Development of a new OpenFOAM solver for plasma cutting modelling

Nicolas Godinaud, Pierre Boivin, Pierre Freton, Jean-Jacques Gonzalez, Frédéric Camy-Peyret

To cite this version:

Nicolas Godinaud, Pierre Boivin, Pierre Freton, Jean-Jacques Gonzalez, Frédéric Camy-Peyret. Development of a new OpenFOAM solver for plasma cutting modelling. Computers and Fluids, 2022, 10.1016/j.compfluid.2022.105479 . hal-03661919

Development of a new OpenFOAM solver for plasma cutting modelling

Nicolas Godinaud[a,b,c,*], Pierre Boivin[a], Pierre Freton[b], Jean-Jacques Gonzalez[b], Frédéric Camy-Peyret[c]

[a]*Aix Marseille Univ, CNRS, Centrale Marseille, M2P2, 13451 Marseille, France*
[b]*LAPLACE, Université de Toulouse, CNRS, INPT, UPS, Toulouse, France*
[c]*Akryvia, 1 rue de la Noë, 44321 Nantes, France*

Abstract

A new OpenFOAM solver is presented, for the simulation of plasma cutting torches. The mathematical model that is introduced is based on the compressible Navier-Stokes equations coupled via source terms to the electric current conservation equation. Due to the conservative and hyperbolic nature of the model, a Godunov-type scheme is used for the first time in the context of plasma cutting simulation. The numerical method consists of a second-order Total Variation Diminishing (TVD) integration with flux Harten-Lax-van Leer-Contact (HLLC) Riemann solver for the flow conservation equations, coupled with a Laplace solver for the current conservation equation. An efficient formulation for the equation of state, accurately taking into account the plasma properties, is also presented. The solver is validated through a set of canonical test cases (shock tubes and 2D Riemann problems) and it is used to simulate a three-dimensional plasma cutting torch. Good agreement is found with the literature, with an improvement in the ability to deal with the shocks occurring during plasma cutting.

Keywords: Plasma cutting, HLLC, Shocks, Jets

---

[*] Corresponding author
Email address: `ng@akryvia.com` (Nicolas Godinaud)

1. Introduction

Plasma cutting is an industrial process widely used for metal cutting. The principle of a plasma cutting torch is to establish an electric arc between the torch and a workpiece, producing a high temperature jet which impacts the piece and causes the metal of the plate to melt. Due to the strong velocity of this jet, the molten metal is pushed out of the impingement area and leaves a kerf in the plate. This process is usually cheaper than laser cutting, but it is less accurate. However, with the usage of new types of nozzles and other innovations in the design of plasma cutting torches, recent advances are constantly improving the efficiency and accuracy of plasma cutting. This is making the process more and more interesting in comparison with laser cutting.

Despite these new progresses, many physical phenomena related to plasma cutting still need to be investigated in order to better understand the process and design even more efficient torches. Since the first generation of plasma cutting torches dating from the late 1950's, many experimental studies have been carried out to analyse and improve this cutting process. Among these experimental works, several are based on spectroscopic and imaging determination of pressure, temperature, velocity and other quantities in the plasma column [1–8]. All these studies point out the presence of shock waves in the jets coming out of plasma cutting torches, showing their underexpanded aspect.

Along with these experimental investigations, several modellings of the process have been presented since the end of the 1990's [1, 4, 5, 9–17]. The first CFD study of a plasma cutting configuration has been proposed by González-Aguilar *et al* [10]. The method presented consists in solving the Navier-Stokes equations, coupled with the calculation of the electric and magnetic fields through joule heating and Lorentz force. For this purpose, a two-dimensional axisymmetric and steady calculation with a pressure-based solver using the SIMPLE algorithm [18] has been performed. The same numerical method has been used in the investigations of Freton *et al* [4, 5], completed by a turbulence model. The influence of the radiation models has also been studied, and comparisons between 2D-axisymmetric and 3D calculations have been done. The results obtained by these numerical studies have been validated with experimental data. Although there are various other modellings of plasma cutting available in the literature, none of them presents any new numerical method to solve the equations. All calculations that have been performed until now to simulate the plasma cutting process are pressure based and use SIMPLE-like algorithms. As for the experimental works, these numerical investigations agree on the underexpanded aspect of the high temperature jets with the observations of shock waves. In a realistic cutting configuration, the shock waves can have a significant influence on the quality of the cuts. Consequently, a thorough investigation of these shock waves is required in the context of improving plasma cutting.

For this study, we use OpenFOAM (OF) [19] which is an open source CFD code. The novel approach introduced in this paper has been performed with the development of a new three-dimensional transient solver. It is based on a Godunov-type numerical method with the HLLC approximate Riemann solver [20], particularly adapted in presence of shock waves. Several other extensions of the standard single-phase gas dynamic Godunov methods are found in the literature. For example, there are various cases in which these methods have been adapted for mixture two-phase flows and applied to the simulation of different phenomena, such as wave propagation [21], cavitation [22], as well as volcanic flows [23].

Thus, we propose for the first time to apply this extended solver to the simulation of plasma cutting. This choice can be explained by the conservative and hyperbolic nature of the presented model. Moreover, this kind of method is exactly designed to handle compressible flows with discontinuities, such as shock waves. The mathematical model that is used is close to those of previous studies. It solves the flow conservation equations, coupled with the calculation of the electric potential by solving a Laplace equation, corresponding to the current conservation equation. An efficient interpolation method for the approximation of the plasma thermodynamics properties is presented as well. The cell-centered numerical scheme is second-order TVD, with a MUSCL reconstruction and a second order Runge-Kutta time integration. The main purpose of this work is to validate the new OF solver on a plasma cutting configuration. Simultaneously, this study allows to demonstrate the ability of the Godunov-type method to simulate this kind of plasma flows and deal with strong shock waves more efficiently than the commonly used pressure-based methods. Thus, the calculations have been performed with both the present solver and the pressure-based solver used by Freton *et al* [4, 5].

First, the mathematical model that has been used for the simulation of a plasma cutting torch is introduced: the main assumptions leading to the set of equations are discussed and the methods used for the interpolations of the plasma thermodynamic properties are detailed and validated.

Secondly, the numerical method that has been employed and developed in OF is presented. After the introduction of a general three-dimensional finite volume discretization, the formulation of the HLLC flux that has been chosen to solve the hyperbolic part of the equations is given, followed by the MUSCL scheme used to reconstruct variables at cell faces in order to reach a second order vspace0.3cmtization. The procedure used to calculate the source terms is also detailed and the time integration scheme is presented.

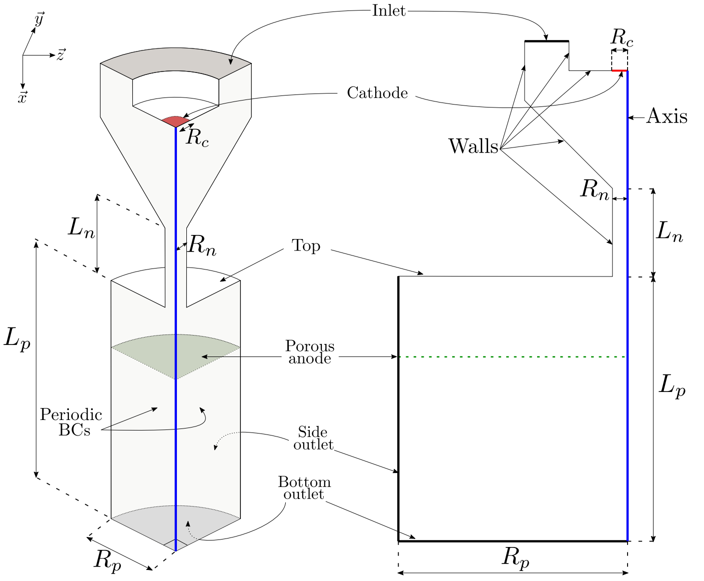
Then, validations of the HLLC scheme along with the MUSCL reconstruction are introduced with two common test cases: a one-dimensional Sod shock tube, and a two-dimensional Riemann problem.

Finally, the complete solver is used on a plasma cutting torch configuration. The results are compared with the ones obtained with another solver and they are validated.

2. Mathematical model

2.1. Problem description

The main configuration studied for the validation of the present solver is a simplified model of a plasma cutting torch. The geometry used is shown in Fig. 1.



Symbol	Length (mm)
L_n	3.00
R_n	0.50
L_p	20.0
R_p	8.00
R_c	0.44 (see Sec. 5.2)

Figure 1: Details of the geometry considered for the present work.

The gas considered is pure air, whose properties are discussed in Sec. 2.3. The fluid enters the plasma torch through the inlet with a high generating pressure of several atmospheres. Due to a current density imposed on the cathode of radius R_c, the gas is ionized and it becomes a plasma, conducting the current through an electric arc between the cathode and the anode. Because of the joule heating and the constriction of the plasma in the narrow nozzle of length L_n and radius R_n, the gas reaches a temperature of almost 30 000 K near the axis of the torch. Finally, the plasma is coming out of the nozzle in the ambient air, forming a high temperature underexpanded jet in a plenum of radius R_p and length L_p. In a realistic setup, a metal plate (workpiece), which is also the anode, is located a few millimeters downstream of the nozzle exit. Thus, it is impacted by the jet, whose temperature can reach several thousand kelvins. Combined with the high momentum of the jet, this results in the melting and the expulsion of the metal out of the piece, forming a kerf in the plate. In the present study, the interaction between the solid work piece and the jet is not taken into account. Instead, the metal plate is modeled by a porous anode [4], located 5 mm below the nozzle exit. This porous anode has no major impact on the flow, the jet can go through it without deflection, but the electric potential is fixed to 0 V downstream of it. It gives a reference value of the potential, representing the arc attachment location.

2.2. Governing equations

The plasma is considered as a compressible newtonian fluid and is assumed to be a thermal plasma at local thermodynamic equilibrium (LTE). The magnetic properties of the plasma are not considered, only the electric field is taken into account through the calculation of the electric potential, corresponding to the conservation of the current. The radiation of the plasma is modeled using a net emission coefficient assumption. The unstable areas corresponding to the arc attachment are not modeled. Since the solid workpiece is replaced by a porous anode in our model, there is no metal vapor to take into account.

The electric potential V is calculated by solving the following Laplace equation derived from the Maxwell-Gauss equation:

$$\boldsymbol{\nabla} \cdot (\sigma \boldsymbol{\nabla} V) = 0, \tag{1}$$

where σ is the electrical conductivity, depending on the local temperature and pressure.

