equations has been studied in these two papers, where multiple scale architectures have been used to solve the Poisson and generalized screened Poisson equations. The receptive field of the neural network architectures need to be high enough so that every point can communicate any other point of the domain, linking the architecture of the network with the elliptic nature of the equations. Other equations could be targeted to assess how far this analogy can be taken: hyperbolic equations would need a lower receptive field for example than elliptic equations. One network to solve a whole class of differential equations, such as the generalized Poisson equations, could also be the subject of future work.

Appendices

Appendix A

Differential relations for thermodynamic potentials

Property A.1. *From a function $z = z(x, y) = z(r, \theta)$ the following differential relations hold true:*

1. *The inverse rule*
$$\left(\frac{\partial z}{\partial y}\right)_x = 1/\left(\frac{\partial y}{\partial z}\right)_x \tag{A.1}$$
2. *The cycle rule*
$$\left(\frac{\partial z}{\partial x}\right)_y = -\left(\frac{\partial z}{\partial y}\right)_x \left(\frac{\partial y}{\partial x}\right)_z \tag{A.2}$$
3. *The chain rule*
$$\left(\frac{\partial z}{\partial r}\right)_\theta = \left(\frac{\partial z}{\partial x}\right)_y \left(\frac{\partial x}{\partial r}\right)_\theta + \left(\frac{\partial z}{\partial y}\right)_x \left(\frac{\partial y}{\partial r}\right)_\theta \tag{A.3}$$

Proof. The inverse and cycle rules are demonstrated together. By considering $y = y(x, z)$ the definition of the differentials yields

$$\mathrm{d}z = \left(\frac{\partial z}{\partial x}\right)_y \mathrm{d}x + \left(\frac{\partial z}{\partial y}\right)_x \mathrm{d}y \tag{A.4}$$
$$\mathrm{d}y = \left(\frac{\partial y}{\partial x}\right)_z \mathrm{d}x + \left(\frac{\partial y}{\partial z}\right)_x \mathrm{d}z \tag{A.5}$$

Inserting Eq. (A.5) into Eq. (A.4) the following expression for the z differential holds

$$\mathrm{d}z = \left[\left(\frac{\partial z}{\partial x}\right)_y + \left(\frac{\partial z}{\partial y}\right)_x \left(\frac{\partial y}{\partial x}\right)_z\right] \mathrm{d}x + \left(\frac{\partial z}{\partial y}\right)_x \left(\frac{\partial y}{\partial z}\right)_x \mathrm{d}z \tag{A.6}$$

By equating the two sides of this equation the inverse rule and the cycle rule follow. The chain rule is simply an application of the change of variables property of the differentials. □

Appendix B

Quantum statistical mechanics

B.1 Quantum mechanics

The summary of quantum mechanics presented here is taken from Basdevant and Dalibard [2005].

B.1.1 Wave function

- The state of a particle in space is described by a wave function $\psi(\boldsymbol{r}, t)$ whose square modulus yields the probability density at point $\boldsymbol{r}$ and instant t.
- The time evolution of the wave function of a particle in the influence of a potential $V(\boldsymbol{r})$ is given by the Schrödinger equation:

$$i\hbar\frac{\partial}{\partial t}\psi(\boldsymbol{r}, t) = \hat{H}\psi(\boldsymbol{r}, t), \tag{B.1}$$

  where the energy observable $\hat{H}$ or hamiltonian of the system, is:

$$\hat{H} = -\frac{\hbar^2}{2m}\nabla^2 + V(\boldsymbol{r}). \tag{B.2}$$

- The probability amplitude of the momentum of the particle is given by the Fourier transform of the wave function

$$\varphi(\boldsymbol{p}, t) = \int \mathrm{e}^{-i\boldsymbol{p}\cdot\boldsymbol{r}/\hbar}\psi(\boldsymbol{r}, t)\frac{\mathrm{d}^3\mathbf{r}}{(2\pi\hbar)^{3/2}} \tag{B.3}$$

- As a consequence the following *Heisenberg incertitude relations* are derived:

$$\Delta x \Delta p_x \geq \hbar/2. \tag{B.4}$$

- To each physical value A is associated an *observable* $\hat{A}$, linear hermitian operator acting on the wave functions. The mean value $\langle a\rangle_t$ at the instant t of the measure of the physical value A is:

$$\langle a\rangle_t = \int \psi^*(\boldsymbol{r}, t)\Big[\hat{A}\psi(\boldsymbol{r}, t)\Big]\mathrm{d}^3\mathbf{r}. \tag{B.5}$$

- The position observable $\hat{\boldsymbol{r}}$ corresponds to $\boldsymbol{r}$ multiplication while the momentum observable is:

$$\hat{\boldsymbol{p}} = \frac{\hbar}{\mathrm{i}}\boldsymbol{\nabla} \tag{B.6}$$

These observables do not commute; *e.g.* $[\hat{x}, \hat{p}_x] = \mathrm{i}\hbar$.

- If the wave function is an eigenfunction of the observable $\hat{A}$ corresponding to the eigenvalue a_α, the result of the mesure of A is a_α with probability one.

- For an isolated system placed in a stationary potential, the stationary states are the eigenfunctions of the energy, with a wave function of the form:

$$\psi(\boldsymbol{r}, t) = \psi_\alpha(\boldsymbol{r}) \mathrm{e}^{-\mathrm{i}E_\alpha t/\hbar}, \tag{B.7}$$

where ψ_α is a normalized solution ($\int |\psi_\alpha|^2 = 1$) of the stationary Schrödinger equation:

$$\hat{H}\psi_\alpha(\boldsymbol{r}) = E_\alpha \psi_\alpha(\boldsymbol{r}). \tag{B.8}$$

The time evolution of the wave function $\psi(\boldsymbol{r}, t)$ is then:

$$\psi(\boldsymbol{r}, t) = \sum_\alpha C_\alpha \mathrm{e}^{-\mathrm{i}E_\alpha t/\hbar} \psi_\alpha(\boldsymbol{r}) \quad \text{with} \quad C_\alpha = \int \psi_\alpha^*(\boldsymbol{r}) \psi(\boldsymbol{r}, t=0) \mathrm{d}^3\boldsymbol{r}. \tag{B.9}$$

B.1.2 Hilbert space

A Hilbert space is a complex inner product space. In the following the Hilbert space is supposed to be complete.

First principle of quantum mechanics To every physical system is associated a Hilbert space $\mathcal{E}_H$. The state of the system is defined at each instant by a normed vector $|\psi(t)\rangle$ of $\mathcal{E}_H$.

Second principle of quantum mechanics: physical values measurement

1. To every physical value A corresponds a hermitian operator $\hat{A}$ of $\mathcal{E}_H$: $\hat{A}$ is the observable representing A.

2. Let $|\psi\rangle$ be the state of the particle when a measurement of A is made. The only possible results are eigenvalues of the observable $\hat{A}$.

3. Let $\hat{P}_\alpha$ be the projector on the subspace associated to eigenvalue a_α. The probability of finding the value a_α when measuring A is:

$$\mathcal{P}(a_\alpha) = \|\psi_\alpha\|^2 \quad \text{where} \quad |\psi_\alpha\rangle = \hat{P}_\alpha |\psi\rangle . \tag{B.10}$$

4. Immediately after an a_α measurement of A, the new state of the sytem is:

$$|\psi'\rangle = |\psi_\alpha\rangle / \|\psi_\alpha\|. \tag{B.11}$$

Third principle of quantum mechanics: time evolution Let $|\psi(t)\rangle$ be the state of the system at the instant t. While no observation on the system is made, its time evolution

is given by the Schrödinger equation:

$$\mathrm{i}\hbar\frac{\mathrm{d}}{\mathrm{d}t}\left|\psi(t)\right\rangle = \hat{H}\left|\psi(t)\right\rangle . \tag{B.12}$$

B.1.3 Observable commutation

B.1.3.1 Uncertainty principles

Let us consider two physical values A and B and their respective operators $\hat{A}$ and $\hat{B}$. The state of the system is specified by $|\psi\rangle$; the measure of A and B yields the mean values $\langle a\rangle$ and $\langle b\rangle$, as well as their standard deviations Δa and Δb. More precisely, we prepare N systems ($N \gg 1$) in the state $|\psi\rangle$. For each half we calculate the mean and standard deviation for A and B. We want now to have a relation between Δa and Δb. After some computations:

$$\Delta a \Delta b \geq \frac{1}{2}\left|\left\langle\psi\right|\left[\hat{A},\hat{B}\right]\left|\psi\right\rangle\right| \tag{B.13}$$

