Thermodynamic Cycle Analysis of a Rotating Detonation Engine for Stationary Applications

Master's Thesis

Riccardo Faggion

KIT: 2540221

PoliTo: s317427

At the Department of Mechanical Engineering (KIT)
Institute for Thermal Energy Technology and Safety (ITES)
and Politecnico di Torino (PoliTo)

Reviewer:	Prof. Daniel Banuti
	Prof. Andreas Class
Supervisor:	Prof. Daniel Banuti
	Prof. Luciano Rolando

31/01/2025

Masterarbeit

Aufgabenstellung

für

xxx

(Matr.-Nr.: 2540221)

Acknowledgement

xxx

Abstract

In this thesis, possible implementations of a rotating detonation engine (RDE) for energy generation are investigated and compared. In rotating detonation engines, unlike conventional systems, detonation of reactants is substituted to deflagration combustion, leading to an increase in the thermodynamic efficiency due to the higher temperature and pressure of the chemical reaction products.

This thesis will concentrate on stationary RDE application for electricity generation, in contrast to most of the other studies on this topic, which primarily address RDE application for propulsion systems. Specifically, two rotating detonation combustor models will be presented, and different power plant configurations and thermodynamic cycles will be investigated and compared.

The analyses are carried out in a Python environment, where the Cantera library is used for the evaluation of gas properties and the SDToolbox library is used for solving the detonation equations and determining the post-detonation conditions.

Because of its well-known detonability and ability to operate the plant without emitting CO2, hydrogen is the fuel of choice. The obtained values of efficiency are compared with those of a Joule-Brayton cycle under different operating conditions, in order to understand under which circumstances the usage of a rotating detonation combustor can present an advantage with respect to a standard deflagration combustor.

Contents

List of Figures

List of Tables

List of Abbreviations

CFD	Computational Fluid Dynamic
CJ	Chapman-Jouguet
PDE	Pulse Detonation Engine
ZND	Zeldovich, Von Neumann and Doering
RDE	Rotating Detonation Engine

1. Introduction

1.1 Motivation

Global warming is one of the main problems in today's world. According to the report of the World Meteorological World Meteorological Organization, 2024, carbon-dioxide concentration in the atmosphere has never been so high. This is leading to catastrophic consequences, such as the increase of Earth's surface temperature due to the greenhouse effect and always more extreme weather conditions. In 2022, the electricity and heat generation were responsible for nearly 44 % of global CO2 emissions from fuel combustion (IEA, 2023), therefore, it is clear that an energy transition in which fossil fuel combustion is replaced by renewable sources is of fundamental importance. On the other hand, due to their volatility, renewable energy production alone is not sufficient to guarantee an affordable and continuous current injection to the electric grid, and for that reason it is necessary to couple it with energy storage technologies and with power plants able to operate whenever the grid conditions require it. The advantage of a rotating detonation engine (RDE) power plant is that it can be seen both as a base-load power plant as well as a way to use stored energy. Like all other fuel-based engines, it can be run whenever the grid needs it, and it can also operate continuously, ensuring affordable energy generation. At the same time, one possible way to store energy is by using excess electricity produced by renewable sources to run an electrolysis process, in which hydrogen is extracted from water. Hydrogen produced in this way is called green hydrogen, and can be used as a fuel for running an RDE power plant. Another great advantage of an RDE power plant would be the thermodynamic cycle efficiency, which is expected to be higher than that of a standard Joule-Brayton cycle. This increase in efficiency is due to the use of detonation instead of a classic deflagration combustion, which presents higher outlet pressure and temperature of the combustion products.

1.2 Fundamentals of Detonation

In this section, the theoretical background necessary to understand a planar ideal detonation process is presented. All the reported information, as well as other insights, can be found in Browne et al., 2023 and Kuo, 1986.

1.2.1 Detonation theory

According to Zeldovich, Von Neumann and Doering theory (ZND theory) , detonation can be modeled as a shock wave followed by a chemical reaction. The description of detonation theory starts by assuming a planar detonation wave front. In Figure 1.1, in the laboratory reference frame (lab frame), a shock wave is traveling with a speed U_s from the region 2 to the region 1. In the case of a detonation wave, 2 would be the region of products and 1 the one of reactants. The flow in region 1 is moving at a speed u_1, while the speed of the flow in region 2 is u_2.

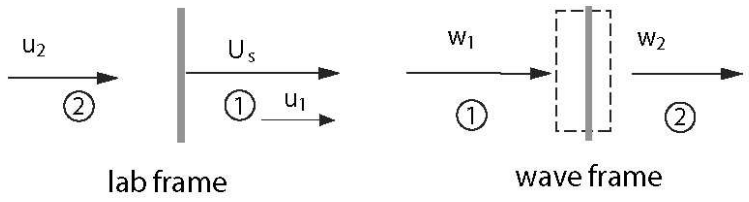

Figure 1.1: Schematic representation of laboratory and wave-fixed reference frames (Browne et al., 2023)

The same situation can be represented in a reference system that is moving at the wave speed U_s in the same direction as the shock wave. In this new reference system, the shock wave results in being steady, while the flow in zone 1 is traveling through the shock front at a speed equal to w_1. At the same time, the flow in the zone behind the shock is moving away from the shock front at a speed w_2. In particular, it is possible to apply the following transformation to pass from the laboratory reference frame to the wave-fixed reference frame:

$$w_1 = U_s - u_1 \tag{1.1}$$

$$w_2 = U_s - u_2 \tag{1.2}$$

The following analyses are all carried out in the shock wave reference frame. The continuity equation can be written as:

$$\rho_1 w_1 = \rho_2 w_2 \tag{1.3}$$

Where ρ_1 is the density before the shock, and ρ_2 the one after the shock.