The plasma flow is modeled by the compressible Navier-Stokes equations, coupled with the electric potential via source terms that take the joule heating and the radiation into account:

$$\frac{\partial \boldsymbol{U}}{\partial t}+\frac{\partial \boldsymbol{F}(\boldsymbol{U})}{\partial x}+\frac{\partial \boldsymbol{G}(\boldsymbol{U})}{\partial y}+\frac{\partial \boldsymbol{H}(\boldsymbol{U})}{\partial z}=\boldsymbol{S}(\boldsymbol{U}). \tag{2}$$

In this conservative formulation of the Navier-Stokes equation, $\boldsymbol{U}$ is the vector of conserved variables defined as

$$\boldsymbol{U}=\left[\rho,\ \rho u,\ \rho v,\ \rho w,\ \rho E\right]^{\top}, \tag{3}$$

where ρ is the density, u, v and w are the components of the velocity vector $\boldsymbol{u}$ and $E = e+(u^2+v^2+w^2)/2$ is the total energy by mass unit, sum of internal energy e and kinetic energy.

The flux vectors $\boldsymbol{F}(\boldsymbol{U})$, $\boldsymbol{G}(\boldsymbol{U})$ and $\boldsymbol{H}(\boldsymbol{U})$ are given by

$$\boldsymbol{F}(\boldsymbol{U})=\left[\rho u,\ \rho u^2+p,\ \rho uv,\ \rho uw,\ u(\rho E+p)\right]^{\top}, \tag{4}$$

$$\boldsymbol{G}(\boldsymbol{U})=\left[\rho v,\ \rho uv,\ \rho v^2+p,\ \rho vw,\ v(\rho E+p)\right]^{\top}, \tag{5}$$

$$\boldsymbol{H}(\boldsymbol{U})=\left[\rho w,\ \rho uw,\ \rho vw,\ \rho w^2+p,\ w(\rho E+p)\right]^{\top}, \tag{6}$$

and the source term vector $\boldsymbol{S}(\boldsymbol{U})$ reads

$$\boldsymbol{S}(\boldsymbol{U})=\left[0,\ (\boldsymbol{\nabla}\cdot\overline{\overline{\boldsymbol{\tau}}})\cdot\overrightarrow{e_x},\ (\boldsymbol{\nabla}\cdot\overline{\overline{\boldsymbol{\tau}}})\cdot\overrightarrow{e_y},\ (\boldsymbol{\nabla}\cdot\overline{\overline{\boldsymbol{\tau}}})\cdot\overrightarrow{e_z},\ \boldsymbol{\nabla}\cdot(\kappa\boldsymbol{\nabla}T)+\boldsymbol{\nabla}\cdot\left(\overline{\overline{\boldsymbol{\tau}}}\cdot\boldsymbol{u}\right)+\underbrace{\frac{||\boldsymbol{j}||^2}{\sigma}}_{\text{joule heating}}-\underbrace{4\pi\varepsilon_N}_{\text{radiative losses}}\right]^{\top}. \tag{7}$$

In the previous expressions, p and T are respectively the pressure and the temperature,

$\overline{\overline{\boldsymbol{\tau}}}$ is the viscous stress tensor, κ is the thermal conductivity, ε_N is the net emission coefficient and $\boldsymbol{j}$ is the current density vector. Under the Stokes hypothesis, the viscous stress tensor for a compressible fluid can be written as

$$\overline{\overline{\boldsymbol{\tau}}} = \mu \left(\boldsymbol{\nabla} \boldsymbol{u} + (\boldsymbol{\nabla} \boldsymbol{u})^\top \right) - \frac{2}{3} \mu \left(\boldsymbol{\nabla} \cdot \boldsymbol{u} \right) \overline{\overline{\boldsymbol{I}}}, \tag{8}$$

where μ is the dynamic viscosity.

Finally, the current density vector is calculated as

$$\boldsymbol{j} = -\sigma \boldsymbol{\nabla} V. \tag{9}$$

The formulation of the model (2) corresponds to the conservative form of the Euler equations with source terms. It is conservative, and its hyperbolic nature depends on the equation of state (EOS) discussed in the following section.

2.3. Closure models

For the system of equations to be fully defined, let us now detail

- the equation of state, providing the link between energy and density (related to the conserved variables) on one hand, and pressure and temperature on the other hand,
- the procedure to compute the speed of sound c according to pressure and temperature,
- the interpolations for electrical conductivity σ, the dynamic viscosity μ and the thermal conductivity κ

2.3.1. Equation of state

As the temperature can rise up to 30 000 K during the plasma cutting process, different reactions occurs, such as dissociation and ionisation. Consequently, the gas thermodynamic properties cannot be calculated with a simple equation of state. Due to this wide range of temperature, the transport coefficients are also pressure and temperature dependant.

Direct calculation of these thermodynamic properties and transport coefficient for each value of pressure and temperature is too expensive in term of computational time and is beyond the scope of the present work. Instead, two tables of calculated values have been used for each thermodynamic property and transport coefficient. One table corresponds to values calculated for temperatures going from 300 K to 30 000 K at a pressure $p_1 = 1$ bar. The second table gives the values at a pressure $p_2 = 8$ bars for the same range of temperature. These calculated values in the tables are obtained from a plasma composition and thermodynamic properties in-house software assuming LTE [24].

For the computation of the thermodynamic properties at pressure values different from p_1 and p_2, two types of interpolations had to be performed: one allowing the calculation of the density ρ and internal energy e from pressure p and temperature T, and a second one giving T and p from ρ and e.

In a hyperbolic code, the first interpolation $(p, T) \rightarrow (\rho, e)$ is only used to initialize the conservative variable vector $\boldsymbol{U}$ (3); whereas the second (reverse) interpolation $(\rho, e) \rightarrow (p, T)$ is required in the flux evaluation, at every grid point and time-step. The latter is therefore more critical in an efficient solver.

Density and internal energy interpolations from pressure and temperature

Since the numerical method used in this work is density-based, ρ and e have to be interpolated from p and T at the initial step. For this purpose, the calculated values in the tables of ρ and e at $p = p_1$ and $p = p_2$ have been used. We denote as $\rho_1(T) = \rho_1(p = p_1, T)$

and $\rho_2(T) = \rho_2(p = p_2, T)$ the tables of density at $p = p_1$ and $p = p_2$. In the same way, $e_1(T) = e_1(p = p_1, T)$ and $e_2(T) = e_2(p = p_2, T)$ are the tables of internal energy.

For the calculation of the density, we found that a linear interpolation method is sufficient:

$$\rho(p, T) = \left(1 - \frac{p - p_1}{p_2 - p_1}\right) \rho_1(T) + \left(\frac{p - p_1}{p_2 - p_1}\right) \rho_2(T). \tag{10}$$

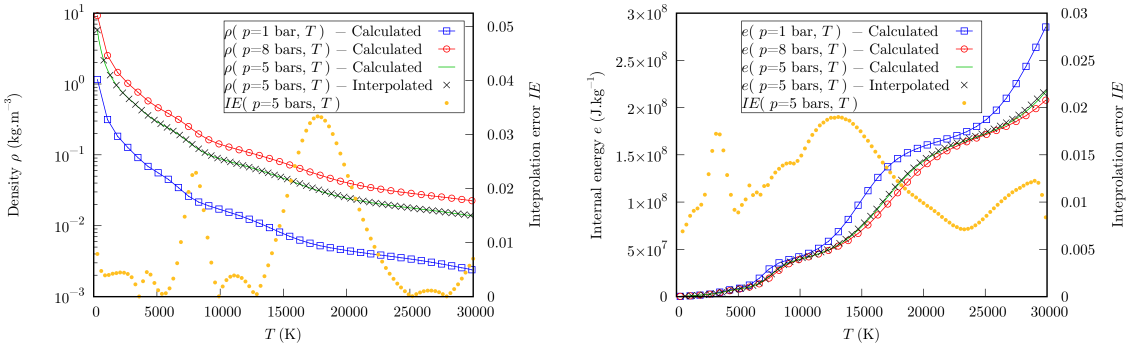
Regarding the interpolation of the internal energy, a quadratic method is performed

$$e(p, T) = [1 - \xi(p)]\ e_1(T) + \xi(p)\ e_2(T), \tag{11}$$

with
$$\xi(p) = \begin{cases} \dfrac{|p - p_1|}{p - p_1} \sqrt{\dfrac{|p - p_1|}{p_2 - p_1}} & \text{if } p \neq p_1, \\ 0 & \text{if } p = p_1. \end{cases}$$

Figure 2 shows values of e and ρ interpolated at $p = 5$ bars for different temperature values between 300 K and 30 000 K. Note that the choice of a quadratic interpolation is not random. The peaks in the physical properties – see, e.g. the heat capacity, corresponding to the slope of e in Fig. 2 or thermal conductivity, Fig. 4 – are associated with dissociation reactions (e.g. $\mathrm{O_2 + M \longrightarrow 2\,O + M}$) which typically have square pressure dependencies.

In order to assess the accuracy of these interpolated values, they are compared with the values calculated in a third table, at $p = 5$ bars. Relative errors between the interpolated and calculated values are also plotted. For both variables ρ and e, the interpolation method yields excellent accuracy: interpolation errors IE between the interpolated and calculated values reaches a maximum of 0.03 for the density and a maximum of 0.02 for the internal energy.



(a) Density (b) Internal energy

Figure 2: Interpolations of internal energy e and density ρ at $p = 5$ bars - Error IE between interpolated and calculated values.

Pressure and temperature interpolations from density and internal energy

As outlined above, this interpolation is critical, as it is performed at every grid point and time-step. The easiest and most efficient way that has been found to determine the values of p and T according to ρ and e is achieved by using the linear interpolation of density

$$p(\rho, e) = \left(1 - \frac{\rho - \rho_1(T_1(e))}{\rho_2(T_2(e)) - \rho_1(T_1(e))}\right) p_1 + \left(\frac{\rho - \rho_1(T_1(e))}{\rho_2(T_2(e)) - \rho_1(T_1(e))}\right) p_2, \tag{12}$$

where T_1 and T_2 are the inverse of the internal energy tables e_1 and e_2: $T_1(e) = e_1^{-1}(T)$ and $T_2(e) = e_2^{-1}(T)$. Once the value of the pressure is computed according to ρ and e, the value of the temperature can easily be calculated using the expression

$$T(p, e) = [1 - \xi(p)]\ T_1(e) + \xi(p)\ T_2(e). \tag{13}$$

In order to assess the efficiency of this reverse interpolation, a test has been conducted for several initial values of the pressure-temperature couple: $(p,\ T)$. For each value of $(p,\ T)$, the direct interpolation method has been performed, giving a corresponding couple

(ρ, e). Then, using this interpolated couple (ρ, e), the reverse interpolation has been used to determine a final value of the pressure-temperature couple: (p_f, T_f). By comparing the final couples (p_f, T_f) with the initial ones (p, T), the following errors can be defined:

$$E_p(p,T) = \frac{|p_f - p|}{p} \quad \text{and} \quad E_T(p,T) = \frac{|T_f - T|}{T}. \tag{14}$$

The contour lines of the pressure error $E_p(p,T)$ are shown in Fig. 3. A maximum error of 0.026 is reached for a pressure of 3 bars and a temperature of 18 000 K. The same behaviour could be observed for the temperature error $E_T(p,T)$, with a maximum error below 0.016. Thus, the reverse interpolation is an accurate method that allows a low-cost computation of the pressure and temperature according to the density and internal energy.