B.1.3.2 The Ehrenfest theorem

The time evolution of the mean $\langle a\rangle$ of a physical value A is given by:

$$\frac{\mathrm{d}}{\mathrm{d}t}\langle a\rangle = \frac{1}{\mathrm{i}\hbar}\left\langle\psi\right|\left[\hat{A},\hat{H}\right]\left|\psi\right\rangle + \left\langle\psi\right|\frac{\partial\hat{A}}{\partial t}\left|\psi\right\rangle . \tag{B.14}$$

If the operator $\hat{A}$ is stationary:

$$\frac{\mathrm{d}}{\mathrm{d}t}\langle a\rangle = \frac{1}{\mathrm{i}\hbar}\left\langle\psi\right|\left[\hat{A},\hat{H}\right]\left|\psi\right\rangle . \tag{B.15}$$

From this equation, constants of motion from classical mechanics can be retrieved:

- For a time independent problem, if $\hat{A} = \hat{H}$, the theorem yields energy conservation:

$$\frac{\mathrm{d}\langle E\rangle}{\mathrm{d}t} = 0. \tag{B.16}$$

- Let's consider the motion of a particle in free space *i.e.* $\hat{H} = \hat{\boldsymbol{p}}^2/2m$. $\hat{\boldsymbol{p}}$ commutes with $\hat{H}$ so we find the momentum conservation:

$$\frac{\mathrm{d}\langle\boldsymbol{p}\rangle}{\mathrm{d}t} = 0. \tag{B.17}$$

B.1.3.3 Complete set of commuting observables (CSCO)

The following algebra theorem holds:

If two observables $\hat{A}$ and $\hat{B}$ commute, there exists a basis $\mathcal{E}_H$ made of eigenvectors common to $\hat{A}$ and $\hat{B}$.

We say that a set of operators $\hat{A},\hat{B},\hat{C}\ldots$ is a complete set of commuting observables if the eigenvector basis is unique.

This is important because if we want to know for sure in which state the system is in, we need to measure physical values related to each one of the observables of a CSCO.

B.1.4 Algebraic approach to the quantum harmonic oscillator

The algebraic proof of the eigenvalues and eigenfunctions of the quantum harmonic oscillator is given here. Let us define the normalized variables:

$$\hat{X} = \sqrt{\frac{m\omega}{\hbar}}\hat{x} \quad \text{and} \quad \hat{P} = \frac{P}{\sqrt{m\hbar\omega}} \tag{B.18}$$

Let us loose the hats for now until the end of this paragraph and define the decrement and increment operators respectively:

$$a = \frac{1}{\sqrt{2}}(X + iP) \quad \text{and} \quad a^+ = \frac{1}{\sqrt{2}}(X - iP) \tag{B.19}$$

The Hamiltonian can be rewritten as:

$$H = \hbar\omega(a^+a + \frac{1}{2}) \tag{B.20}$$

We define the number operator:

$$N = a^+a \tag{B.21}$$

N and H share the same eigenvectors are the following properties can be derived:

$$Sp(N) = \mathbb{N} \tag{B.22}$$

$$Sp(H) = \left(\mathbb{N} + \frac{1}{2}\right)\hbar\omega \tag{B.23}$$

$$a^+ |n\rangle = \sqrt{n+1}\,|n+1\rangle \tag{B.24}$$

$$a\,|n\rangle = \sqrt{n}\,|n-1\rangle \tag{B.25}$$

B.2 The canonical ensemble in statistical mechanics

The canonical ensemble in statistical mechanics is recalled in this section and adapted from [Capitelli et al., 2012, Chap. 3]. A more complete description of the different statistical ensembles can be found in Bouchaud et al. [2016]. Let us consider a system of $\mathcal{N}$ particles subdivided by energies as

$$(\mathcal{N}_1, \varepsilon_1), (\mathcal{N}_2, \varepsilon_2), (\mathcal{N}_3, \varepsilon_3), \ldots, (\mathcal{N}_i, \varepsilon_i), \ldots \tag{B.26}$$

where $\mathcal{N}_i$ is the number of particles with energy ϵ_i. The total energy and the total number of particles can be written as

$$\mathcal{N} = \sum_i \mathcal{N}_i \tag{B.27}$$

$$U = \sum_i \mathcal{N}_i \epsilon_i \tag{B.28}$$

Fixing $\mathcal{N}$ and U, the number of micro-states W satisfying the appropriate conditions is:

$$W = \frac{\mathcal{N}!}{\mathcal{N}_1!\mathcal{N}_2!\ldots\mathcal{N}_i!\ldots} = \frac{\mathcal{N}!}{\prod_i \mathcal{N}_i!} \tag{B.29}$$

Starting from the Boltzmann formula for the entropy:

$$S(W) = k\ln(W), \tag{B.30}$$

where k_b is the Boltzmann constant.

The equilibrium state is the state of maximum entropy. Taking into account Stirling's formula and solving the optimization problem with the constraints of constant energy and number of particles yields: The summation

$$\mathcal{Q} = \sum_i \mathrm{e}^{-\beta\epsilon_i} \tag{B.31}$$

is the partition function and we have

$$\frac{\mathcal{N}_i}{\mathcal{N}} = \frac{\mathrm{e}^{-\beta\epsilon_i}}{\mathcal{Q}} \tag{B.32}$$

where $\beta = 1/kT$.

The partition function gives a complete description of the system at equilibrium and all the thermodynamic functions can be extracted from it. The internal energy is thus:

$$U = \mathcal{N}kT^2\left(\frac{\partial\ln\mathcal{Q}}{\partial T}\right)_V = \mathcal{N}kT\left(\frac{\partial\ln\mathcal{Q}}{\partial\ln T}\right)_V \tag{B.33}$$

These formulas are valid only if the different states are not degenerate (we could also consider that the sum are over the micro-states of the system). In the case of degeneracy the formulas have to be slightly modified:

$$\mathcal{Q} = \sum_i g_i\mathrm{e}^{-\beta\epsilon_i} \tag{B.34}$$

$$\frac{\mathcal{N}_i}{\mathcal{N}} = g_i\frac{\mathrm{e}^{-\beta\epsilon_i}}{\mathcal{Q}} \tag{B.35}$$

Appendix C

Operators in cylindrical coordinates

A cylindrical frame is presented in Fig. 3.1. Some results concerning operators in this kind of frame are derived in this section which are relevant in Part II of this thesis. The two moving axis unit vectors $\mathbf{e}_r$ and $\mathbf{e}_\theta$ have non-zero derivatives with respect to θ:

$$\frac{\partial \mathbf{e}_r}{\partial \theta} = \mathbf{e}_\theta \quad \text{and} \quad \frac{\partial \mathbf{e}_\theta}{\partial \theta} = -\mathbf{e}_r \tag{C.1}$$

The gradient, divergence and laplacian operators have the following general definitions for a tensor $\mathbf{u}$ or a scalar variable u:

$$\boldsymbol{\nabla} u = \frac{\partial u}{\partial x_i} \boldsymbol{e}_i \tag{C.2}$$

$$\boldsymbol{\nabla} \cdot \mathbf{u} = \frac{\partial \mathbf{u}}{\partial x_i} \cdot \boldsymbol{e}_i \tag{C.3}$$

$$\nabla^2 u = \boldsymbol{\nabla} \cdot \boldsymbol{\nabla} u \tag{C.4}$$

The scalar gradient in cylindrical coordinates has a straightforward expression:

$$\boldsymbol{\nabla} u = \frac{\partial u}{\partial r} + \frac{1}{r}\frac{\partial u}{\partial \theta} + \frac{\partial u}{\partial z} \tag{C.5}$$

The vector divergence, following Eq. (C.3), is:

$$\begin{aligned} \boldsymbol{\nabla} \cdot \mathbf{u} &= \frac{\partial u_r}{\partial r} + \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{\partial u_z}{\partial z} + \frac{u_r}{r}\frac{\partial \mathbf{e}_r}{\partial \theta} \cdot \mathbf{e}_\theta \\ &= \frac{1}{r}\frac{\partial r u_r}{\partial r} + \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + \frac{\partial u_z}{\partial z} \end{aligned} \tag{C.6}$$

where Eq. (C.1) has been used from the first line to the second. The scalar laplacian is deduced from the previous two equations:

$$\nabla^2 u = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2} + \frac{\partial^2 u}{\partial z^2} \tag{C.7}$$