The momentum equation is:

$$P_1 + \rho_1 {w_1}^2 = P_2 + \rho_2 {w_2}^2 \tag{1.4}$$

Where P_1 is the pressure before the shock and P_2 the one after.

Finally, the energy conservation equation is:

$$h_1 + \frac{{w_1}^2}{2} = h_2 + \frac{{w_2}^2}{2} \tag{1.5}$$

Where h_1 and h_2 are the specific enthalpies of the flow before and after the shock, respectively. In the case of a detonation wave, the enthalpies must account both for the sensible enthalpy and for the enthalpy of formation.

Finally, also an entropy condition needs to be fulfilled:

$$s_2 \geq s_1 \tag{1.6}$$

With s_1 and s_2 being respectively the specific entropy before and after the shock.

1.2.2 Rayleigh line

Combining the continuity and momentum equations, the following expression is obtained:

$$P_2 = P_1 - {\rho_1}^2 {w_1}^2 (v_2 - v_1) \tag{1.7}$$

Where v_1 and v_2 are the specific volumes of the flow before and after the shock, respectively, and they are equal to $1/\rho_1$ and $1/\rho_2$.

Assuming the initial conditions 1 of the flow before the shock are known, equation 1.7, in a pressure-specific volume diagram, is the equation of a line. This line is called the Rayleigh line. This line must pass both from the initial state 1 and from the final state 2. The slope of the Rayleigh line is:

$$\frac{dP}{dv} = \frac{P_2 - P_1}{v_2 - v_1} = -\frac{{w_1}^2}{v_1^2} \tag{1.8}$$

It is possible to notice that the slope is proportional to the square of the flow velocity before the shock in the wave reference frame, w_1, which, according to equation 1.1, and assuming $U_s >> u_1$, can be approximated with the shock velocity U_s.

Due to the continuity equation 1.3, the slope of the Rayleigh line can also be expressed as a function of the post-shock speed and specific volume, as it follows:

$$\frac{dP}{dv} = \frac{P_2 - P_1}{v_2 - v_1} = -\frac{{w_2}^2}{v_2^2} \tag{1.9}$$

1.2.3 Hugoniot curve

Combining the energy conservation equation with the continuity equation and the momentum conservation equation, (or equivalently the equation of the Rayleigh line with the momentum conservation equation), the following equivalence is obtained:

$$h_2 - h_1 = \frac{1}{2}(P_2 - P_1)(v_2 + v_1) \tag{1.10}$$

This is the equation of the Hugoniot curve. In order to represent it, as the Rayleigh line, in a pressure-specific volume diagram, it is necessary to use an equation of state, where the enthalpy h can be expressed as a function of pressure P and specific volume v. For example, using the ideal gas equation of state:

$$Pv = RT \tag{1.11}$$

where T is the gas temperature and R is the universal gas constant divided by its molar mass, the Hugoniot relation can be rewritten as:

$$\frac{\gamma}{\gamma-1}(P_2 v_2 - P_1 v_1) - \frac{1}{2}(P_2 - P_1)(v_1 + v_2) = q \tag{1.12}$$

Where γ is the ratio of specific heats, defined as:

$$\gamma = \frac{c_p}{c_v} \tag{1.13}$$

Where c_p is the specific heat capacity at constant pressure and c_v the one at constant volume.

The parameter q is defined as:

$$q = h_1^0 - h_2^0 \tag{1.14}$$

and it is the difference of the enthalpies of formation between reactants and products.

The Hugoniot curve represents all the possible value couples (P_2, v_2) obtainable in the final condition, after the shock wave. The final condition, represented by the point 2, is determined by the intersection, in the pressure-specific volume diagram, of the Rayleigh line and the Hugoniot curve. In particular, if the chemical composition of the gases in the initial state 1 is the same as the one in the final state 2, and therefore no chemical reaction is happening, as in the case of a shock wave, the Hugoniot curve passes both from the initial point 1 and from the final point 2, as shown in Figure 1.2

On the other hand, in the presence of a chemical reaction, the reactants and the products will lay on the same Rayleigh line but they will belong to different Hugoniot curves. In particular, in this case, it is possible to distinguish between the following five regions represented in Figure 1.3.

The point A corresponds to the gas reactants initial state, so is equivalent to the state 1 of the previous equations. From this point, an horizontal and a vertical line are plotted, until the Hugoniot curve of the products is reached. Those two lines represent respectively a constant pressure and a constant volume combustion. Regions above the constant volume combustion (regions I and II), are the detonation regions, and regions below the constant pressure combustion (regions III and IV) are the deflagration regions. Regarding region V, it is possible to notice that here the pressure P_2 is higher than the gas initial pressure P_1, and also the specific volume v_2 is greater than the one of the reactants v_1. Therefore, looking at equation 1.8, which indicates the slope of a Rayleigh line, it is possible to notice that the left term is positive, while the right term is always negative. This means that there is no physical solution in this region, since the possible mathematical solutions involve a shock velocity w_1 which is imaginary.