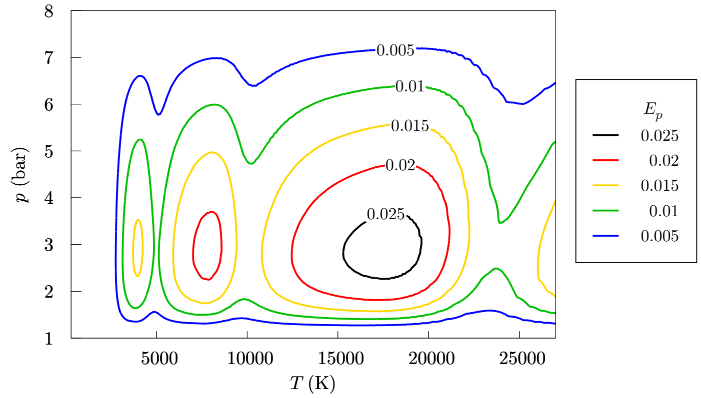


Figure 3: Contour lines for the pressure error $E_p(T,p)$.

2.3.2. Speed of sound calculation

The calculation of the speed of sound c values is required for the evaluation of the numerical fluxes (see Sec. 3.2). In order to compute the values of c according to pressure and temperature, the following relation has been used:

$$c(p,T) = \sqrt{\frac{\gamma(p,T)\ p}{\rho}}, \tag{15}$$

with γ corresponding to the pressure and temperature dependant heat capacity ratio, which can be defined as

$$\gamma(p,T) = \frac{c_p(p,T)}{c_v(p,T)}, \tag{16}$$

where c_p and c_v are respectively the specific heat capacities at constant pressure and constant volume. Regarding c_p, its values have been calculated according to temperature at $p = p_1$ and $p = p_2$:

$$\begin{cases} c_p(p = p_1, T) = c_{p,1}(T) = \left(\dfrac{\partial h}{\partial T}\right)_{p=p_1} = \dfrac{\mathrm{d}h_1(T)}{\mathrm{d}T}, \\ c_p(p = p_2, T) = c_{p,2}(T) = \left(\dfrac{\partial h}{\partial T}\right)_{p=p_2} = \dfrac{\mathrm{d}h_2(T)}{\mathrm{d}T}. \end{cases} \tag{17}$$

In previous relations, $h_1(T)$ and $h_2(T)$ are the enthalpy tables at $p = p_1$ and $p = p_2$, which are calculated from the internal energy tables and density tables as follow:

$$\begin{cases} h_1(T) = e_1(T) + \dfrac{p_1}{\rho_1(T)}, \\ h_2(T) = e_2(T) + \dfrac{p_2}{\rho_2(T)}. \end{cases} \tag{18}$$

For pressure values different from p_1 and p_2, the heat capacity c_p is interpolated linearly between the two tables.

Concerning the calculation of the heat capacity c_v, it is done using Mayer's relation

$$c_v(p,T) = R_s(p,T) - c_p(p,T), \tag{19}$$

with $R_s(p,T) = p/(\rho T)$ denoting the specific gas constant. For the present equation of state, unlike the ideal gas case, the value of R_s is not constant and depends on pressure an temperature.

Using Eq. (19) to express c_v, the heat capacity ratio can finally be written as

$$\gamma(p,T) = \left(1 - \frac{R_s(p,T)}{c_p(p,T)}\right)^{-1}, \tag{20}$$

leading to the following expression of the speed of sound:

$$c(p,T) = \sqrt{\frac{p}{\rho\left(1 - \frac{R_s(p,T)}{c_p(p,T)}\right)}}. \tag{21}$$

According to this last formulation of c, it is always positive and defined if $R_s < c_p$. This condition is always true in the range of pressure and temperature considered in this study. Consequently, as demonstrate in [25], since the speed of sound values calculated according to the present EOS are strictly positive, they are physical and the set of equations of the model (2) is hyperbolic.

To summarize, assuming that the density ρ, the pressure p and the temperature T are known, the speed of sound c is determined in two steps:

(1) linear interpolation of c_p between the two tables at $p = p_1$ and $p = p_2$,

(2) application of Eq. (21).

To give an idea of the range of values in which the speed of sound lies, it has been plotted in Fig. A.13 in the appendices, for two different pressure values: $p = 1$ bar and $p = 8$ bars.

2.3.3. Transport coefficients

The transport coefficients σ, κ and μ have been linearly interpolated between tabulated values, also calculated at $p_1 = 1$ bar and $p_2 = 8$ bars. These transport coefficients are obtained from the LTE plasma composition and the Chapman-Enskog method [24]. They take into account the reactions inside the plasma, like dissociation or ionisation. An example is given in Fig. 4 for the thermal conductivity of air at $p = 1$ bar and $p = 8$ bars. Some peaks can be seen, corresponding to the dissociation of nitrogen and dioxygen and to their successive ionizations. The influence of the pressure is mainly significant for temperatures above 15000 K. The temperature and pressure dependency of the viscosity μ and electrical conductivity σ of air are shown in the appendices, in Fig. B.14.

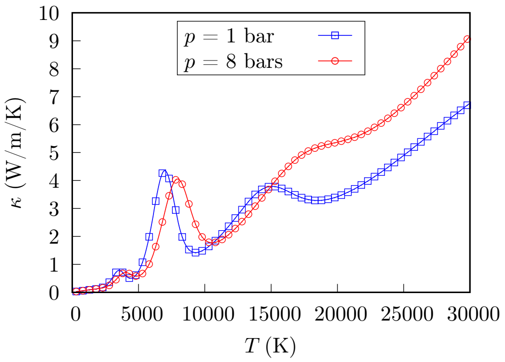


Figure 4: Calculated values of the thermal conductivity κ of air, at $p = 1$ bar and $p = 8$ bars.

3. Numerical method

3.1. Spatial discretization

In this work, a finite volume cell-centered method has been used, with a reconstruction of conserved variables at the centers of cells faces (see Sec. 3.3). The 3D computational domain has been discretized with an unstructured mesh, made of N_C different tetrahedral or hexahedral elements. An example of two neighbouring tetrahedral cells is shown in Fig. 5.

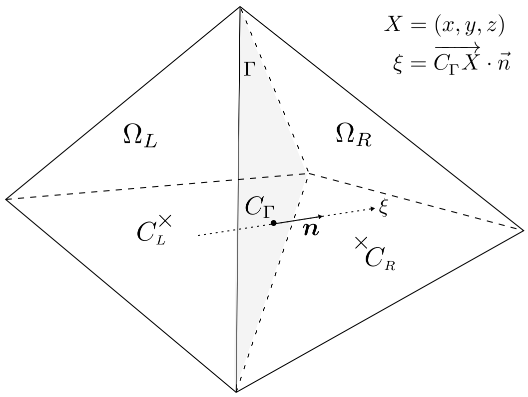


Figure 5: Example of two neighbouring tetrahedral cells Ω_L and Ω_R. C_L and C_R are the centroids of the cells, Γ denotes the face between Ω_L and Ω_R, C_Γ is the center of Γ and $\boldsymbol{n}$ is the unit vector normal to Γ.

In this section, we introduce the numerical scheme that has been used for the approximation of the cell-centered values $\boldsymbol{U}_i(t)$ $(i = 1, ..., N_C)$:

$$\boldsymbol{U}_i(t) \equiv \frac{1}{|\Omega_i|} \iiint\limits_{\Omega_i} \boldsymbol{U}(x, y, z, t)\, \mathrm{d}\Omega. \tag{22}$$

By integrating Eq. (2) over the volume of any cell Ω_i, the following equation is obtained:

$$\frac{\mathrm{d}\boldsymbol{U}_i}{\mathrm{d}t} + \frac{1}{|\Omega_i|} \oiint\limits_{\partial\Omega_i} \overline{\overline{\boldsymbol{\mathcal{W}}}}(\boldsymbol{U}_i) \cdot \boldsymbol{n}_i \, \mathrm{d}\Gamma = \boldsymbol{S}_i, \tag{23}$$

where $\overline{\overline{\boldsymbol{\mathcal{W}}}}(\boldsymbol{U}_i) = (\boldsymbol{F}(\boldsymbol{U}_i),\ \boldsymbol{G}(\boldsymbol{U}_i),\ \boldsymbol{H}(\boldsymbol{U}_i))$ is the tensor of fluxes and $\boldsymbol{n}_i$ is the outward unit vector normal to the boundary of Ω_i. Then, the surface integral over the boundary may be separated in the sum of surface integrals over the N_{f_i} faces Γ_k $(k = 1, ..., N_{f_i})$ of the cell Ω_i:

$$\frac{\mathrm{d}\boldsymbol{U}_i}{\mathrm{d}t} + \frac{1}{|\Omega_i|}\sum_{k=1}^{N_{f_i}} \iint\limits_{\Gamma_k} \overline{\overline{\boldsymbol{\mathcal{W}}}}(\boldsymbol{U}_i)\cdot\boldsymbol{n}_k \,\mathrm{d}\Gamma = \boldsymbol{S}_i, \tag{24}$$

with $\boldsymbol{n}_k = (n_k^x,\ n_k^y,\ n_k^z)^\top$ corresponding to the outward unit vector normal to the face Γ_k. By defining $q_k = \boldsymbol{u}_i \cdot \boldsymbol{n}_k$ the projection of the velocity on the normal vector, the projection $\hat{\mathsf{F}}_k$ of the tensor of fluxes on the normal vector can be expressed as:

$$\hat{\mathsf{F}}_k = \overline{\overline{\boldsymbol{\mathcal{W}}}}(\boldsymbol{U}_i)\cdot\boldsymbol{n}_k = [\rho_i q_k,\ \rho_i u_i q_k + p_i\ n_k^x,\ \rho_i v_i q_k + p_i\ n_k^y,\ \rho w_i q_k + p_i\ n_k^z,\ q_k(\rho_i E_i + p_i)]^\top \tag{25}$$

Introducing the numerical flux F_k as:

$$\mathsf{F}_k = \iint\limits_{\Gamma_k} \hat{\mathsf{F}}_k \,\mathrm{d}\Gamma, \tag{26}$$

Equation (24) can be simplified as

$$\frac{\mathrm{d}\boldsymbol{U}_i}{\mathrm{d}t} + \frac{1}{|\Omega_i|}\sum_{k=1}^{N_{f_i}} \mathsf{F}_k = \boldsymbol{S}_i. \tag{27}$$

Once a numerical scheme is chosen to calculate the approximation $\hat{\boldsymbol{\mathcal{F}}}_k$ of the flux value $\hat{\mathsf{F}}_k$ on the face Γ_k (see Sec. 3.2), the numerical flux can be written as

$$\mathsf{F}_k = |\Gamma_k|\hat{\boldsymbol{\mathcal{F}}}_k. \tag{28}$$

Thus, the following semi-discrete scheme is obtained:

$$\frac{\mathrm{d}\boldsymbol{U}_i}{\mathrm{d}t} = -\frac{1}{|\Omega_i|}\sum_{k=1}^{N_{f_i}} |\Gamma_k|\hat{\boldsymbol{\mathcal{F}}}_k + \boldsymbol{S}_i. \tag{29}$$

Regarding the method used for the computation of the approximation $\hat{\boldsymbol{\mathcal{F}}}_k$, due to the conservative and hyperbolic nature of the model, a HLLC scheme (see, e.g. [20]) has been chosen and is introduced in the next section.