The divergence of a two-dimensional tensor requires a bit more work. We introduce the tensor $\mathbf{t} = t_{ij}\mathbf{e}_i \otimes \mathbf{e}_j$ (it is used to represent the Euler momentum flux i.e. $\mathbf{t} = \rho \mathbf{u} \otimes \mathbf{u} + p\mathbf{I}$ or viscous momentum flux $\mathbf{t} = \boldsymbol{\tau}$). From Eq. (C.3):

$$\boldsymbol{\nabla}\cdot\mathbf{t} = \frac{\partial t_{ij}\mathbf{e}_i\otimes\mathbf{e}_j}{\partial x_j}\cdot\mathbf{e}_k \tag{C.8}$$

$$= \left[\frac{\partial t_{ij}}{\partial x_k}\mathbf{e}_i\otimes\mathbf{e}_j + t_{ij}\frac{\partial \mathbf{e}_i}{\partial x_k}\otimes\mathbf{e}_j + t_{ij}\mathbf{e}_i\otimes\frac{\partial \mathbf{e}_j}{\partial x_k}\right]\cdot\mathbf{e}_k \tag{C.9}$$

$$= \frac{\partial t_{ij}}{\partial x_j}\mathbf{e}_i + t_{ij}\frac{\partial \mathbf{e}_i}{\partial x_j} + \frac{t_{ir}}{r}\mathbf{e}_i \tag{C.10}$$

$$= \frac{\partial t_{ij}}{\partial x_j}\mathbf{e}_i + \frac{t_{ir}}{r}\mathbf{e}_i + \frac{t_{r\theta}}{r}\mathbf{e}_\theta - \frac{t_{\theta\theta}}{r}\mathbf{e}_r \tag{C.11}$$

Explicitely writing this expression in brackets, the two-dimensional cylindrical divergence is

$$\boldsymbol{\nabla}\cdot\mathbf{t} = \begin{bmatrix} \dfrac{\partial t_{rr}}{\partial r} + \dfrac{1}{r}\dfrac{\partial t_{r\theta}}{\partial\theta} + \dfrac{\partial t_{rz}}{\partial z} - \dfrac{t_{\theta\theta}}{r} + \dfrac{t_{rr}}{r} \\ \dfrac{\partial t_{\theta r}}{\partial r} + \dfrac{1}{r}\dfrac{\partial t_{\theta\theta}}{\partial\theta} + \dfrac{\partial t_{\theta z}}{\partial z} + \dfrac{t_{r\theta}}{r} + \dfrac{t_{\theta r}}{r} \\ \dfrac{\partial t_{zr}}{\partial r} + \dfrac{1}{r}\dfrac{\partial t_{r\theta}}{\partial\theta} + \dfrac{\partial t_{rz}}{\partial z} + \dfrac{t_{zr}}{r} \end{bmatrix} \tag{C.12}$$

Regrouping radiuses above, the following more condensed formula can also be retrieved:

$$\boldsymbol{\nabla}\cdot\mathbf{t} = \begin{bmatrix} \dfrac{1}{r}\dfrac{\partial r t_{rr}}{\partial r} + \dfrac{1}{r}\dfrac{\partial t_{r\theta}}{\partial\theta} + \dfrac{\partial t_{rz}}{\partial z} - \dfrac{t_{\theta\theta}}{r} \\ \dfrac{1}{r}\dfrac{\partial r t_{\theta r}}{\partial r} + \dfrac{1}{r}\dfrac{\partial t_{\theta\theta}}{\partial\theta} + \dfrac{\partial t_{\theta z}}{\partial z} + \dfrac{t_{r\theta}}{r} \\ \dfrac{1}{r}\dfrac{\partial r t_{zr}}{\partial r} + \dfrac{1}{r}\dfrac{\partial t_{r\theta}}{\partial\theta} + \dfrac{\partial t_{rz}}{\partial z} \end{bmatrix} \tag{C.13}$$

Appendix D

Linear Algebra

This summary of basic linear algebra is taken from [Quarteroni et al., 2007, Chap.1].

D.1 Vector space, matrices and linear maps

Definition D.1. *A vector space over the commutative numeric field K is a non-emty set V, whose elements are called vectors, and in which two operations $+$ and $\cdot$ are defined such that:*

- *$(V, +)$ is an Abelian group*
- *$0 \cdot \mathbf{v} = \mathbf{0}$ and $1 \cdot \mathbf{v} = \mathbf{v}$ where 0 and 1 are respectively the zero and unit of K*
- *the following distributive properties hold:*

$$\forall \alpha \in K, \forall \mathbf{v}, \mathbf{w} \in V,\ \alpha(\mathbf{v} + \mathbf{w}) = \alpha\mathbf{v} + \alpha\mathbf{w} \tag{D.1}$$

$$\forall \alpha, \beta \in K, \forall \mathbf{v} \in V,\ (\alpha + \beta)\mathbf{v} = \alpha\mathbf{v} + \beta\mathbf{v} \tag{D.2}$$

- *the following associative property holds:*

$$\forall \alpha, \beta \in K, \forall \mathbf{v} \in V,\ (\alpha\beta)\mathbf{v} = \alpha(\beta\mathbf{v}). \tag{D.3}$$

Definition D.2. *We say that a nonempty part W of V is a vector subspace of V iff W is a vector subspace over K.*

In particular, the set W of the linear combinations of a system of p vectors of V, $\mathbf{v}_1, \ldots, \mathbf{v}_p$, is a vector subspace of V, called the generated subspace or span of the vector system, and is denoted by

$$W = \text{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_p\} \tag{D.4}$$

$$= \{\mathbf{v} = \sum_{i=1}^{p} \alpha_i \mathbf{v}_i \quad \text{with} \quad \alpha_i \in K,\ i = 1, \ldots, p\} \tag{D.5}$$

The system $\mathbf{v}_1, \ldots, \mathbf{v}_p$ is called a system of generators for W. If $W_1, \ldots, W_m$ are vector subspaces of V, then the set

$$S = \{\mathbf{w} : \mathbf{w} = \sum_{i=1}^{m} \mathbf{v}_i \quad \text{with} \quad \mathbf{v}_i \in W_i,\ i = 1, \ldots, m\} \tag{D.6}$$

is also a vector subspace of V. We say that S is the direct sum of the subspaces W_i if any element $\mathbf{s} \in S$ admits a unique representation. In such a case, we shall write $S = \oplus_{i=1}^{m} W_i$.

Definition D.3. *A system of vectors $\mathbf{v}_1, \ldots, \mathbf{v}_p$ of a vector space V is called linearly independent if*

$$\sum_{i=1}^{p} \alpha_i \mathbf{v}_i = 0 \implies \forall i \in [\![1, p]\!]\ \alpha_i = 0 \tag{D.7}$$

Otherwise, the system is called linearly dependent.

We call a basis of V any system of linearly independent generators of V. If $\{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$ is a basis of V, the expression $\mathbf{v} = v_1\mathbf{u}_1 + \ldots + v_n\mathbf{u}_n$ is called the decomposition of $\mathbf{v}$ with respect to the basis and the scalars $v_1, \ldots, v_n \in K$ are the components of $\mathbf{v}$ with respect to the given basis. Moreover the following property holds.

Property D.1. *Let V be a vector space which admits a basis of n vectors. Then every system of linearly independant vectors of V has at most n elements and any other basis of V has n elements. The number n is called the dimension of V and we write $dim(V) = n$.*

If instead, for any n there always exist n linearly independent vectors of V, the vector space is called infinite dimensional.

Definition D.4 (Spectral radius)**.** *The maximum module of the eigenvalues of a matrix A is called the spectral radius $\rho(A)$:*

$$\rho(A) = \max_{\lambda \in \mathrm{Sp}(A)} |\lambda| \tag{D.8}$$

D.2 Similarity transformations

Property D.2 (Schur decomposition)**.** *Given $A \in \mathcal{M}_n(\mathbb{C})$ there exists U unitary such that*

$$U^{-1}AU = U^H AU = \begin{bmatrix} \lambda_1 & \cdots & & b_{1n} \\ 0 & \lambda_2 & & b_{2n} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix} \tag{D.9}$$

where λ_i are the eigenvalues of the matrix.

Property D.3 (Canonical Jordan form)**.** *Let A be any square matrix. There exists a non-singular matrix X that transforms A into a block diagonal matrix J*

$$X^{-1}AX = J = \mathrm{diag}(J_{k_1}(\lambda_1), \ldots, J_{k_l}(\lambda_l)) \tag{D.10}$$

which is called canonical Jordan form where λ_j are the eigenvalues of A and $J_k(\lambda) \in \mathcal{M}_k(\mathbb{C})$ with $J_k(\lambda) = 1$ if $k = 1$ and

$$J_k(\lambda) = \begin{bmatrix} \lambda & 1 & 0 & \ldots & 0 \\ 0 & \lambda & 1 & \ldots & 0 \\ 0 & 0 & \lambda & & 0 \\ \vdots & \vdots & & \ddots & 1 \\ 0 & 0 & & \ldots & \lambda \end{bmatrix} \tag{D.11}$$

otherwise.