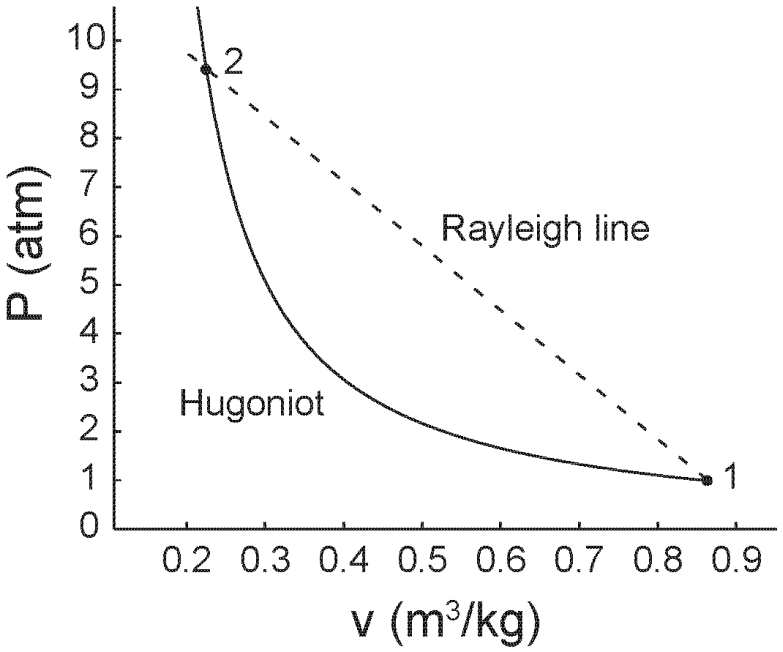

Figure 1.2: Hugoniot curve and Rayleigh line for a non reacting gas mixture (Browne et al., 2023)

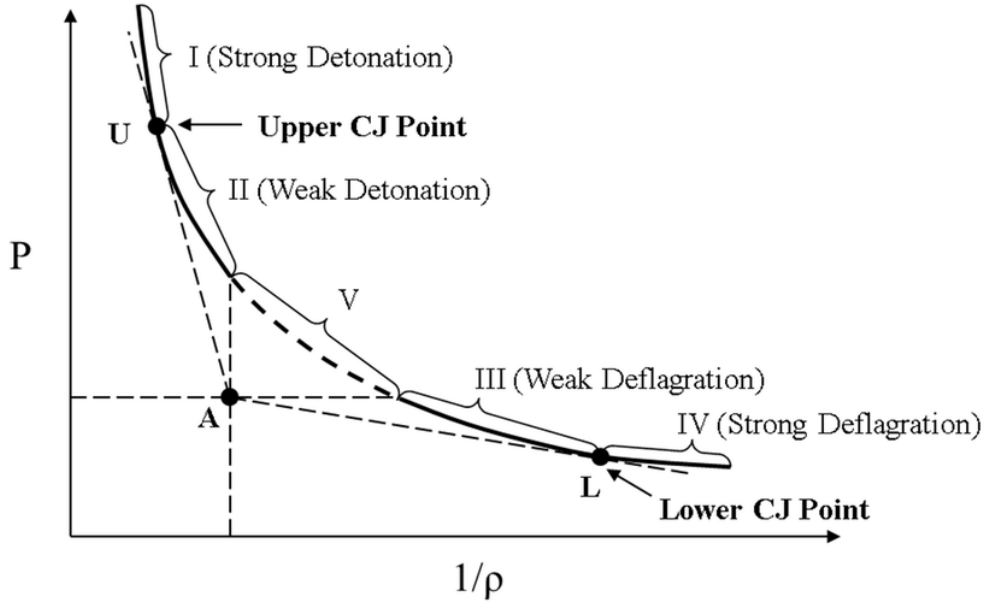

Figure 1.3: Possible solution regions on Hugoniot curve depending on the type of combustion (Kuo, 1986)

1.2.4 Upper Chapman-Jouguet point

Considering the detonation branches (region I and II), a Rayleigh line passing from the initial point A will always intersect the products Hugoniot curve in two points, the first one belonging to the weak detonation region (region II) and the second one belonging to the strong detonation region (region I). There is just one slope of the Rayleigh line that crosses the product Hugoniot curve at only one point, and therefore the line is tangent to the curve. This point is called the Upper Chapman-Jouguet (CJ) point, and corresponds to the point U in figure 1.3. The same applies to the deflagration branch, and the tangent point is called lower Chapman-Jouguet (CJ) point, which is represented by the point L in figure 1.3. Since the slope of the Rayleigh line is proportional to the square of the shock speed, the minimum value of shock speed for which it exist a solution in the detonation branch is called the Chapman-Jouguet speed, and the solution provided from this slope is the upper CJ point. Values of shock speed which are lower than the CJ speed will not intersect the Hugoniot curve of products, and values that are higher will intersect it in both the weak detonation region and in the strong detonation region.

1.2.5 Entropy and speed of sound

The speed of sound can be related to the slope of an isentropic curve in a $P - v$ diagram from the following expression:

$$c^2 = -v^2 \left(\frac{dP}{dv}\right)_{s=constant} \tag{1.15}$$

Where c is the speed of sound.