3.2. Numerical flux: the HLLC scheme

In order to approximate the values of $\hat{\mathbf{F}}$ on the faces of the cells, a Riemann problem has to be solved for each face:

$$\begin{cases} \dfrac{\partial \boldsymbol{U}}{\partial t} + \dfrac{\partial \hat{\mathbf{F}}}{\partial \xi} = 0, \\ \boldsymbol{U}(\xi, 0) = \begin{cases} \boldsymbol{U}_L, & \text{if } \xi < 0, \\ \boldsymbol{U}_R, & \text{if } \xi > 0, \end{cases} \end{cases} \tag{30}$$

where $\boldsymbol{U}_R$ and $\boldsymbol{U}_L$ are the values reconstructed on the right and left side of the face, as explained in next section. To solve these Riemann problems and compute the values of $\hat{\mathbf{F}}$ on the face ($\xi = 0$), the approximate Riemann solver HLLC [20] has been chosen. To calculate the HLLC fluxes, a three-wave structure is assumed. The wave speeds of the left, middle and right waves are respectively denoted as S_L, S_* and S_R, giving the following expression of the HLLC flux:

$$\hat{\boldsymbol{\mathcal{F}}}^{HLLC} = \begin{cases} \hat{\mathbf{F}}_L, & \text{if } \ 0 \leq S_L, \\ \hat{\mathbf{F}}_L^* = \hat{\mathbf{F}}_L + S_L \ (\hat{\boldsymbol{U}}_L^* - \boldsymbol{U}_L), & \text{if } \ S_L \leq 0 \leq S^*, \\ \hat{\mathbf{F}}_R^* = \hat{\mathbf{F}}_R + S_R \ (\hat{\boldsymbol{U}}_R^* - \boldsymbol{U}_R), & \text{if } \ S^* \leq 0 \leq S_R, \\ \hat{\mathbf{F}}_R, & \text{if } \ S_R \leq 0, \end{cases} \tag{31}$$

with the intermediate states $\hat{\boldsymbol{U}}_K^*$ ($K = L$ or $K = R$) given by

$$\hat{\boldsymbol{U}}_K^* = \frac{S_K - q_K}{S_K - S^*} \begin{pmatrix} \rho_K \\ \rho_K \; u_K + \dfrac{p^* - p_K}{S_K - q_K} \; n_x \\ \rho_K \; v_K + \dfrac{p^* - p_K}{S_K - q_K} \; n_y \\ \rho_K \; w_K + \dfrac{p^* - p_K}{S_K - q_K} \; n_z \\ \rho_K \; E_k + \dfrac{p^* \; S^* - p_K \; q_K}{S_K - q_K} \end{pmatrix}. \tag{32}$$

The value of the pressure p^* is the same in both intermediate states, as it is constant through the contact discontinuity

$$p^* = p_L + \rho_L(S_L - q_L)(S^* - q_L) = p_R + \rho_R(S_R - q_R)(S^* - q_R). \tag{33}$$

Regarding the wave speed S^* of the middle wave, it can be expressed as

$$S^* = \frac{p_R - p_L + \rho_L \; q_L \; (S_L - q_L) - \rho_R \; q_R \; (S_R - q_R)}{\rho_L(S_L - q_L) - \rho_R(S_R - q_R)}. \tag{34}$$

These previous expressions of variables in the intermediate states are obtained using algebraic manipulations from the Rankine–Hugoniot jump conditions across the waves of speeds S_L and S_R:

$$\begin{cases} \hat{\mathbf{F}}_L^* - \hat{\mathbf{F}}_L = S_L \; (\hat{\boldsymbol{U}}_L^* - \boldsymbol{U}_L), \\ \hat{\mathbf{F}}_R^* - \hat{\mathbf{F}}_R = S_R \; (\hat{\boldsymbol{U}}_R^* - \boldsymbol{U}_R). \end{cases} \tag{35}$$

More details about the derivation of the intermediate states are given in [20].

Regarding the left and right wave speeds S_L and S_R, they have been estimated as proposed by Davis [26]:

$$S_L = \min(q_L - c_L, q_R - c_R) \quad \text{and} \quad S_R = \max(q_L + c_L, q_R + c_R), \tag{36}$$

where c_L and c_R are the left and right speed of sound values, computed as described in Sec. 2.3.2.

3.3. Reconstruction with a flux limiter

In order to achieve a better precision, a second order MUSCL-type scheme [27] has been used. For one cell Ω_K, the idea is to reconstruct the conserved variables $\boldsymbol{U}_K$ at the centers C_{Γ_k} of its faces. The index k takes values from 1 to N_{f_K}, where N_{f_K} denotes the total number of faces of the cell Ω_K. Thus, for a face Γ_k, the expression of the reconstructed value is:

$$\boldsymbol{U}_K(C_{\Gamma_k}) = \boldsymbol{U}_K(C_K) + \Theta_K \, \overrightarrow{C_K C_{\Gamma_k}} \cdot \boldsymbol{\nabla} \boldsymbol{U}(C_K). \tag{37}$$

In the previous equation, $\Theta_K \in [0, 1]$ is a flux limiter, which can be calculated as

$$\Theta_K = \min_{k \in [\![1, N_{f_K}]\!]} (\theta_{\Gamma_k}). \tag{38}$$

To prevent any overshoot in the solution, a Minmod limiter [28] has been chosen for the calculation of θ_{Γ_k}:

$$\theta_{\Gamma_k} = \max\left[0,\ \min\left(1,\ \phi_{\Gamma_k}\right)\right], \tag{39}$$

with:

$$\phi_{\Gamma_k} = \begin{cases} \dfrac{\boldsymbol{U}^{max} - \boldsymbol{U}_K}{2\delta_k}, & \text{if } \ \delta_k > 0, \\ \dfrac{\boldsymbol{U}^{min} - \boldsymbol{U}_K}{2\delta_k}, & \text{if } \ \delta_k < 0, \\ 1, & \text{if } \ \delta_k = 0, \end{cases} \tag{40}$$

where $\delta_k = \overrightarrow{C_K C_{\Gamma_k}} \cdot \boldsymbol{\nabla U}(C_K)$, and the variables $\boldsymbol{U}^{max}$ and $\boldsymbol{U}^{min}$ are respectively the maximum and minimum values between cell Ω_K and its direct neighbors. The calculation of the gradient at the cell center $\boldsymbol{\nabla U}(C_K)$, is performed using a Least-Squares scheme [29].

3.4. Source terms calculation

The components of the source term vector $\boldsymbol{S(U)}$ defined in Eq. (7) are computed explicitly. To calculate the current density vector $\boldsymbol{j}$, the field of electric potential V must be determined first. This is performed at each time step by solving the Laplace Eq. (1), using the Laplacian discretization scheme of OF. The numerical scheme that has been chosen for this purpose is called Gauss Harmonic Corrected, using the Gauss theorem to integrate the Laplacian. During the discretization procedure, the transport coefficient σ is interpolated at the centers of the faces using an harmonic mean. In addition, a correction is performed for non-orthogonal faces. Regarding the calculations of the gradients, it is done with a weighted least squares method, as described in [29]. There are two main advantages of using this method: it is made to handle strong conductivity jumps and it does not require any correction for non-orthogonal faces. Finally, the stress tensor's divergence $\boldsymbol{\nabla} \cdot \overline{\overline{\boldsymbol{\tau}}}$ is computed using the Gauss Linear scheme of OF [30]. With this scheme, the Gauss theorem is used to integrate the divergence terms and the values of the fields are interpolated at the centers of the faces thanks to a central differencing.

3.5. Time integration and general procedure

The time integration is performed using a second-order TVD Runge-Kutta scheme. The timestep value Δt is calculated according to

$$\Delta t = CFL \min_{i \in [\![1, N_C]\!]} \left(\frac{d_i}{||\boldsymbol{u}_i||^2 + c_i} \right), \tag{41}$$

where $0 < CFL < 1$ is the Courant–Friedrichs–Lewy number and d_i is the shortest distance between the center of the cell Ω_i and the center of its faces:

$$d_i = \min_{k \in [\![1, N_{f_i}]\!]} ||\overrightarrow{C_i C_{\Gamma_k}}||. \tag{42}$$

Finally, the general procedure that is used at each time step of the calculation is the following:

1. Reconstruction of conserved variables at faces centers: the gradients of conserved variables are computed and Eq. (37) is applied.

2. Computation of HLLC fluxes through each face of the domain: the speed of sound values are calculated according to pressure and temperature (Eq. (21)) and Eq. (31) is used.

3. Source terms calculation (see Sec. 3.4): solving of the Laplace equation Eq. (1) to determine the electric potential field V along with the current density $\boldsymbol{j}$, and calculation of the divergence of the viscous stress tensor.

4. Time integration: the conserved variables are updated from time $t = t^n$ to time $t = t^{n+1} = t^n + \Delta t$. This step allows the determination of ρ^{n+1} and e^{n+1}, corresponding respectively to the approximation of the density field and internal energy fields at

time t^{n+1}. At this point of the procedure, the pressure p^{n+1} and temperature T^{n+1} are not computed.

5. Reverse interpolation: calculation of p^{n+1} and T^{n+1} according to ρ^{n+1} and e^{n+1} using the interpolations (12) and (13).

For the initialization, the temperature and pressure are imposed in the domain. Thus, the interpolations of density (10) and internal energy (11) according to pressure and temperature are performed to compute the density and internal energy at the initial time.

4. Validations of the HLLC and MUSCL schemes

In order to check that the methods implemented in OF are efficient for the capture of shocks and discontinuities, two test cases have been performed. The first one is the Sod shock tube test case and the second one is a 2D Riemann problem. For both test cases, only the Euler equations are solved, corresponding to Eq. (2) with the components of the source term $\boldsymbol{S}(\boldsymbol{U})$ set to 0. Moreover, an ideal gas law is considered for the equation of state, with a heat capacity ratio γ set to 1.4.

4.1. Sod shock tube

For this test case, a computational domain $\Omega = [0,1] \times [-0.005, 0.005] \times [-0.005, 0.005]$ has been considered and discretized with a cartesian grid of size 200 × 1 × 1. The initial state at $t = 0$ s is the following:

$$[\rho,\ u,\ v,\ w,\ p] = \begin{cases} [1,\ 0,\ 0,\ 0,\ 1] & \text{if}\ \ 0 \leq x \leq 0.5, \\ [0.125,\ 0,\ 0,\ 0,\ 0.1] & \text{if}\ \ 0.5 \leq x \leq 1. \end{cases} \tag{43}$$

Figure 6 shows the results obtained at $t = 0.25$ s with a CFL number of 0.9. The numerical solution without reconstruction (first order HLLC) and the one with a MUSCL reconstruction (second order) are compared with the analytical solution. Both methods are efficient to resolve the discontinuities. As expected, a better accuracy is achieved by the second order scheme, especially in the contact discontinuity area.