Property D.4 (Singular Value Decomposition)**.** *Let $A \in \mathcal{M}_{mn}(\mathbb{C})$. There exists two unitary matrices $U \in \mathcal{M}_m(\mathbb{C})$ and $V \in \mathcal{M}_n(\mathbb{C})$ such that*

$$U^H AV = \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_p) \in \mathcal{M}_{mn}(\mathbb{C}) \tag{D.12}$$

where $p = \min(m, n)$.

Σ is called the Singular Value decomposition of A and $\sigma_1 > \sigma_2 > \ldots > \sigma_p$ are called the singular values of A.

D.3 Matrix Norms

D.3.1 Definitions

Definition D.5 (Consistency)**.** *A matrix norm $||\cdot||$ is said to be compatible or consistent with a vector norm $||\cdot||$ if:*

$$\forall x \in \mathbb{R}^n \ ||Ax|| \leq ||A||\,||x|| \tag{D.13}$$

Definition D.6 (Submultiplicity)**.** *A matrix norm is said to be submultiplicative if*

$$||AB|| \leq ||A||\,||B|| \tag{D.14}$$

Property D.5 (Compatibility of Frobenius norm)**.** *The Frobenius norm*

$$||A||_F = \text{tr}(AA^H) \tag{D.15}$$

is compatible with the 2-norm.

Definition D.7 (Induced norm)**.** *Let $||\cdot||$ be a vector norm. The induced or natural matrix norm is:*

$$||A|| = \sup_{x \neq 0} \frac{||Ax||}{||x||} \tag{D.16}$$

Property D.6 (Canonical induced norms)**.** *The induced 1-norm and infinity-norm are easily computable since*

$$||A||_1 = \max_j \sum_i |a_{ij}| \tag{D.17}$$

$$||A||_\infty = \max_i \sum_j |a_{ij}| \tag{D.18}$$

The 2-norm computation is much more expensive. Denoting by $\sigma_1(A)$ the largest singular value of A:

$$||A||_2 = \sqrt{\rho(A^H A)} = \sqrt{\rho(AA^H)} = \sigma_1(A) \tag{D.19}$$

In particular if A is hermitian:

$$||A||_2 = \rho(A) \tag{D.20}$$

D.3.2 Properties

Property D.7 (Induced norm characterization)**.** *Let $||\cdot||$ be a consistent matrix norm, then:*

$$\forall A \in \mathbb{C}^{n\times n} \ \rho(A) \leq ||A|| \tag{D.21}$$

More precisely:

$$\rho(A) = \inf_{||\cdot||} ||A|| \tag{D.22}$$

Definition D.8 (Matrix convergence)**.** *A sequence of matrix A_k is said to converge to a matrix A if*

$$\lim_{k\to+\infty} ||A_k - A|| = 0 \tag{D.23}$$

Property D.8 (Sequences of matrices)**.** *Let A be a square matrix; then*

$$\lim_k A^k = 0 \iff \rho(A) < 1 \iff \sum_k A^k = (I - A)^{-1} \tag{D.24}$$

Moreover if $\rho(A) < 1$ and $||\cdot||$ is a matrix norm such that $||A|| < 1$ then $I - A$ is invertible and the following inequalities hold

$$\frac{1}{1 + ||A||} \leq ||(I - A)^{-1}|| \leq \frac{1}{1 - ||A||} \tag{D.25}$$

Appendix E

The Dirac distribution

The brief description of distribution presented here is adapted from [Basdevant and Dalibard, 2005, Appendix B].

E.1 Definition of $\delta(x)$

In physics the notion of point object is often used. The mass (or charge) density $\rho(\mathbf{r})$ of such a point (at $\mathbf{r}_0$) is not a function in the usual sense, since it is zero except at $\mathbf{r}_0$, and its integral is finite:

$$\int \rho(\mathbf{r})\, d^3r = m. \tag{E.1}$$

The delta function introduced by the physicist Paul Dirac allows to describe such a density. Its mathematical definition has been made by Laurent Schwartz in the frame of the distribution theory which will be briefly presented in this section. The formalism most often used in physics, although abusive strictly speaking, is adopted in the rest of this section. Let us a consider a real variable x. The Dirac delta "function" is defined as:

$$\delta(x) = 0 \quad \text{for} \quad x \neq 0 \quad \text{and} \quad \int_{\mathbb{R}} \delta(x)\, dx = 1. \tag{E.2}$$

The real definition of the delta distribution is made through its application on functions. For any function $F(x)$ continuous in $x = 0$, we have by definition:

$$\int F(x)\, \delta(x)\, dx = F(0). \tag{E.3}$$

By change of variable, the function $\delta(x - x_0)$ is:

$$\int F(x)\, \delta(x - x_0)\, dx = F(x_0). \tag{E.4}$$

The generalization to multiple dimensions is straightforward:

$$\delta(\mathbf{r} - \mathbf{r}_0) = \delta(x - x_0)\, \delta(y - y_0)\, \delta(z - z_0), \tag{E.5}$$

and

$$\int F(\mathbf{r})\, \delta(\mathbf{r} - \mathbf{r}_0)\, d^3r = F(\mathbf{r}_0). \tag{E.6}$$

Other importants properties of the delta function are:

- $\delta(x)$ is an even function: $\delta(x - x_0) = \delta(x_0 - x)$.
- We have $\delta(ax) = \frac{1}{|a|}\delta(x)$ (a real).

E.2 Examples of functions that tend to $\delta(x)$

It is possible to make close-to-point distributions thanks to functions concentrated at the neighborhood of a point. To do so we consider functions that depend on a parameter controlling the width of the function:

1. Let y_ϵ be the sequence of functions such that:

$$y_\epsilon = \begin{cases} 1/\epsilon & \text{for} \quad |x| \leq \epsilon/2 \\ 0 & \text{otherwise} \end{cases} \tag{E.7}$$

We have:

$$\int F(x)\, y_\epsilon(x)\, dx = \frac{1}{\epsilon} \int_{-\epsilon/2}^{\epsilon/2} F(x) dx = F(\theta\epsilon/2) \quad \text{with} \quad -1 \leq \theta \leq 1 \tag{E.8}$$

where the mean value theorem has been used for the last equality. Hence:

$$\int F(x)\, y_\epsilon(x)\, dx \xrightarrow[\epsilon \to 0]{} F(0). \tag{E.9}$$

2. The gaussian functions with standard deviation σ are another sequence of functions that tend to the dirac delta function:

$$g_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-x^2/2\sigma^2\right). \tag{E.10}$$

These sequences of functions are shown in Fig. E.1.

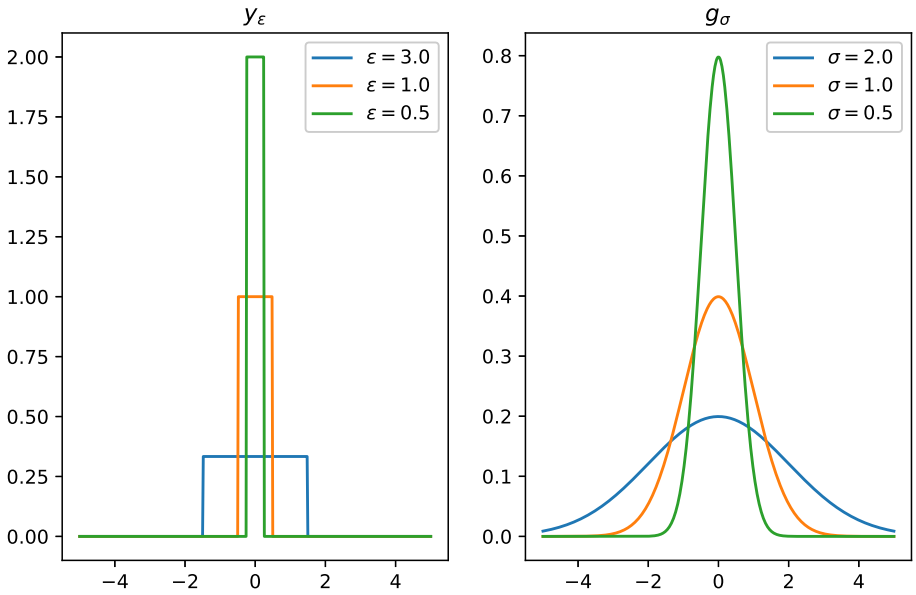


Figure E.1: Step and gaussian sequences of functions

E.3 Distributions

The notions presented previously are rigorously defined in the distribution theory which reinterprets functions as linear functionals acting on test functions. Let us consider the subvector space S:

$$S = \{\phi \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{C}) \, \forall k \in \mathbb{N} \; \phi^{(k)} \text{ tends to 0 at infinity faster than any power of } 1/|x|\} \tag{E.11}$$

For example $\exp(-x^2)$ belongs to S.