Four different isentropic curves, the Hugoniot curve and the Rayleigh line for a non-reacting gas are shown in figure 1.4

The first isentrope $s1$ passes from the initial state 1, while the last one $s4$ passes from the final state 2. In particular, $s1$ is smaller than $s4$, so that the condition reported in 1.6 is fulfilled. The other two isentropes have intermediate values between $s1$ and $s4$.

From the Rayleigh line slope equation 1.8, it is possible to deduce that:

$${w_1}^2 = -{v_1}^2 \frac{dP}{dv} \tag{1.16}$$

Since, from figure 1.4, it is evident that the slope of the Rayleigh line is greater than that of the isentrope $s1$ in the point 1, comparing the expression 1.16 with the sound speed equation 1.15 applied to the initial point 1, it is found that:

$${c_1}^2 = -{v_1}^2 \left(\frac{dP}{dv}\right)_{s=s1} < -{v_1}^2 \left(\frac{dP}{dv}\right)_{Rayleigh\ line} = {w_1}^2 \tag{1.17}$$

This means that the shock wave velocity w_1 is greater than the speed of sound relative to point 1, and so it is supersonic. This is true also for detonation waves, since they consist of a shock wave followed by a chemical reaction.

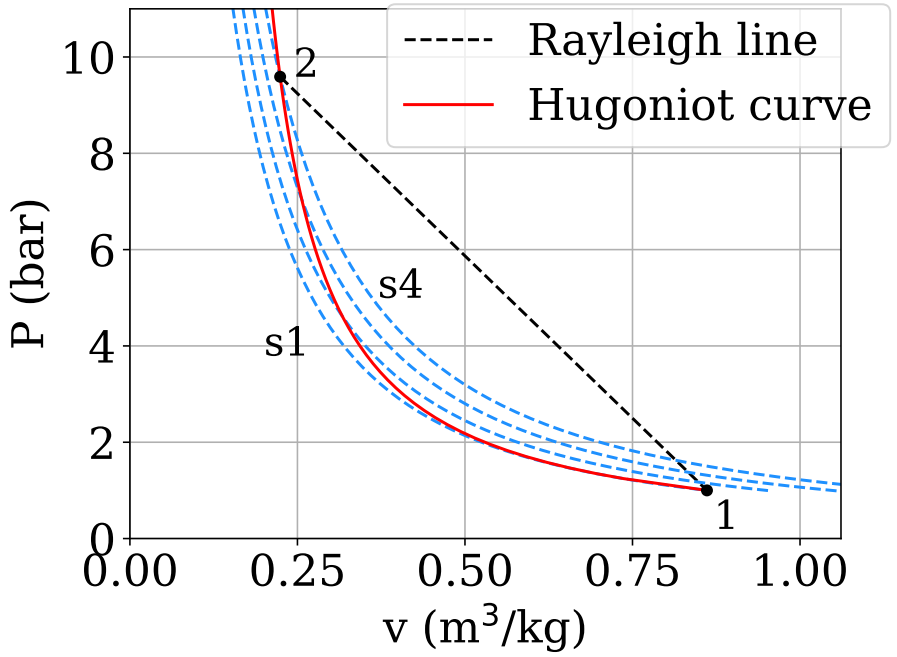

Figure 1.4: Hugoniot curve, Rayleigh line and 4 isentropes for a non reacting gas mixture

The same procedure can be applied in point 2, but this time the slope of the isentrope $s4$ is greater than the one of the Rayleigh line. So, comparing equation 1.9 with the speed of sound equation 1.15 in point 2, the following inequality is obtained:

$$c_2{}^2 = -v_2{}^2 \left(\frac{dP}{dv}\right)_{s=s4} > -v_2{}^2 \left(\frac{dP}{dv}\right)_{Rayleigh\ line} = w_2{}^2 \tag{1.18}$$

This means that in the wave-fixed reference frame, in the case of a shock wave in non-reacting gases, the speed of the gases in the final state w_2 is subsonic.

Considering a reacting gas mixture, for the reason explained before, the detonation wave velocity w_1 is supersonic. Regarding the products velocity in the wave-fixed reference frame w_2, figure 1.5 must be taken into account.

In figure 1.5, the Hugoniot product curve, the Rayleigh line having a CJ detonation wave speed, and the isentrope passing through the upper CJ point are represented. It is possible to notice that all the three lines, in the CJ point, are tangent to each other. This means that they all have the same slope. Therefore, comparing the equation of the Rayleigh line slope 1.9 with the sound speed equation 1.15 applied to the CJ point, it is found that:

$$c_{CJ}{}^2 = -v_{CJ}{}^2 \left(\frac{dP}{dv}\right)_{s=s_{CJ}} = -v_{CJ}{}^2 \left(\frac{dP}{dv}\right)_{Rayleigh\ line} = w_2{}^2 \tag{1.19}$$

This means that, at the CJ point, the speed of the products is sonic.

This can also be demonstrated analytically if the ideal gas equation of state is used.