This case has also been performed with the second order scheme for four different grid resolutions: 100, 200, 400 and 800 cells. The numerical results computed at $t = 0.25$ s with a CFL number of 0.9 on these 4 grids are compared with the analytical solution in Fig. 7. The plotted profiles indicate that the numerical results converge to the analytical solution when the grid is refined and the numerical scheme seems to achieve grid independence for this test case.

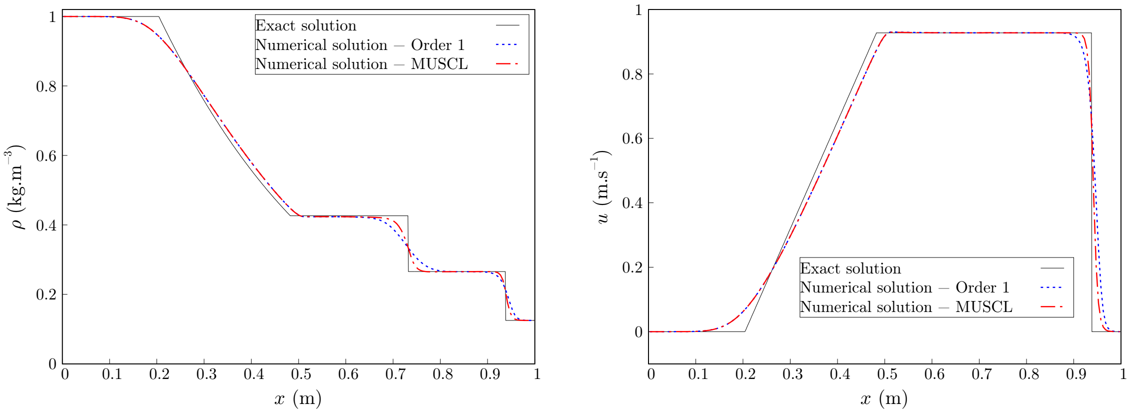
4.2. 2D Riemann problem

To validate our implementation, a 2D Riemann problem test case has also been carried out. A cartesian grid of size $400 \times 400 \times 1$ has been used with a computational domain $\Omega = [0,1] \times [0,1] \times [0,0.01]$. At $t = 0$ s, the initial state is divided as follow:

$$[\rho,\ u,\ v,\ w,\ p] = \begin{cases} [1,\ -0.75,\ 0.5,\ 0,\ 1] & \text{if}\ \ 0 \leq x \leq 0.5\ \ \text{and}\ \ 0 \leq y \leq 0.5, \\ [3,\ -0.75,\ -0.5,\ 0,\ 1] & \text{if}\ \ 0.5 \leq x \leq 1\ \ \text{and}\ \ 0 \leq y \leq 0.5, \\ [2,\ 0.75,\ 0.5,\ 0,\ 1] & \text{if}\ \ 0 \leq x \leq 0.5\ \ \text{and}\ \ 0.5 \leq y \leq 1, \\ [1,\ 0.75,\ -0.5,\ 0,\ 1] & \text{if}\ \ 0.5 \leq x \leq 1\ \ \text{and}\ \ 0.5 \leq y \leq 1. \end{cases} \tag{44}$$

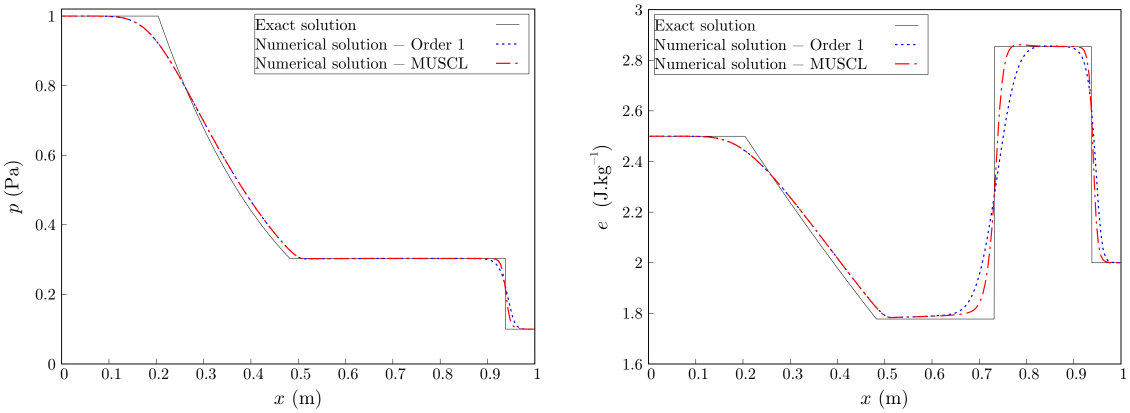
Although there is no analytical solution for this test case, many references are available in the literature. The results obtained with the present scheme and a CFL number of 0.9 are compared with the ones of [31] in Fig. 8, which shows the contour lines of the density field at $t = 0.3$ s.

These results indicate that the method used in this work is able to capture contact



(a) Density ρ

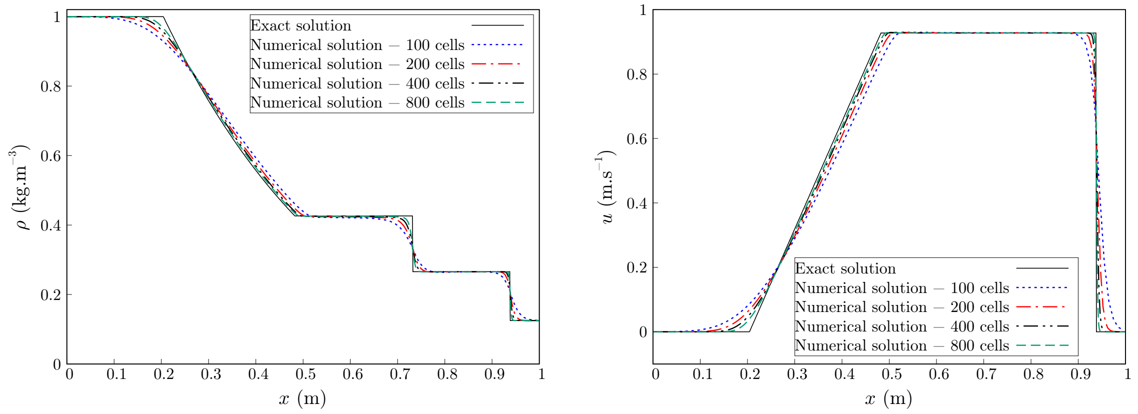
(b) Velocity u



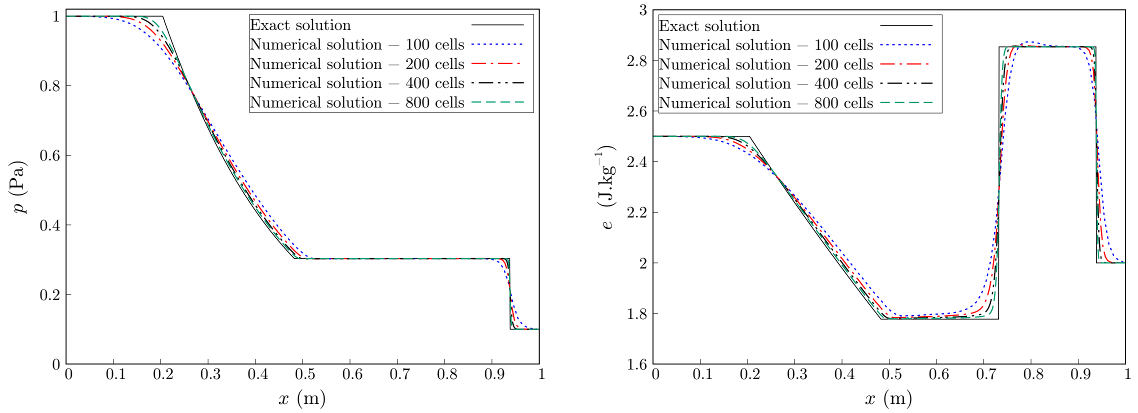
(c) Pressure p

(d) Internal energy e

Figure 6: Profiles along the x-axis for the Sod shock tube test case at $t = 0.25$ s ($CFL = 0.9$). Comparison between the exact solution, the numerical solution without reconstruction (order 1) and the numerical solution with the MUSCL reconstruction.

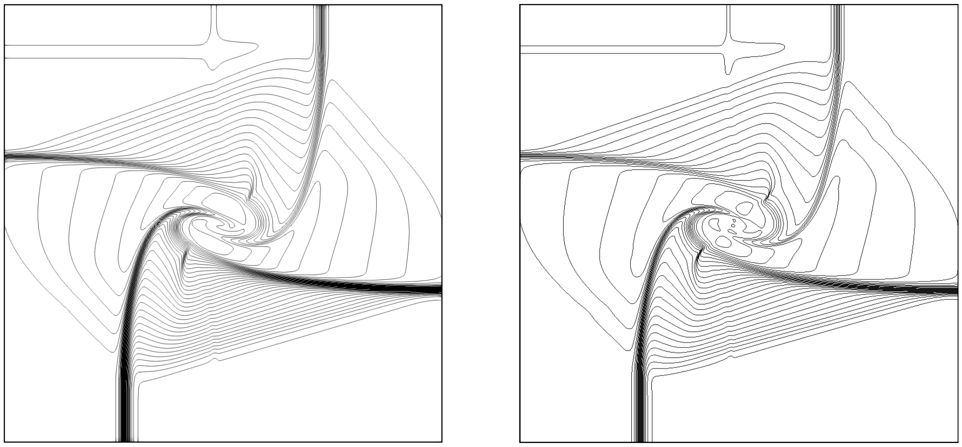


(a) Density ρ (b) Velocity u



(c) Pressure p (d) Internal energy e

Figure 7: Profiles along the x-axis for the Sod shock tube test case at $t = 0.25$ s ($CFL = 0.9$). Comparison between the exact solution and numerical solutions (order 2) with several mesh resolutions.



(a) HLLC with MUSCL reconstruction

(b) Reference [31]

Figure 8: Density contour for the 2D Riemann problem at $t = 0.3$ s ($CFL = 0.9$).

discontinuities for two-dimensional cases. It has also been checked with other 2D Riemann problems that rarefaction and shock waves are well captured with this method.

5. Validation of the coupling with the electric potential - Simulation of a plasma cutting torch

5.1. Comparison with a pressure-based method

In order to validate the numerical method presented in this work, it has been compared with a pressure-based method used on the same configuration. The geometry used for this calculation has been presented in Fig. 1. The pressure-based method uses the SIMPLE algorithm [18] of ANSYS Fluent combined with User Defined Functions. It has been assessed and validated in various studies [4, 5]. The calculation with the current method (OF) has been performed in 3D for a quarter of the torch, whereas a 2D-axisymmetric calculation has been done with the pressure-based method (Fluent).