A continuous linear functional f on the space S is a linear map from S to $\mathbb{C}$ that is continuous. To each test function $\phi \in S$ it associates a complex number (f, ϕ). It is continuous in the following sense: if the sequence of functions $\phi_1, \phi_2, \ldots, \phi_n$ tend to zero in S, the sequence of numbers $(f, \phi_1), (f, \phi_2), \ldots, (f, \phi_n)$ tend to zero.

Such a continuous linear functional is called a tempered distribution. Examples of tempered distributions are:

1. Let $f(x)$ be a locally integrable function that is bounded by a power of $|x|$ when $|x| \to \infty$. A functional may be associated to f, that we also denote by f, by the formula:
$$(f, \phi) = \int f(x) \, \phi(x) \, dx \quad \phi \in S. \tag{E.12}$$

2. The Dirac distribution is the functional that associates any function $\phi(x) \in S$ to the number $\phi(0)$, which we denote:
$$(\delta, \phi) = \phi(0). \tag{E.13}$$
It is common for physicists (but abusive) to write:
$$\int \delta(x) \, \phi(x) \, dx = \phi(0). \tag{E.14}$$
In the examples mentioned at the beginning of the section, the assertion that y_ϵ or g_σ tend to δ is incorrect. However the assertion
$$(g_\sigma, \phi) \xrightarrow[\sigma \to 0]{} (\delta, \phi) \quad \forall \phi \in S \tag{E.15}$$
is correct. It is said that g_σ tends to δ in the sense of distributions.

E.4 Derivation of a distribution

When the distribution is associated to a derivable function $f(x)$, one can write by integration by parts:

$$(f', \phi) = \int_{\mathbb{R}} \frac{\mathrm{d}f(x)}{\mathrm{d}x} \, \phi(x) \, dx = - \int_{\mathbb{R}} f(x) \, \frac{\mathrm{d}\phi}{\mathrm{d}x} \, dx = -(f, \phi') \tag{E.16}$$

The derivative of a linear functional is defined by:

$$\left(\frac{\mathrm{d}f}{\mathrm{d}x}, \phi\right) = -\left(f, \frac{\mathrm{d}\phi}{\mathrm{d}x}\right). \tag{E.17}$$

From this result:

1. The derivative of the Dirac distribution is:

$$(\delta', \phi) = -\phi'(0). \tag{E.18}$$

2. Let Θ be the Heaviside function. Then the following equality holds in the distribution sense:

$$\frac{\mathrm{d}\Theta(x)}{\mathrm{d}x} = \delta(x). \tag{E.19}$$

3. In three dimensions: $\Delta(\frac{1}{r}) = -4\pi\delta(\mathbf{r})$. Indeed:

$$\int \Delta\left(\frac{1}{r}\right)\phi(\mathbf{r})d^3r = -\int \left[\nabla\left(\frac{1}{r}\right)\cdot\nabla\phi\right] 4\pi r^2 dr \tag{E.20}$$

$$= \int_0^{+\infty} \frac{1}{r^2}\,\frac{\partial\phi}{\partial r}\,4\pi r^2 dr \tag{E.21}$$

$$= -4\pi\phi(0). \tag{E.22}$$

Appendix F

Orthogonal functions and expansions

Expansions in orthogonal functions are common in partial differential equations and quantum mechanics. The main definitions and some examples are given but more can be found in [Jackson, 1999, Chapters 2-3].

F.1 Definitions

Let us consider a real interval $[a, b]$ and $\mathcal{L}^2([a, b], \mathbb{C})$ the Hilbert space of square integrable functions with the following inner product:

$$(f, g) = \int_a^b f^*(x)g(x)dx. \tag{F.1}$$

where the asterix denotes the complex conjugate of a complex number.

A set of functions $(U_n) \in \mathcal{L}^2([a, b], \mathbb{C})$ is called orthogonal if:

$$(U_m, U_n) = 0 \quad \text{if} \quad m \neq n \tag{F.2}$$

and is orthonormal if:

$$(U_m, U_n) = \delta_{mn} \tag{F.3}$$

A set of functions is called complete if it satisfies the *completeness* or *closure* relation:

$$\sum_{n=0}^{+\infty} U_n^*(x')U_n(x) = \delta(x - x') \tag{F.4}$$

where δ is the Dirac delta function presented in E.

If (U_n) is a complete set of orthonormal functions any square integrable function f can be expanded in this set:

$$f(x) = \sum_{n=0}^{+\infty} a_n U_n(x) \tag{F.5}$$

with

$$a_n = \int_a^b U_n^*(x)f(x)dx. \tag{F.6}$$

F.2 Fourier series

F.2.1 Real Fourier series

The most famous complete set of orthogonal functions are sines and cosines, an expansion in terms of them being a Fourier series. If the interval is $(-a/2, a/2)$ then the orthonormal functions are

$$\sqrt{\frac{2}{a}}\sin\left(\frac{2\pi nx}{a}\right), \quad \sqrt{\frac{2}{a}}\cos\left(\frac{2\pi nx}{a}\right) \quad n \in \mathbb{N} \tag{F.7}$$

The series expansion yields:

$$f(x) = \frac{A_0}{2} + \sum_{n=1}^{+\infty}\left[A_n\cos\left(\frac{2\pi nx}{a}\right) + B_n\sin\left(\frac{2\pi nx}{a}\right)\right] \tag{F.8}$$

where

$$A_n = \frac{2}{a}\int_{-a/2}^{a/2} f(x)\cos\left(\frac{2\pi nx}{a}\right)dx \tag{F.9}$$

$$B_n = \frac{2}{a}\int_{-a/2}^{a/2} f(x)\sin\left(\frac{2\pi nx}{a}\right)dx \tag{F.10}$$

F.2.2 Complex Fourier series

If f is a complex function of a real variable x then f can be expanded in terms of the orthonormal functions

$$\frac{1}{\sqrt{a}}\exp\left(\frac{2\mathrm{i}\pi nx}{a}\right) \tag{F.11}$$

and one can write

$$f(x) = \sum_{n\in\mathbb{Z}} c_n \frac{1}{\sqrt{a}}\exp\left(\frac{2\mathrm{i}\pi nx}{a}\right) \tag{F.12}$$

$$c_n = \frac{1}{\sqrt{a}}\int_{-a/2}^{a/2}\exp\left(-\frac{2\mathrm{i}\pi nx'}{a}\right)f(x')dx'. \tag{F.13}$$

In the specific case of a real valued function the following relationship holds for the c_n:

$$c_n = c_{-n}^* \qquad \forall n \in \mathbb{Z}. \tag{F.14}$$

The Parseval identity is verified:

$$\sum_{n\in\mathbb{Z}} |c_n|^2 = \int_{-a/2}^{a/2} |f(x)|^2 dx. \tag{F.15}$$

F.3 Fourier transform

F.3.1 Definition

Many conventions are applied so to get things straight if $f \in \mathcal{L}^2(\mathbb{R})$, its Fourier transform is:

$$\mathcal{F}(f) : k \mapsto \hat{f}(k) = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} f(x)\, \mathrm{e}^{-\mathrm{i}kx} dx. \tag{F.16}$$

The following relations hold:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} \hat{f}(k) \mathrm{e}^{\mathrm{i}kx} dk \tag{F.17}$$

$$\mathcal{F}\left(\frac{\mathrm{d}f}{\mathrm{d}x}\right)(k) = \mathrm{i}k\hat{f}(k) \tag{F.18}$$

$$\mathcal{F}(xf)(k) = i\hat{f}'(k) \tag{F.19}$$

F.3.2 Gaussian Fourier transform

The Fourier transform of a Gaussian is a Gaussian. Starting from:

$$f(x) = A \exp\left(-\frac{x^2}{\sigma^2}\right) \tag{F.20}$$

Deriving and applying the Fourier transform:

$$\hat{f}'(k) = -\frac{\sigma k}{2} \hat{f}(k) \tag{F.21}$$

Solving the equation and finding the appropriate constant through the Gauss integral yields:

$$\hat{f}(k) = \frac{A\sigma}{\sqrt{2}} \exp\left(-\frac{\sigma^2 k^2}{4}\right) \tag{F.22}$$

F.3.3 Inverse of the Fourier transform

The Fourier transform can be inverted and we have:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} \hat{f}(x)\, \mathrm{e}^{\mathrm{i}kx} dx. \tag{F.23}$$

F.3.4 Isometry of the Fourier transform: the Parseval-Plancherel theorem

One fundamental property of the fourier transform is the Parseval-Plancherel theorem:

$$\int \hat{f}_1^*(k)\, \hat{f}_2(k)\, dk = \int f_1^*(x)\, f_2(x)\, dx. \tag{F.24}$$

Introducing the scalar product in $\mathcal{L}^2(\mathbb{R})$:

$$\langle f_1, f_2 \rangle = \int f_1^*(x)\, f_2(x)\, dx \tag{F.25}$$

the Parseval-Plancherel theorem states that the Fourier transform is an isometry: $\langle f_1, f_2 \rangle = \langle \hat{f}_1, \hat{f}_2 \rangle$.