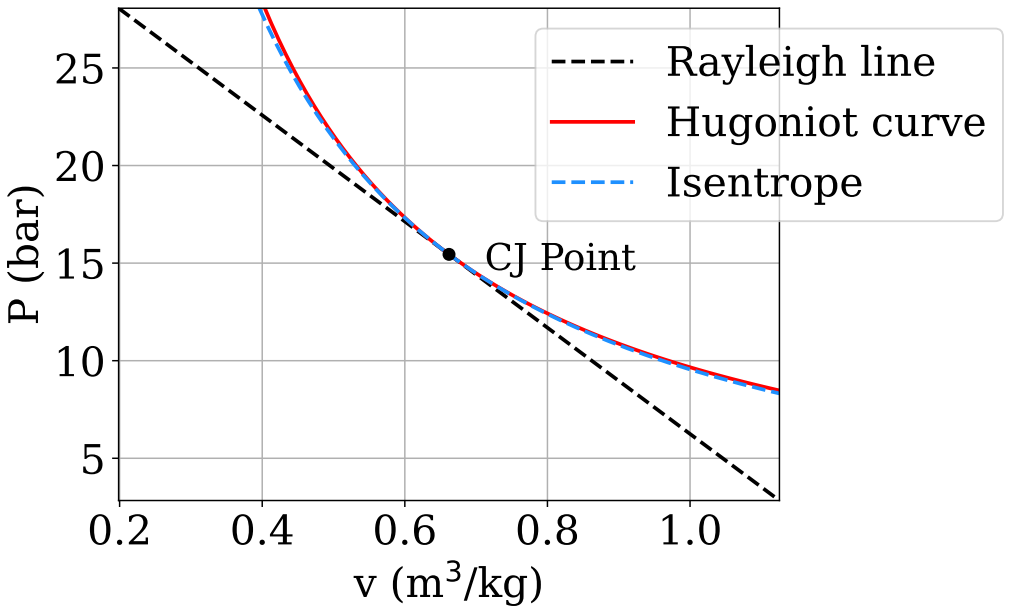

Figure 1.5: Hugoniot product curve, Rayleigh line and the isentropic curve passing through the CJ point for a reacting gas mixture, in the case of $w_1 = w_{CJ}$

The product Hugoniot curve equation for an ideal gas is reported in equation 1.12. It is possible to evaluate the slope of this curve by deriving this expression for v_2. After some mathematical manipulations, the following expression is obtained:

$$\frac{dP_2}{dv_2} = \frac{(P_2 - P_1) - (\frac{2\gamma}{\gamma-1})P_2}{(\frac{2\gamma}{\gamma-1})v_2 - (v_1 + v_2)} \tag{1.20}$$

At the same time, at the CJ point the slope of the Hugoniot is the same as the slope of the Rayleigh line, therefore:

$$\frac{dP_2}{dv_2} = \frac{P_2 - P_1}{v_2 - v_1} \tag{1.21}$$

By equating the last two expressions, the following one is obtained:

$$\frac{P_2 - P_1}{v_2 - v_1} = -\gamma\frac{P_2}{v_2} \tag{1.22}$$

Equating this last expression with the slope of the Rayleigh line expressed as a function of products velocity, equation 1.9, the following relation is obtained.:

$$w_2{}^2 = \gamma P_2 v_2 \tag{1.23}$$

Since, for an idela gas, the speed of sound is given by:

$$c = \sqrt{\gamma P v} \tag{1.24}$$

It is possible to state that:

$$w_2{}^2 = c_2{}^2 \tag{1.25}$$

Which means that, in the wave-fixed reference frame, the speed of the detonation products is exactly sonic at the CJ point.

Now, the case of a reacting gas mixture in which the detonation wave speed is higher than the CJ speed is considered. In this scenario, the Rayleigh line will intersect the products Hugonit curve in two points, one belonging to the strong detonation branch and one belonging to the weak detonation branch, as shown in figure 1.6.