5.2. Boundary conditions

Boundary conditions details are given in Table 1. The cathode is considered as a slipping wall, on which a parabolic current density profile $J_c(r)$ is given:

$$J_c(r) = J_{\max}\left(1 - \frac{r^2}{R_c^2}\right), \tag{45}$$

where $r = \sqrt{y^2 + z^2}$ is the radial distance from the axis and $J_{\max} = 1.6 \times 10^8$ A.m^{-2} is the maximum current density. This value of $J_{\max}$ has been chosen according to an experimental study [32]. The value of the cathode radius R_c has been calculated so that a 50 A corresponding current I_c is imposed on the cathode:

$$I_c = 2\pi \int_0^{R_c} J_c(r)\; r\,\mathrm{d}r = 50 \text{ A}. \tag{46}$$

Boundaries	Variables			
	p	T	$\boldsymbol{u}$	V
Inlet	$p_{in} = 5$ bars	300 K	–	$\boldsymbol{\nabla} V = 0$
Cathode	$\boldsymbol{\nabla} p = 0$	$\boldsymbol{\nabla} T = 0$	Slip	$-(\sigma \boldsymbol{\nabla} V) \cdot \boldsymbol{n} = J_c(r)$
Walls	$\boldsymbol{\nabla} p = 0$	$\boldsymbol{\nabla} T = 0$	Slip	$\boldsymbol{\nabla} V = 0$
Top	$\boldsymbol{\nabla} p = 0$	300 K	Slip	$\boldsymbol{\nabla} V = 0$
Side outlet	1 bar	300 K	–	$\boldsymbol{\nabla} V = 0$
Bottom outlet	$\boldsymbol{\nabla} p = 0$	$\boldsymbol{\nabla} T = 0$	$\boldsymbol{\nabla} \boldsymbol{u} = 0$	$V = 0$

Table 1: Boundary conditions for the simulation of the plasma cutting torch.

5.3. Grid independence study with the OpenFOAM solver

Prior to the comparison with Fluent, the present OF solver has been used to simulate the plasma cutting configuration with three different levels of axial grid refinement (in the x-axis direction) in the vicinity of the shock wave. The coarsest grid has a cell size

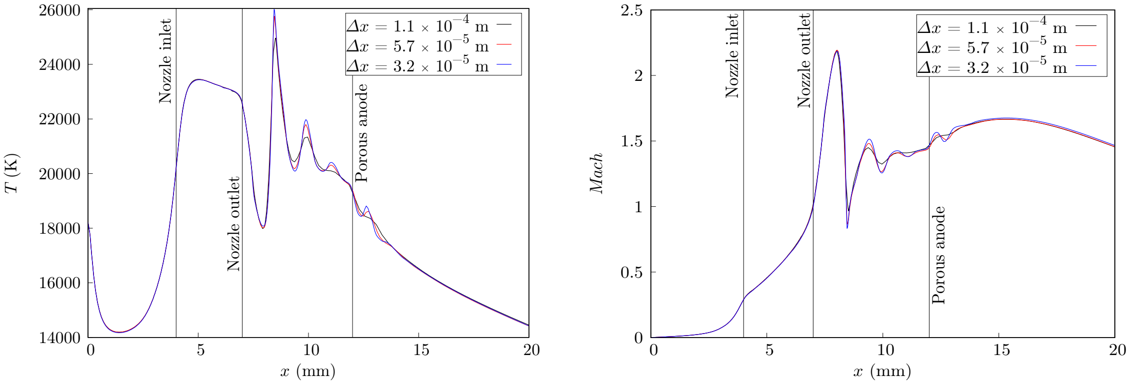
$\Delta x = 1.1 \times 10^{-4}$ m in the axial direction in the shock area, which is located a few millimeters downstream of the nozzle outlet. For the most refined grid, $\Delta x = 3.2 \times 10^{-5}$ m and for the middle level of refinement, $\Delta x = 5.7 \times 10^{-5}$ m. The three calculations have been performed with a CFL number of 0.9. The profiles computed on the axe ($r = 0$) for the three grids are compared in Fig. 9. Other profiles and fields are presented in the following section dedicated to the comparison between OF and Fluent. The results plotted in Fig. 9 show that the shock wave is well captured with the three meshes, even with the coarsest one. Although the amplitude of the waves is slightly damped in the case of the coarsest grid, the results obtained with second level of refinement and the maximum level of refinement are close enough to consider that the second level is sufficient in this case. Consequently, the second level of refinement ($\Delta x = 5.7 \times 10^{-5}$ m) is the one that has been used for the comparison between OF and Fluent results.

5.4. Results comparison

The computation with OF using the present method has been performed on a mesh made of 4×10^5 hexahedral cells, whereas 2×10^4 quadrilateral cells have been used for the 2D-axisymmetric calculation with Fluent. Both meshes had the same refinements in the axial (see previous section) and radial directions. The transient simulation with OF has been run with a CFL number of 0.9 until stabilization of the mass flow rates through the inlet and the nozzle, at $t = 3$ ms. Regarding the calculation with Fluent, it has been done with a steady solver until convergence.

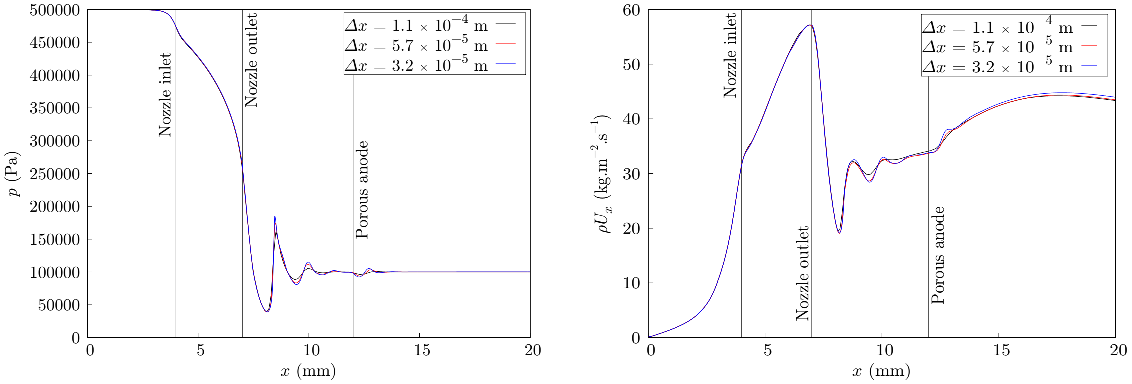
5.4.1. Fields analysis

Both methods give the same results in terms of mass flow rate and voltage: an inlet mass flow rate of 1.47×10^{-4} kg.s^{-1} is obtained, along with a voltage of 132 V between the cathode and the porous anode. As the arc attachment and the sheath are not described,



(a) Temperature T

(b) Mach number M



(c) Pressure p

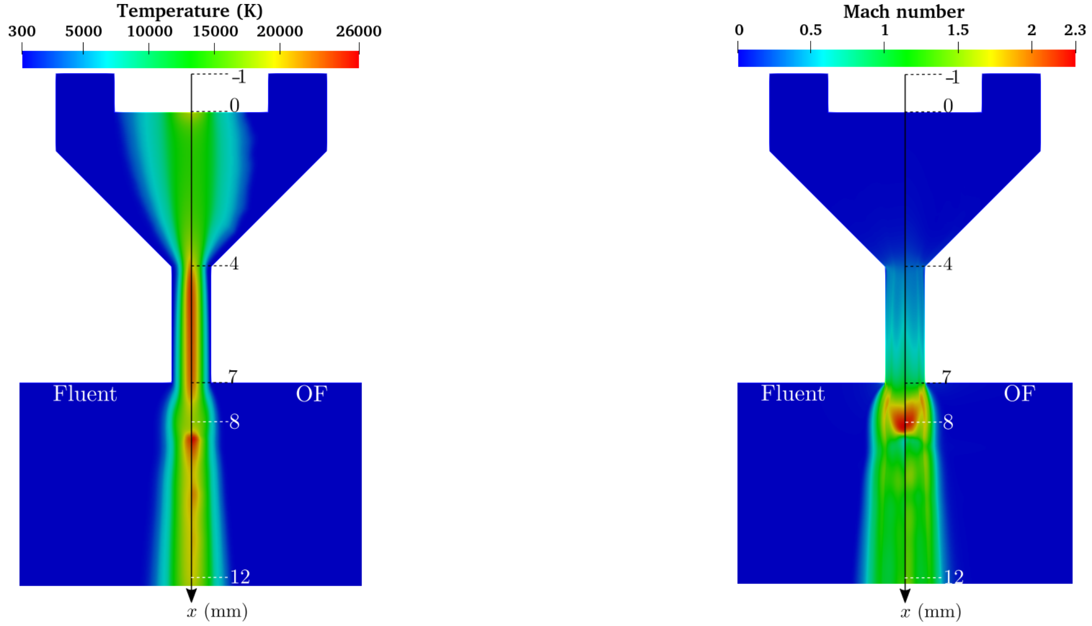
(d) Axial momentum ρu

Figure 9: Profiles along the x-axis - Comparison between the different levels of grid refinement.

this voltage only corresponds to the plasma column. Figure 10 compares different fields computed with OF and Fluent: the temperature T, mach number M, electric potential V and axial current density j_x fields are shown. Since the calculation with OF is 3D, a slice in the middle of the domain at $z = 0$ mm is plotted. Although only one slice is shown here, it has been checked that the results were isotropic by plotting other slices for different values of z. For the four fields, a good agreement is observed overall between OF and Fluent results. The main difference appears in the top part of torch, were the plasma is more constricted towards the center of the cathode in the OF results. The behaviour is opposite at the inlet of the nozzle ($x = 4$ cm), were Fluent results show a more constricted plasma, with higher temperature and current density than the ones calculated with OF. This can also be observed on the axial profiles shown in Fig. 11. The reasons of these differences have not been investigated precisely in the present work, since this area was not the main focus of the study. Farther downstream, as depicted in Fig. 10(b), the mach number reaches a maximum of 2.3 in the core of the underexpanded jet, upstream of a shock wave. Downstream of it, the temperature computed with OF rises up to 26 000 K, whereas it increases up to 24 000 K regarding the Fluent results. This difference of magnitude downstream of the shock wave is studied more thoroughly by analysing the axial profiles.