F.3.5 Relation between the Fourier transform and Fourier series

Going back to the Fourier series, the Fourier transform is obtained in the limit $a \to +\infty$. Starting from:

$$f(x) = \sum_{n\in\mathbb{Z}} \hat{f}_n \frac{1}{\sqrt{a}} \exp\left(\frac{2\mathrm{i}\pi n x}{a}\right) \tag{F.26}$$

$$\hat{f}_n = \frac{1}{\sqrt{a}} \int_{-a/2}^{a/2} \exp\left(-\frac{2\mathrm{i}\pi n x'}{a}\right) f(x')dx' \tag{F.27}$$

and making the following correspondances when $a \to +\infty$:

$$\frac{2\pi n}{a} \to k \tag{F.28}$$

$$\sum_m \to \int_R dm = \frac{a}{2\pi} \int_R dk \tag{F.29}$$

$$\hat{f}_n \to \sqrt{\frac{2\pi}{a}} \hat{f}(k) \tag{F.30}$$

Bibliography

Steven Chu and Arun Majumdar. Opportunities and challenges for a sustainable energy future. *Nature*, 488:294–303, 08 2012. doi: 10.1038/nature11475. (Cited on page 1.)

Xingcai Lu, Dong Han, and Zhen Huang. Fuel design and management for the control of advanced compression-ignition combustion modes. *Progress in Energy and Combustion Science*, 37(6):741–783, 2011. ISSN 0360-1285. doi: https://doi.org/10.1016/j.pecs.2011.03.003. URL https://www.sciencedirect.com/science/article/pii/S0360128511000219. (Cited on page 1.)

Robert Tacina, Chien-Pei Mao, and Changlie Wey. Experimental investigation of a multiplex fuel injector module with discrete jet swirlers for low emission combustors. *41st Aerospace Sciences Meeting and Exhibit*, 09 2004. doi: 10.2514/6.2004-135. (Cited on page 1.)

Antonio Cavaliere and Mara de Joannon. Mild combustion. *Progress in Energy and Combustion Science*, 30(4):329–366, 2004. ISSN 0360-1285. doi: https://doi.org/10.1016/j.pecs.2004.02.003. URL https://www.sciencedirect.com/science/article/pii/S0360128504000127. (Cited on page 1.)

Yiguang Ju and Wenting Sun. Plasma assisted combustion: Dynamics and chemistry. *Progress in Energy and Combustion Science*, 48:21–83, June 2015. doi: 10.1016/j.pecs.2014.12.002. URL https://doi.org/10.1016/j.pecs.2014.12.002. (Cited on pages 1, 2 and 3.)

Andrey Starikovskiy and Nickolay Aleksandrov. Plasma-assisted ignition and combustion. *Progress in Energy and Combustion Science*, 39(1):61 – 110, 2013. ISSN 0360-1285. doi: https://doi.org/10.1016/j.pecs.2012.05.003. URL http://www.sciencedirect.com/science/article/pii/S0360128512000354. (Cited on page 1.)

Charles Kruger, Christophe Laux, Lan Yu, Denis Packan, and Laurent Pierrot. Nonequilibrium discharges in air and nitrogen plasmas at atmospheric pressure. *Pure and Applied Chemistry*, 74:337–348, 01 2002. doi: 10.1351/pac200274030337. (Cited on page 3.)

Séverine Barbosa, Guillaume Pilla, Deanna Lacoste, Philippe Scouflaire, Sébastien Ducruix, Christophe Laux, and Denis Veynante. Influence of nanosecond repetitively pulsed discharges on the stability of a swirled propane/air burner representative of an aeronautical combustor. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 373, 08 2015. doi: 10.1098/rsta.2014.0335. (Cited on page 3.)

Da Xu, Deanna Lacoste, and Christophe Laux. Ignition of quiescent lean propane–air mixtures at high pressure by nanosecond repetitively pulsed discharges. *Plasma Chemistry and Plasma Processing*, 36, 01 2016. doi: 10.1007/s11090-015-9680-3. (Cited on page 3.)

Sébastien Celestin. *Study of the dynamics of streamers in air at atmospheric pressure*. PhD thesis, 2008. URL http://www.theses.fr/2008ECAP1115. Thèse de doctorat dirigée par Bourdon, Anne et Rousseau, Antoine Physique Châtenay-Malabry, Ecole centrale de Paris 2008. (Cited on pages 3, 27, 59, 97, 99, 101 and 106.)

Fabien Tholin. *Numerical simulation of nanosecond repetitively pulsed discharges in air at atmospheric pressure : Application to plasma-assisted combustion*. PhD thesis, 2012. URL http://www.theses.fr/2012ECAP0055. Thèse de doctorat dirigée par Bourdon, Anne Physique Châtenay-Malabry, Ecole centrale de Paris 2012. (Cited on pages 3, 99, 106, 202 and 214.)

Maria Luis Gracio Bilro Castela. *Direct Numerical Simulations of plasma-assisted ignition in quiescent and turbulent flow conditions*. Theses, Université Paris-Saclay, May 2016. URL https://tel.archives-ouvertes.fr/tel-01356708. (Cited on pages 3, 4, 229 and 233.)

Dario Maestro. *Large Eddy Simulations of the interactions between flames and thermal phenomena : application to wall heat transfer and combustion control*. PhD thesis, Institut National Polytechnique de Toulouse, 2018. URL https://oatao.univ-toulouse.fr/24184/. (Cited on pages 3 and 47.)

V. Joncquieres. *Modélisation et simulation numérique des moteurs à effet Hall*. PhD thesis, Institut National Polytechnique de Toulouse, 2019. (Cited on pages 3, 57, 60, 92, 129, 142 and 152.)

J.A. Bittencourt. *Fundamentals of Plasma Physics*. Springer, third edition, 2004. (Cited on pages 7, 19, 21, 22, 52 and 59.)

Arthur Sherman. Magnetohydrodynamic channel flows with nonequilibrium ionization. *The Physics of Fluids*, 9(9):1782–1787, 1966. doi: 10.1063/1.1761933. URL https://aip.scitation.org/doi/abs/10.1063/1.1761933. (Cited on page 9.)

F. Taccogna and L. Garrigues. Latest progress in hall thrusters plasma modelling. *Reviews of Modern Plasma Physics*, 3(1), July 2019. doi: 10.1007/s41614-019-0033-1. URL https://doi.org/10.1007/s41614-019-0033-1. (Cited on page 9.)

J-L. Raimbault. *Introduction to the kinetic theory of weakly ionized plasma*. Laboratoire de Physique des Plasmas, 2018. (Cited on pages 10, 18, 50, 59, 60 and 214.)

John David Jackson. *Classical Electrodynamics*. John Wiley & Sons, 1999. (Cited on pages 10, 59, 73, 74, 75, 76, 77, 78 and 315.)

Ludwig Boltzmann. Weitere studien über das wärmegleichgewicht unter gasmolekülen. *Sitzungsberichte Akademie der Wissenschaften*, 66:275–370, 1872. (Cited on page 15.)

N. Liu. *Dynamics of positive and negative streamers in sprites*. PhD thesis, The Pennsylvania State University, 2006. (Cited on page 22.)

V.I. Kisin Yuri P. Raizer, John E. Allen. *Gas discharge physics*. Springer, corrected edition, 1991. (Cited on page 22.)