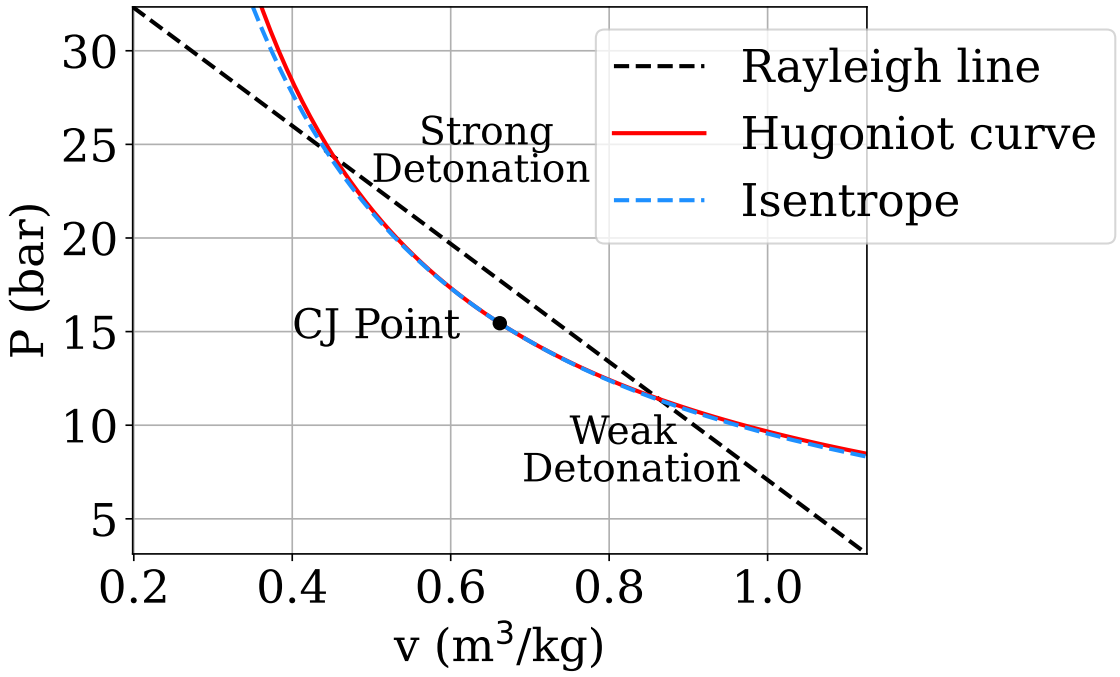

Figure 1.6: Hugoniot products curve, Rayleigh line and the isentrope passing through the CJ point for a reacting gas mixture, in the case of $w_1 > w_{CJ}$

It is possible to notice, both from figure 1.5 and figure 1.6, that the Hugoniot curve is well approximated by the isentrope passing through the CJ point.

Considering the intersection of the Hugoniot curve and Rayleigh line in the weak detonation region, it is evident that the slope of the isentrope is smaller than the one of the Rayleigh line; therefore:

$$c_2{}^2 = -v_2{}^2 \left(\frac{dP}{dv}\right)_{s=s4} < -v_2{}^2 \left(\frac{dP}{dv}\right)_{Rayleigh\ line} = w_2{}^2 \tag{1.26}$$

So, in the weak detonation region, products velocity w_2 is supersonic.

On the other hand, in the strong detonation region, the slope of the Rayleigh line is lower than that of the isentrope, and so:

$$c_2{}^2 = -v_2{}^2 \left(\frac{dP}{dv}\right)_{s=s4} > -v_2{}^2 \left(\frac{dP}{dv}\right)_{Rayleigh\,line} = w_2{}^2 \tag{1.27}$$

Which means that, in the strong detonation region, the products speed w_2 is subsonic.

Even if the Rayleigh line intersects the Hugoniot in two points, there must be only one solution to the detonation problem. The point that is usually experimentally observed in this condition is the one belonging to the strong detonation branch.

By rearranging equation 1.2, it is possible to say that:

$$U_s = w_2 + u_2 \tag{1.28}$$

Since, as it has been just shown in equation 1.27, in a strong detonation, the product speed in the wave fixed reference frame w_2 is subsonic, it follows that:

$$U_s < c_2 + u_2 \tag{1.29}$$

This means that, in the case of a strong detonation, in presence of any acoustic disturbance, which is moving at speed c_2, the resultant propagation velocity of the products (the sum of c_2 and u_2) is higher than the shock speed. Therefore, the acoustic disturbance and the products can catch up with the detonation wave, and reduce its strength. As a consequence, the detonation wave speed will be decreased. Once the detonation wave speed reaches the CJ value, the products velocity in the wave fixed frame w_2 became sonic, and, therefore, the detonation wave speed will no longer be affected by disturbances behind it, since:

$$U_s = w_2 + u_2 = c_2 + u_2 \tag{1.30}$$

In this condition, the detonation can sustain itself, since its strength will not be reduced by acoustic disturbances.

In conclusion, in the case of a strong detonation, the system, after some time, will stabilize at the CJ point. For this reason, in this thesis, the CJ point will always be used as the solution of a planar detonation wave.

1.2.6 Zel'dovich, Von Neumann and Doering theory (ZND theroy)

Zel'dovich, Von Neumann and Doering, during the Second world war, developed independently the same one-dimensional detonation theory. According to their model, in which a steady one-dimensional flow is assumed, a detonation wave consists of a shock wave followed by a reaction zone, where the chemical reaction occurs.

This can be visualized in a P-v diagram, as reported in Figure 1.7.