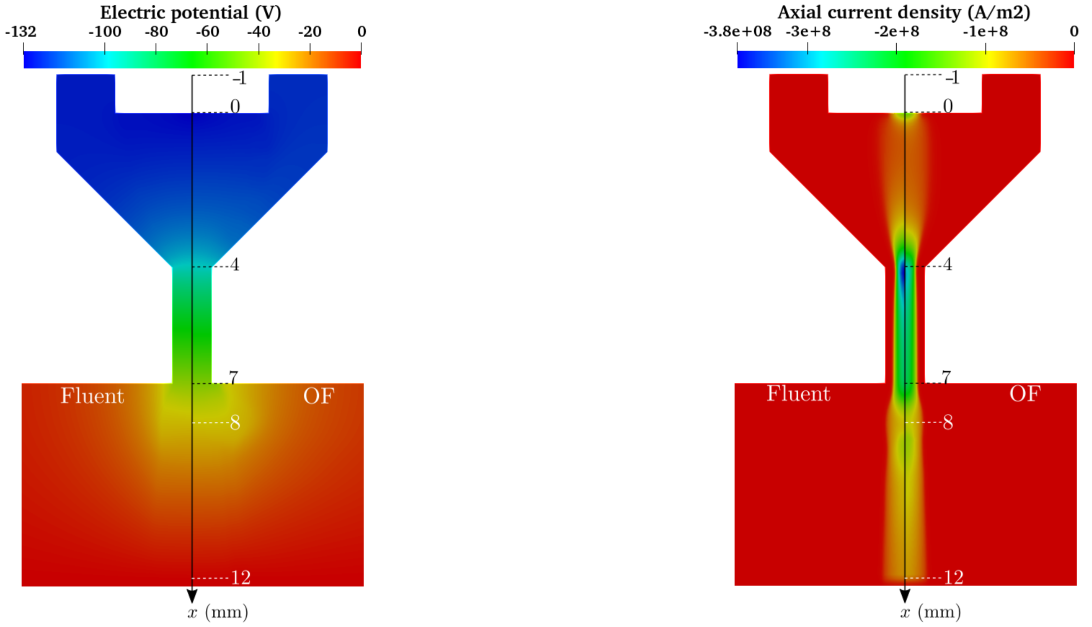
5.4.2. Axial profiles analysis

Profiles along the axis ($r = 0$) of T, M, p and x-momentum ρu are plotted in Fig. 11. From the cathode to the shock ($0 < x < 8$ mm), the profiles obtained with OF and Fluent are close, except for the temperature at the nozzle inlet, as mentioned previously. After the shock ($x > 8$ mm), the values of temperature, mach number and axial momentum computed with Fluent are much lower than the values calculated with OF. As a small discontinuity is observed after the shock on the ρu profile obtained with Fluent, the OF results seem to be more reliable. Moreover, the numerical method used with OF is suited for the capture



(a) Temperature field T

(b) Mach number field M



(c) Electric potential field V

(d) Axial current density field j_x

Figure 10: Comparison of fields computed with OF and Fluent - $p_{in} = 5$ bars.

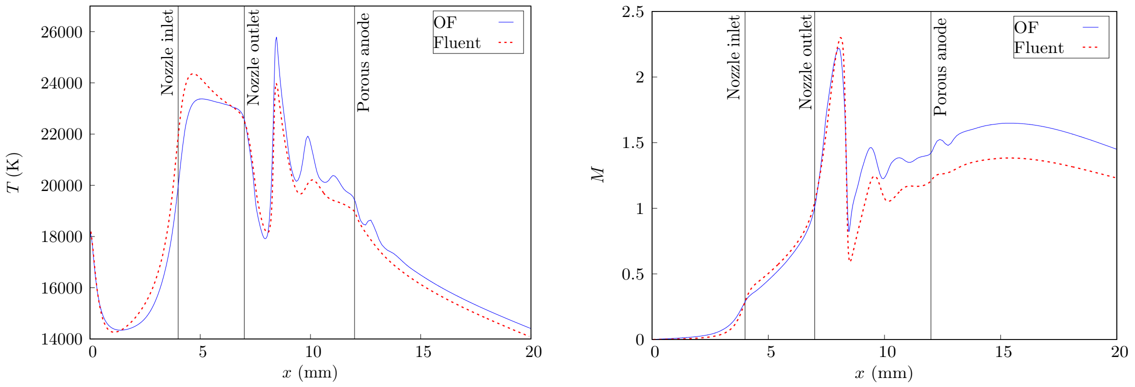
of shocks, as shown in the previous section.

In order to determine more carefully which of the two methods gives the best results downstream of the shock, more calculations have been performed and the results have been compared with the ones of other CFD softwares. These simulations have been carried out on the same geometry as the one shown in Fig. 1, without any coupling with the electric potential and without the diffusive source terms ($\boldsymbol{S(U)} = \vec{0}$). Thus, only the Euler equations have been solved and an ideal gas law with $\gamma = 1.4$ has been used for the equation of state. Four different solvers have been compared: the present solver (HLLC with MUSCL) in OF, the pressure-based method of Fluent (SIMPLE algorithm), the density-based method of Fluent (Roe scheme) and the HLLC scheme of SU2 [33]. Several calculations have been done with different values of the pressure imposed at the inlet p_{in}. For $p_{in} = 4$ bars, the results obtained with the four solvers are in good agreement with a maximal mach number of 2.0. With $p_{in} = 5$ bars ($\max(M) = 2.5$) and $p_{in} = 6$ bars ($\max(M) = 3.0$), all solvers give close results, except the pressure-based solver of Fluent. In fact, the higher p_{in}, the more Fluent's pressure-based solver results deviate from the others. This deviation only appears downstream of the shock, with still a good match with the other solvers upstream.

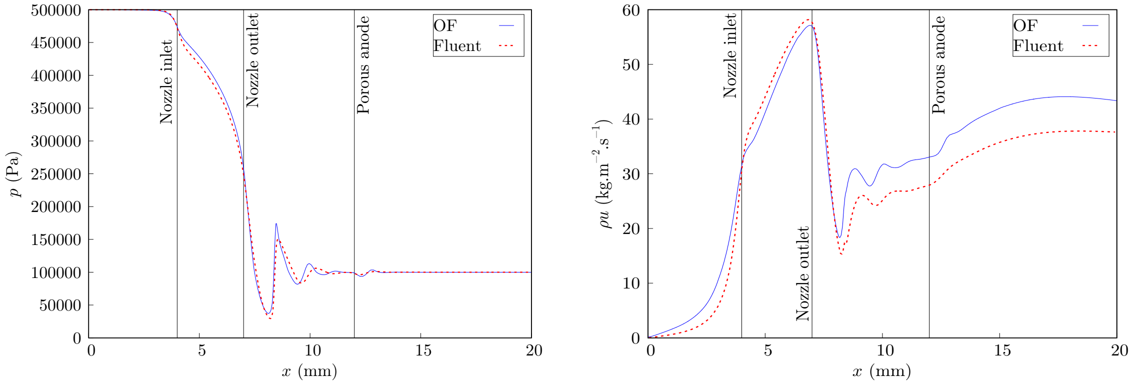
This complementary study leads to the conclusion that OF results shown in Fig. 11 are truly better downstream of the shocks than those of Fluent. In addition, it shows that the pressure-based solver of Fluent does not seem to be well suited for the capture of shock waves for mach number above 2.

5.4.3. Nozzle radial profiles analysis

Figure 12 shows the radial profiles of T and ρu computed with OF and Fluent in the middle section of the nozzle ($x = 5.5$ mm). From the nozzle center to half its radius ($0 < r < 0.25$ mm), OF and Fluent give very close results for both temperature and axial momentum. In the second half of the nozzle ($0.25 < r < 0.5$ mm), close to the wall,



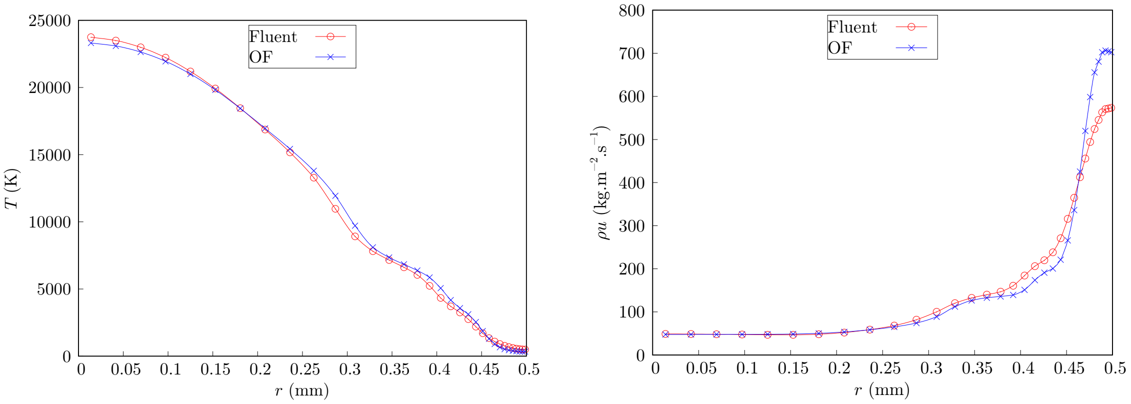
(a) Temperature T (b) Mach number M



(c) Pressure p (d) Axial momentum ρu

Figure 11: Profiles along the x-axis - Comparison between OF (present method) and Fluent (pressure-based) results - $p_{in} = 5$ bars.

more differences appear in the profiles. The values of temperature computed with OF in the vicinity of the wall are lower than the ones calculated with Fluent, leading to higher values of momentum. This might be due to a distinction in the treatment of the boundary condition, and also to the difference of constriction near the inlet of the nozzle, as discussed previously. The thickness of this “cold” gas layer is influenced by several parameters, such as the pressure inlet and the intensity of the current imposed on the cathode. It could be the focus of future investigations.



(a) Temperature T

(b) Axial momentum ρu

Figure 12: Radial profiles in the middle section of the nozzle ($x = 5.5$ mm) - Comparison between OF (present method) and Fluent (pressure-based) results - $p_{in} = 5$ bars.

6. Conclusion

A new OpenFOAM solver for the simulation of plasma cutting torches has been presented. It includes a new efficient formulation for the equilibrium plasma equation of state. Through introduction of a 2^{nd} order TVD scheme with HLLC Riemann solvers, the numerical method is able to accurately solve discontinuities, as those present in plasma cutting torches. This introduction of a Godunov-type scheme is novel in the scope of plasma cutting simulation. The choice of this numerical method has been justified by the conservative and hyperbolic nature of the mathematical model.

Following the presentation of the model and the corresponding EOS, the numerical scheme is detailed. The new OF solver is then validated through a one-dimensional shock tube and a two-dimensional Riemann problem. Finally, a three-dimensional simulation of a plasma cutting torch is presented, showing (i) results consistent with the literature [4, 5] and (ii) a much higher robustness to increasing the generating pressure.

Future works include the introduction of magnetic field equation, and a thorough investigation of the high temperature jet downstream of the nozzle.

Acknowledgements

This work is supported by Programme d'investissement d'avenir, operated by Bpifrance. The National Research Agency (ANRT) is acknowledged for funding the PhD of Nicolas Godinaud at the Laplace and M2P2 labs. Centre de Calcul Intensif d'Aix-Marseille is acknowledged for granting access to its high performance computing resources. Xi Deng is also gratefully acknowledged for fruitful discussions regarding numerical schemes.

References

[1] S. Ramakrishnan, M. W. Rogozinski, Properties of electric arc plasma for metal cutting, J. Phys. D: Appl. Phys. 30 (4) (1997) 636–644. `doi:10.1088/0022-3727/30/4/019`.