A. Fridman. *Plasma Chemistry*. Cambridge University Press, 2008. (Cited on page 22.)

Eduard M. Bazelian; Yuri P. Raizer. *Spark Discharge*. CRC Press, 1998. (Cited on page 24.)

M. B. Zheleznyak, A. K. Mnatsakanyan, and S. V. Sizykh. Photoionization of nitrogen and oxygen mixtures by radiation from a gas discharge. *High Temp.*, 20:357–362, 1982. (Cited on page 27.)

Hiroaki Nishikawa. *I do like CFD, VOL.1: Governing Equations and Exact Solutions*. 02 2017. (Cited on pages 36 and 38.)

Thierry Poinsot and Denis Veynante. *Theoretical and Numerical Combustion*. Third edition, 2012. (Cited on pages 36, 38, 41 and 44.)

Eleuterio Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. 01 2009. doi: 10.1007/b79761. (Cited on pages 38, 104, 129, 134, 136, 137, 138, 139, 140, 142, 144, 145, 153 and 154.)

J.O. Hirschfelder, C.F. Curtiss, R.B. Bird, University of Wisconsin-Milwaukee, University of Wisconsin. Theoretical Chemistry Laboratory, and University of Wisconsin. Naval Research Laboratory. *The Molecular Theory of Gases and Liquids*. Molecular Theory of Gases and Liquids. Wiley, 1954. ISBN 9780471400653. (Cited on page 39.)

V. Giovangigli. *Multicomponent Flow Modeling*. 3Island Press, 1999. ISBN 9781461215813. URL `https://books.google.fr/books?id=E4QgswEACAAJ`. (Cited on page 39.)

Michael A. Lieberman and Allan J. Lichtenberg. *Principles of Plasma Discharges and Material Processings*. John Wiley and Sons, second edition, 2005. (Cited on page 40.)

Stephen B. Pope. *Turbulent Flows*. Cambridge University Press, 2000. doi: 10.1017/CBO9780511840531. (Cited on page 46.)

Norbert Peters. *Turbulent Combustion*. Cambridge Monographs on Mechanics. Cambridge University Press, 2000. doi: 10.1017/CBO9780511612701. (Cited on pages 47 and 48.)

G Hagelaar and L. Pitchford. Solving the boltzmann equation to obtain electron transport coefficients and rate coefficients for fluid models. *Plasma Sources Science and Technology*, 14, 2005. doi: 10.1088/0963-0252/14/4/011. (Cited on pages 50 and 51.)

M Capitelli, G Colonna, and A d'Angola. *Fundamental Aspects of Plasma Chemical Physics*. Springer, Berlin, 2012. (Cited on pages 56 and 302.)

Jeffrey I. Steinfeld. *An Introduction to Modern Molecular Spectroscopy*. The MIT Press, second edition, 1986. (Cited on page 58.)

Robert J Kee, Michael E Coltrin, Peter Glarborg, and Huayang Zhu. *Chemically Reacting Flow: Theory, Modeling and Simulation*. Wiley, September 2017. doi: 10.1002/9781119186304. URL `https://doi.org/10.1002/9781119186304`. (Cited on page 57.)

Lionel Cheng, Nicolas Barleon, Olivier Vermorel, Benedicte Cuenot, and Anne Bourdon. Avip: a low temperature plasma code. *arXiv preprint arXiv:2201.01291*, 2021. (Cited on page 65.)

Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. *Numerical Mathematics*, volume 37. 01 2007. ISBN 978-1-4757-7394-1. doi: 10.1007/b98885. (Cited on pages 65, 87, 91 and 307.)

Charles Hirsch. *Numerical Computation of Internal and External Flows: The Fundamentals of Computational Fluid Dynamics*. Elsevier, second edition, 2007. (Cited on pages 66, 67, 109, 111, 113, 115, 118, 139 and 143.)

N. Lamarque. *Numerical schemes and boundary conditions for the LES of two-phase flows in helicopter chambers*. PhD thesis, CERFACS, 2007. (Cited on pages 67, 82, 112, 115, 136, 143 and 160.)

E. Suli. Finite volume methods on distorted meshes : stability, accuracy, adaptativity. *Tech. Rep. NA89/6. Oxford University Computing Laboratory*, 1989. (Cited on page 68.)

Z. Wang, Krzysztof Fidkowski, Rémi Abgrall, Francesco Bassi, Doru Caraeni, Andrew Cary, H. Deconinck, Ralf Hartmann, Koen Hillewaert, H.T. Huynh, Norbert Kroll, Georg May, Per-Olof Persson, Bram van Leer, and Miguel Visbal. High-order cfd methods: Current status and perspective. *International Journal for Numerical Methods in Fluids*, 72:811–845, 01 2013. doi: 10.1002/fld.3767. (Cited on page 71.)

John Harris and Horst Stoecker. *The Handbook of Mathematics and Computational Science*. 01 1998. ISBN 978-1-4612-5319-8. doi: 10.1007/978-1-4612-5317-4. (Cited on page 71.)

V. Auffray. *Etude comparative de schémas numériques pour la modélisation de phénomènes diffusifs sur maillages multiéléments*. PhD thesis, CERFACS, 2007. (Cited on pages 82 and 83.)

Yousef Saad. *Iterative Methods for Sparse Linear Systems*. 01 2003. doi: 10.1137/1.9780898718003.ch4. (Cited on pages 87 and 100.)

Robert D. Falgout. An introduction to algebraic multigrid. *Computing in Science and Engg.*, 8(6):24–33, nov 2006. ISSN 1521-9615. doi: 10.1109/MCSE.2006.105. URL https://doi.org/10.1109/MCSE.2006.105. (Cited on page 92.)

Francesc Verdugo and Wolfgang Wall. Unified computational framework for the efficient solution of n-field coupled problems with monolithic schemes. *Computer Methods in Applied Mechanics and Engineering*, 310, 05 2016. doi: 10.1016/j.cma.2016.07.016. (Cited on page 93.)

Satish Balay, Shrirang Abhyankar, Mark F. Adams, Steven Benson, Jed Brown, Peter Brune, Kris Buschelman, Emil M. Constantinescu, Lisandro Dalcin, Alp Dener, Victor Eijkhout, William D. Gropp, Václav Hapla, Tobin Isaac, Pierre Jolivet, Dmitry Karpeev, Dinesh Kaushik, Matthew G. Knepley, Fande Kong, Scott Kruger, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Lawrence Mitchell, Todd Munson, Jose E. Roman, Karl Rupp, Patrick Sanan, Jason Sarich, Barry F. Smith, Stefano Zampini, Hong Zhang, Hong Zhang, and Junchao Zhang. PETSc Web page. https://petsc.org/, 2021. URL https://petsc.org/. (Cited on pages 92 and 93.)

Robert D. Falgout and Ulrike Meier Yang. hypre: A library of high performance preconditioners. In Peter M. A. Sloot, Alfons G. Hoekstra, C. J. Kenneth Tan, and Jack J. Dongarra, editors, *Computational Science — ICCS 2002*, pages 632–641, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. ISBN 978-3-540-47789-1. (Cited on page 93.)

B Bagheri, J Teunissen, Ute Ebert, Markus M. Becker, S Chen, O. Ducasse, O. Eichwald, D. Loffhagen, Alejandro Luque, Db Diana Mihailova, J.-M. Plewa, Jan van Dijk, and Mohammed Yousfi. Comparison of six simulation codes for positive streamers in air. *Plasma Sources Science and Technology*, 2018. (Cited on pages 97, 98, 106, 189, 190, 191, 197, 201 and 208.)

D.L. Scharfetter and H.K. Gummel. Large-signal analysis of a silicon read diode oscillator. *IEEE Transactions on Electron Devices*, 16(1):64–77, 1969. doi: 10.1109/T-ED.1969.16566. (Cited on pages 106, 107 and 113.)

A. A. Kulikovsky. A more accurate Scharfetter-Gummel algorithm of electron transport for semiconductor and gas discharge simulation. *Journal of Computational Physics*, 1995. ISSN 00219991. doi: 10.1006/jcph.1995.1123. (Cited on pages 106, 107, 108, 113, 114 and 121.)

Ashish Sharma, Vivek Subramaniam, Evrim Solmaz, and Laxminarayan L Raja. Fully coupled modeling of nanosecond pulsed plasma assisted combustion ignition. *Journal of Physics D: Applied Physics*, 52(9):095204, dec 2018. doi: 10.1088/1361-6463/aaf690. URL https://doi.org/10.1088/1361-6463/aaf690. (Cited on page 106.)