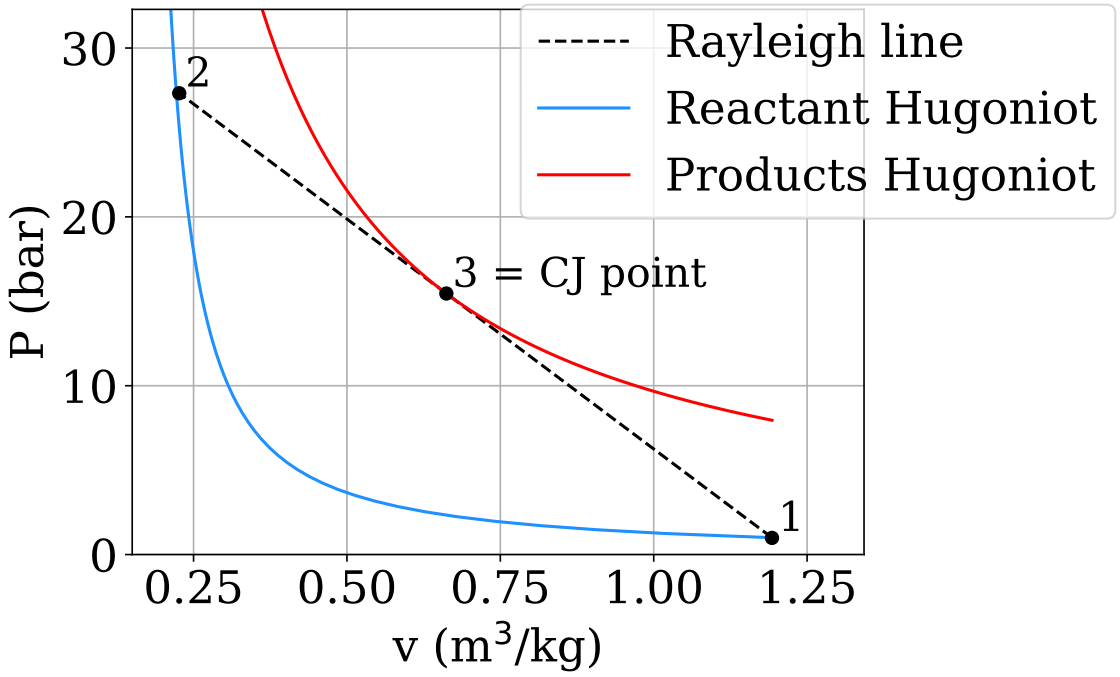

Figure 1.7: ZND theory visualized on a P-v diagram

With reference to Figure 1.7, point 1 corresponds to the reactants initial state. When a shock wave traveling at CJ speed crosses the reactant zone, reactant pressure and temperature increase. At this point, the gas mixture has not yet changed its chemical composition, so its post-shock state, point 2, lays on the same Hugoniot curve which crosses point 1. The post-shock condition 2 is found by intersecting the reactant Hugoniot curve with the Rayleigh line having slope as expressed in 1.8, with w_1 being the CJ speed. The increase in temperature and pressure triggers the chemical reaction to begin. Point 3 represents the final state of the detonation products, once the equilibrium chemical composition is reached, and corresponds to the CJ point. It is determined by the intersection of the products Hugoniot curve and the same Rayleigh line described before. It is important to notice that, since a steady flow is assumed, all the stages of the detonation process will lay on the same Rayleigh line, since this line is a consequence of the mass conservation law. In particular, starting from point 2, the Rayleigh line will cross an infinite number of Hugoniot curves before reaching point 3, each of them corresponding to a partial chemical equilibrium state.

Pressure, temperature and density of the gas mixture, during the detonation process, change as shown in Figure 1.8.

In particular, it is possible to notice that the pressure reaches its maximum after the shock, and this peak pressure is known as Von Neumann spike. What happens after the shock wave can be divided in two main parts: the induction zone and the reaction zone. In the induction zone the reaction rate increases slowly, and is following the Arrhenius law. The slow increase of the reaction rates causes pressure, temperature and density to stay almost constant. During the reaction zone, the reaction rate increases rapidly, and it is possible

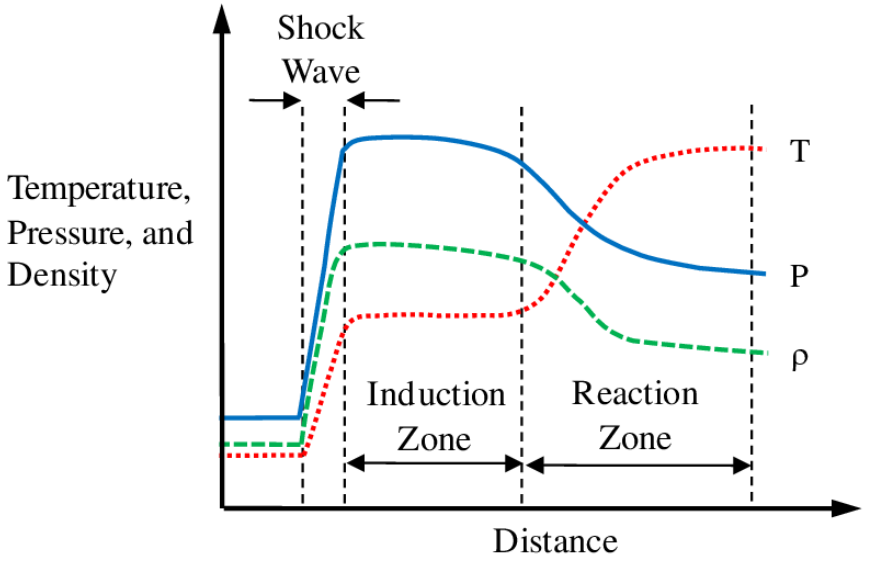

Figure 1.8: Thermodynamic properties variation during a detonation process (Rouser, 2012)

to observe a huge variation of the thermodynamic properties of the mixture.

Also, the Mach number is changing between state 2 and 3 of figure 1.7. In fact, after a shock wave, as discussed previously in equation 1.18, the speed of the flow in the wave-fixed reference frame, w_2, is subsonic. Then, due to the decrease in pressure, the flow is accelerated until the sonic speed, which characterizes the CJ point, is reached in point 3.

1.2.7 Limits of 1-D detonation theory

In experiments dealing with detonation, what is found is that the detonation wave always travels at a speed that is lower than the CJ one. But, according to the one-dimensional detonation theory, the CJ speed corresponds to the minimum speed value at which it is possible to obtain a solution belonging to the detonation branch. So, how is it possible that in the experiments where detonation is observed, the detonation wave has a speed lower than the CJ one?