[2] S. Ramakrishnan, M. Gershenzon, F. Polivka, T. Kearney, M. Rogozinski, Plasma generation for the plasma cutting process, IEEE Trans. Plasma Sci. 25 (5) (1997) 937–946. `doi:10.1109/27.649600`.

[3] C. Pardo, J. González-Aguilar, A. Rodríguez-Yunta, M. A. G. Calderón, Spectroscopic

analysis of an air plasma cutting torch, J. Phys. D: Appl. Phys. 32 (17) (1999) 2181–2189. `doi:10.1088/0022-3727/32/17/308`.

[4] P. Freton, J. J. Gonzalez, A. Gleizes, F. C. Peyret, G. Caillibotte, M. Delzenne, Numerical and experimental study of a plasma cutting torch, J. Phys. D: Appl. Phys. 35 (2) (2001) 115–131. `doi:10.1088/0022-3727/35/2/304`.

[5] P. Freton, J. J. Gonzalez, F. C. Peyret, A. Gleizes, Complementary experimental and theoretical approaches to the determination of the plasma characteristics in a cutting plasma torch, J. Phys. D: Appl. Phys. 36 (11) (2003) 1269–1283. `doi:10.1088/0022-3727/36/11/307`.

[6] L. Girard, P. Teulet, M. Razafinimanana, A. Gleizes, F. Camy-Peyret, E. Baillot, F. Richard, Experimental study of an oxygen plasma cutting torch: I. Spectroscopic analysis of the plasma jet, J. Phys. D: Appl. Phys. 39 (8) (2006) 1543–1556. `doi:10.1088/0022-3727/39/8/014`.

[7] V. Sember, A. Mašláni, P. Křenek, M. Heinrich, R. Nimmervoll, H. Pauser, M. Hrabovský, Spectroscopic Characterization of a Steam Arc Cutting Torch, Plasma Chem. Plasma Process. 31 (5) (2011) 755–770. `doi:10.1007/s11090-011-9312-5`.

[8] J. Peters, J. Heberlein, J. Lindsay, Spectroscopic diagnostics in a highly constricted oxygen arc, J. Phys. D: Appl. Phys. 40 (13) (2007) 3960–3971. `doi:10.1088/0022-3727/40/13/008`.

[9] V. A. Nemchinsky, Plasma flow in a nozzle during plasma arc cutting, J. Phys. D: Appl. Phys. 31 (21) (1998) 3102–3107. `doi:10.1088/0022-3727/31/21/016`.

[10] J. González-Aguilar, C. Sanjurjo, A. Rodriguez-Yunta, M. Calderon, A theoretical

study of a cutting air plasma torch, IEEE Trans. Plasma Sci. 27 (1) (1999) 264–271. `doi:10.1109/27.763132`.

[11] S. Ghorui, J. V. R. Heberlein, E. Pfender, Non-equilibrium modelling of an oxygen-plasma cutting torch, J. Phys. D: Appl. Phys. 40 (7) (2007) 1966–1976.

[12] V. Colombo, A. Concetti, E. Ghedini, S. Dallavalle, M. Vancini, Understanding Plasma Fluid Dynamics Inside Plasma Torches Through Advanced Modeling, IEEE Trans. Plasma Sci. 36 (2) (2008) 389–402. `doi:10.1109/TPS.2008.918664`.

[13] Q. Zhou, H. Li, X. Xu, F. Liu, S. Guo, X. Chang, W. Guo, P. Xu, Comparative study of turbulence models on highly constricted plasma cutting arc, J. Phys. D: Appl. Phys. 42 (1) (2008) 015210. `doi:10.1088/0022-3727/42/1/015210`.

[14] Q. Zhou, H. Li, F. Liu, S. Guo, W. Guo, P. Xu, Effects of Nozzle Length and Process Parameters on Highly Constricted Oxygen Plasma Cutting Arc, Plasma Chem. Plasma Process. 28 (6) (2008) 729–747. `doi:10.1007/s11090-008-9154-y`.

[15] Q. Zhou, H. Yin, H. Li, X. Xu, F. Liu, S. Guo, X. Chang, W. Guo, P. Xu, The effect of plasma-gas swirl flow on a highly constricted plasma cutting arc, J. Phys. D: Appl. Phys. 42 (9) (2009) 095208. `doi:10.1088/0022-3727/42/9/095208`.

[16] S. Guo, Q. Zhou, W. Guo, P. Xu, Computational Analysis of a Double Nozzle Structure Plasma Cutting Torch, Plasma Chem. Plasma Process. 30 (1) (2010) 121–140. `doi:10.1007/s11090-009-9204-0`.

[17] A. B. Murphy, H. Park, Modeling of Thermal Plasma Processes: The Importance of Two-Way Plasma-Surface Interactions, Plasma Process. Polym. 14 (1-2) (2017) 1600177. `doi:10.1002/ppap.201600177`.

[18] S. V. Patankar, Numerical Heat Transfer and Fluid Flow (1st ed.), CRC Press, 1980.

[19] H. G. Weller, G. Tabor, H. Jasak, C. Fureby, A tensorial approach to computational continuum mechanics using object-oriented techniques, Comput. Phys. 12 (6) (1998) 620–631. `doi:10.1063/1.168744`.

[20] E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics - A Pratical Introduction, 3rd Edition, Springer, Berlin, Heidelberg, 2009.

[21] D. Zeidan, E. Romenski, A. Slaouti, E. F. Toro, Numerical study of wave propagation in compressible two-phase flow, Int. J. Numer. Methods Fluids 54 (4) (2007) 393–417. `doi:10.1002/fld.1404`.

[22] R. Saurel, P. Boivin, O. Le Métayer, A general formulation for cavitating, boiling and evaporating flows, Comput. Fluids 128 (2016) 53–64. `doi:10.1016/j.compfluid.2016.01.004`.

[23] D. Zeidan, Assessment of mixture two-phase flow equations for volcanic flows using Godunov-type methods, Appl. Math. Comput. 272 (2016) 707–719. `doi:10.1016/j.amc.2015.09.038`.

[24] A. Harry Solo, M. Benmouffok, P. Freton, J. J. Gonzalez, Stochiometry Air - CH4 Mixture: Composition, Thermodynamic Propertiess and Transport Coefficients, Plasma Phys. Technol. 7 (1) (2020) 21–29. `doi:10.14311/ppt.2020.1.21`.

[25] A. Harten, P. D. Lax, C. D. Levermore, W. J. Morokoff, Convex entropies and hyperbolicity for general euler equations, SIAM J. Numer. Anal. 35 (6) (1998) 2117–2127.

[26] S. F. Davis, Simplified Second-Order Godunov-Type Methods, SIAM J. Sci. Statist. Comput. 9 (3) (1988) 445–473. `doi:10.1137/0909030`.

[27] A. Chiapolino, R. Saurel, B. Nkonga, Sharpening diffuse interfaces with compressible

fluids on unstructured meshes, J. Comput. Phys. 340 (2017) 389–417. `doi:10.1016/j.jcp.2017.03.042`.

[28] P. L. Roe, Some contributions to the modeling of discontinuous flows, Lect. Appl. Math. 22 (1985) 163–192.

[29] M. Trautmann, E. Spaniol, M. Hertel, U. Füssel, Modifications to the gradient schemes on unstructured cell centered grids for the accurate determination of gradients near conductivity changes, Phys. Fluids 31 (4) (2019) 047104. `doi:10.1063/1.5089734`.

[30] OpenFOAM v9 User Guide (Mar. 2017).
URL `https://cfd.direct/openfoam/user-guide/`

[31] X. Deng, P. Boivin, F. Xiao, A new formulation for two-wave Riemann solver accurate at contact interfaces, Phys. Fluids 31 (4) (2019) 046102. `doi:10.1063/1.5083888`.

[32] F. Yin, J. Schein, C. Hackett, J. Heberlein, Investigation of the cathode behavior in a plasma cutting torch, in: ISPC-14 Symp. Proc., Vol. 1, Prague, Czech Republic, 1999, pp. 49–54.

[33] T. D. Economon, F. Palacios, S. R. Copeland, T. W. Lukaczyk, J. J. Alonso, SU2: An Open-Source Suite for Multiphysics Simulation and Design, AIAA J. 54 (3) (2016) 828–846. `doi:10.2514/1.J053813`.

Appendix A. Speed of sound

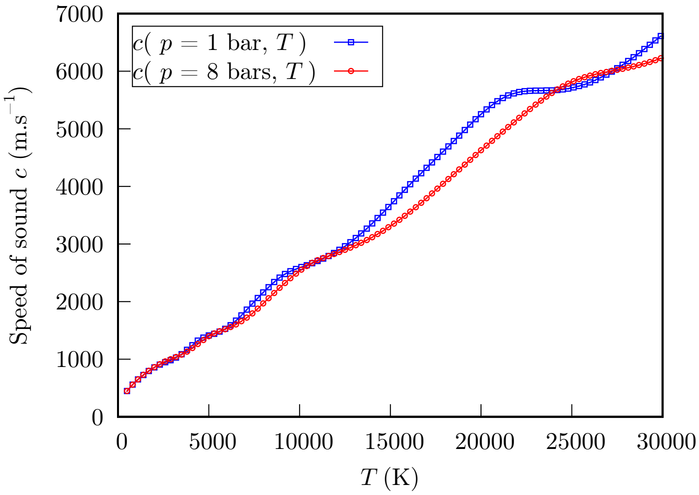
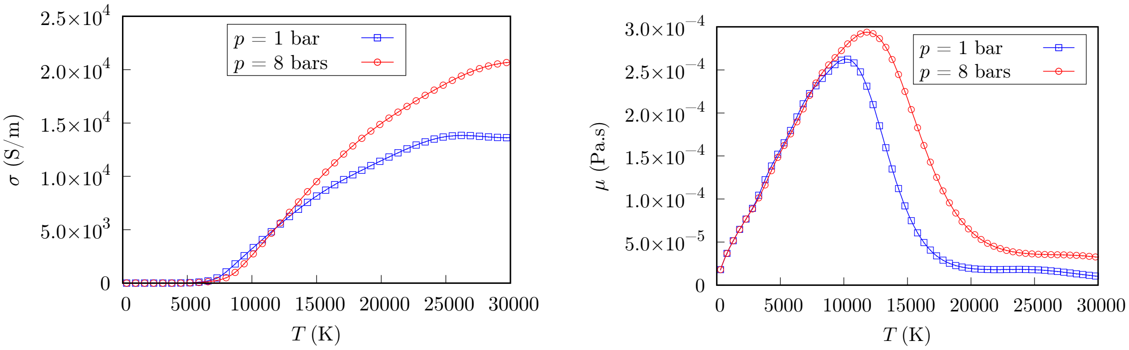


Figure A.13: Speed of sound c values at $p = 1$ bar and $p = 8$ bars.

Appendix B. Transport coefficients



(a) Electrical conductivity

(b) Dynamic viscosity

Figure B.14: Calculated values of the electrical conductivity σ and dynamic viscosity μ of air, at $p = 1$ bar and $p = 8$ bars.