Markus Kantner. Generalized scharfetter–gummel schemes for electro-thermal transport in degenerate semiconductors using the kelvin formula for the seebeck coefficient. *Journal of Computational Physics*, 402:109091, 2020. (Cited on page 106.)

Sergei K. Godunov and I. Bohachevsky. Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics. *Matematičeskij sbornik*, 47(89)(3):271–306, 1959. URL https://hal.archives-ouvertes.fr/hal-01620642. (Cited on pages 109 and 110.)

Bram van Leer. Towards the ultimate conservative difference scheme. ii. monotonicity and conservation combined in a second-order scheme. *Journal of Computational Physics*, 14(4): 361–370, 1974. ISSN 0021-9991. doi: https://doi.org/10.1016/0021-9991(74)90019-9. URL https://www.sciencedirect.com/science/article/pii/0021999174900199. (Cited on pages 109, 110, 111 and 115.)

P. K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 21(5):995–1011, 1984. ISSN 00361429. URL http://www.jstor.org/stable/2156939. (Cited on pages 109, 110, 111, 113, 115 and 146.)

Peter Lax and Burton Wendroff. System of conservation laws. *Comm. Pure Appl. Math*, 13: 217–237, 1960. (Cited on pages 109, 110 and 113.)

J. Carmona. *Modelling of two-phase flow phenomena in aeronautical Airblast injectors*. PhD thesis, Institut National Polytechnique de Toulouse, 2021. (Cited on pages 129 and 152.)

F. Alauzet and A. Loseille. High-order sonic boom modeling based on adaptive methods. *Journal of Computational Physics*, 229(3):561–593, 2010. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2009.09.020. URL https://www.sciencedirect.com/science/article/pii/S0021999109005129. (Cited on page 142.)

T.J Poinsot and S.K Lelef. Boundary conditions for direct simulations of compressible viscous flows. *Journal of Computational Physics*, 101(1):104–129, 1992. ISSN 0021-9991. doi: https://doi.org/10.1016/0021-9991(92)90046-2. URL https://www.sciencedirect.com/science/article/pii/0021999192900462. (Cited on page 143.)

V. Moureau, G. Lartigue, Y. Sommerer, C. Angelberger, O. Colin, and T. Poinsot. Numerical methods for unsteady compressible multi-component reacting flows on fixed and moving grids. *Journal of Computational Physics*, 202(2):710–736, 2005. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2004.08.003. URL https://www.sciencedirect.com/science/article/pii/S0021999104003134. (Cited on page 143.)

Xi Deng, Pierre Boivin, and Feng Xiao. A new formulation for two-wave riemann solver accurate at contact interfaces. *Physics of Fluids*, 31:046102, 04 2019. doi: 10.1063/1.5083888. (Cited on pages 144, 153 and 154.)

J. Vanharen. *High-order numerical methods for unsteady flows around complex geometries*. PhD thesis, Institut National Polytechnique de Toulouse, 2017. (Cited on page 153.)

David G. Goodwin, Raymond L. Speth, Harry K. Moffat, and Bryan W. Weber. Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes. https://www.cantera.org, 2021. Version 2.5.1. (Cited on pages 159 and 165.)

Benedetta Franzelli, Eleonore Riber, L.Y.M. Gicquel, and Thierry Poinsot. Large eddy simulation of combustion instabilities in a lean partially premixed swirled flame. *Combustion and Flame*, 159:621–637, 02 2012. doi: 10.1016/j.combustflame.2011.08.004. (Cited on page 159.)

Gregory P. Smith, David M. Golden, Michael Frenklach, Nigel W. Moriarty, Boris Eiteneer, Mikhail Goldenberg, C. Thomas Bowman, Ronald K. Hanson, Soonho Song, William C. Gardiner, Jr. Vitali V. Lissianski, and Zhiwei Qin. URL http://www.me.berkeley.edu/gri_mech/. (Cited on page 160.)

Antonio Tejero-del Caz, Vasco Guerra, D. Goncalves, M. Lino da Silva, L. Marques, N. Pinhao, Carlos Pintassilgo, and L.L. Alves. The lisbon kinetics boltzmann solver. *Plasma Sources Science and Technology*, 28(4):043001, apr 2019. doi: 10.1088/1361-6595/ab0537. (Cited on page 183.)

Phelps. PHELPS database, retrieved on July 30, 2020. www.lxcat.net/Phelps. (Cited on page 183.)

A Bourdon, V P Pasko, N Y Liu, S Célestin, P Ségur, and E Marode. Efficient models for photoionization produced by non-thermal gas discharges in air based on radiative transfer and the helmholtz equations. 16(3):656–678, aug 2007. doi: 10.1088/0963-0252/16/3/026. URL https://doi.org/10.1088/0963-0252/16/3/026. (Cited on page 197.)

R Morrow and J J Lowke. Streamer propagation in air. *Journal of Physics D: Applied Physics*, 30(4):614–627, February 1997. doi: 10.1088/0022-3727/30/4/017. URL https://doi.org/10.1088/0022-3727/30/4/017. (Cited on pages 199, 201 and 202.)

David Z Pai, Deanna A Lacoste, and Christophe O Laux. Nanosecond repetitively pulsed discharges in air at atmospheric pressure—the spark regime. 19(6):065015, nov 2010. doi: 10.1088/0963-0252/19/6/065015. URL https://doi.org/10.1088/0963-0252/19/6/065015. (Cited on page 213.)

Ciprian Dumitrache, Arnaud Gallant, N Minesi, Sergey A. Stepanyan, G D Stancu, and C O Laux. Hydrodynamic regimes induced by nanosecond pulsed discharges in air: mechanism of vorticity generation. *Journal of Physics D: Applied Physics*, 2019. (Cited on page 229.)

M. Capitelli, Ferreira C.M., B.F. Gordiets, and A.I. Osipov. *Plasma Kinetics in Atmospheric Gases*. Springer, 2000. (Cited on page 234.)

Roger C. Millikan and Donald R. White. Systematics of vibrational relaxation. *The Journal of Chemical Physics*, 39(12):3209–3213, 1963. doi: 10.1063/1.1734182. (Cited on page 234.)

D. Rusterholtz, D. Lacoste, G. Stancu, David Pai, and Christophe Laux. Ultrafast heating and oxygen dissociation in atmospheric pressure air by nanosecond repetitively pulsed discharges. *Journal of Physics D Applied Physics*, 46:4010–, 11 2013. doi: 10.1088/0022-3727/46/46/464010. (Cited on page 235.)

Nikolay Popov. Fast gas heating initiated by pulsed nanosecond discharge in atmospheric pressure air. 01 2013. ISBN 978-1-62410-181-6. doi: 10.2514/6.2013-1052. (Cited on page 235.)

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016. (Cited on pages 246, 250, 251 and 254.)

Michael Nielsen. *Deep Learning and Neural Networks*. The MIT Press, 2016. (Cited on pages 247, 249 and 251.)

Kevin Jarrett, Koray Kavukcuoglu, Marc'Aurelio Ranzato, and Yann LeCun. What is the best multi-stage architecture for object recognition? In *2009 IEEE 12th International Conference on Computer Vision*, pages 2146–2153, 2009. doi: 10.1109/ICCV.2009.5459469. (Cited on page 250.)

Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013. (Cited on page 250.)

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *IEEE International Conference on Computer Vision (ICCV 2015)*, 1502, 02 2015. doi: 10.1109/ICCV.2015.123. (Cited on page 250.)

G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2(4):303–314, December 1989. doi: 10.1007/bf02551274. URL https://doi.org/10.1007/bf02551274. (Cited on page 251.)

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989. ISSN 0893-6080. doi: https://doi.org/10.1016/0893-6080(89)90020-8. URL https://www.sciencedirect.com/science/article/pii/0893608089900208. (Cited on page 251.)

Yann LeCun, L. Bottou, and Y. Bengio. Reading checks with multilayer graph transformer networks. In *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 151–154 vol.1, 1997. doi: 10.1109/ICASSP.1997.599580. (Cited on page 254.)

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf. (Cited on page 254.)

Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv:1603.07285v2*, 2018. (Cited on pages 254 and 257.)

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. (Cited on page 261.)

Jean-Louis Basdevant and Jean Dalibard. *Quantum Mechanics*. 01 2005. doi: 10.1007/3-540-28805-8. (Cited on pages 299 and 311.)

Jean-Philippe Bouchaud, Gilles Montambaux, Antoine Georges, and Marc Mézard. *Physique statistique I*. Editions de l'Ecole Polytechnique, Palaiseau, 2016. (Cited on page 302.)