The limit of the one-dimensional detonation theory lies in the assumption of a planar wave front. In fact, the high pressure reached in correspondence of the Von Neumann spike always deforms the recipient or pipe in which the detonation is happening. As a result, the flow undergoes an expansion, which reduces the detonation wave strength, and consequently the shock wave front will no longer be perfectly planar, but will instead show a certain degree of curvature. The curvature of the detonation front is the reason why the detonation speeds observed in experiments are always lower than the CJ speed. In particular, the difference between the actual detonation speed and the CJ one will be greater the more the radius of the tube is small. This is due to the reason that the expansion happens near the tube walls, therefore, if the radius is big enough, the curvature will be really small and the shock front, except from really close to the walls, can be considered

almost planar. In case of small radius, it is instead necessary to apply some correction to the one-dimensional detonation theory. Another aspect to take into account is the tube material stiffness. In fact, the more the tube material will oppose to the expansion, the more the detonation wave front is going to be similar to a planar one.

In addition to that, especially in the case of gases, detonation waves always present a complex three-dimensional structure, which cannot be predicted with the 1-D detonation model.

The discipline that studies the deviations from the ideal detonation theory is called "Detonation Shock Dynamics", and is not reported here, since all the presented rotating detonation combustor models assume a planar detonation front.

1.3 Detonation in thermodynamic cycle analysis

It is already possible to notice, from Figure 1.3, how deflagration combustion, which is the type of combustion traditionally used in a Joule-Brayton cycle, presents a huge difference in specific volume between reactants and products, while the pressure difference is really small. This is the reason why deflagration combustion is often approximated with an isobaric process.

On the other hand, in a detonation process, the specific volume change between reactants and products is negligible compared to the pressure change, with pressure being way much higher in the products. For this reason, as a first approach, detonation can be modeled as an isochoric process, but this consists of an underestimation of the product pressure compared to using the CJ point as the solution of the detonation.

At the same time, the fact that the pressure is higher in the products, and so detonation is a pressure-gain process, leads to higher thermodynamic efficiency compared to using deflagration combustion. To demonstrate this, the simple Joule-Brayton cycle represented in Figure 1.9 must be considered.

In this case, the gas mixture in point 1 is composed of hydrogen and air at a pressure P_1 of 1 bar and a temperature T_1 of 20^0 C. The mixture of hydrogen and air is stoichiometric, and air is assumed to be composed only of oxygen and nitrogen. The molar composition of the mixture is reported in table 1.1.

	Moles
H_2	2
O_2	1
N_2	3.76

Table 1.1: Mixture composition in moles

Starting from the same initial condition 1 and using the same components represented in Figure 1.9, three different cycles are analyzed. In the first cycle, a constant pressure combustion is applied, the second presents a constant volume combustion, and in the

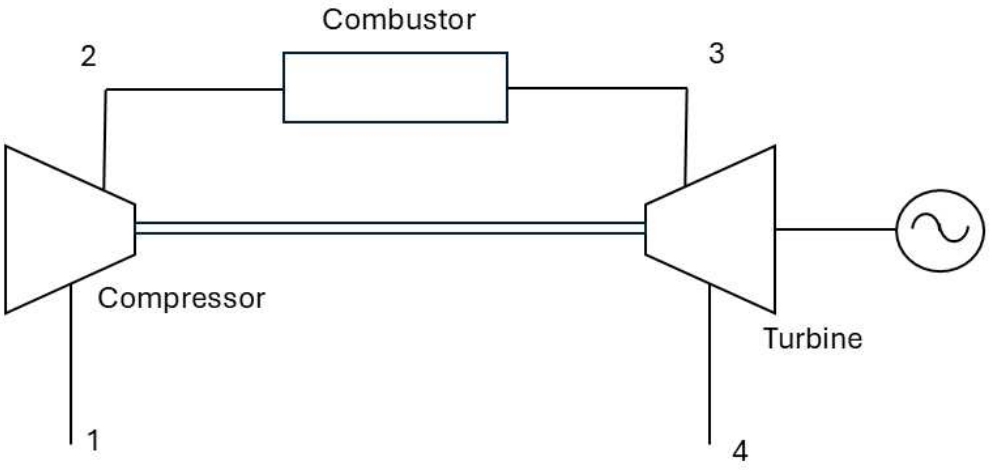

Figure 1.9: Joule-Brayton cycle usual components

last one, a CJ detonation combustion is used. In all the three cycles, the gas mixture is compressed with the same pressure ratio β equal to 5, with β being:

$$\beta = \frac{P_2}{P_1} \tag{1.31}$$

As a consequence of this, in all three cycles, state 2 is exactly the same. Both the compression and the expansion process are assumed to be isentropic.

The three different cycles are represented in a pressure-specific volume diagram (with the pressure axis in a logarithmic scale) in Figure 1.10

Regarding the cycle featuring the detonation combustion, the Von Neumann pressure spike is neglected, and the flow passes directly from the state in point 2 to the post-detonation state, point 3, which corresponds to the CJ point.

Figure 1.11 represents instead the three cycles in a temperature-specific entropy diagram.

For all the three cases, the useful specific work of the cycle is computed as the specific work generated by the turbine minus the specific work used to run the compressor:

$$W_{useful} = W_{turbine} - W_{compressor} \tag{1.32}$$

The useful specific work W_{useful} can be rewritten applying the first law of thermodynamics for open systems to the turbine and the compressor, assuming that for both the components the flow inlet velocity is comparable with the one at the outlet. This results in:

$$W_{useful} = (h_3 - h_4) - (h_2 - h_1) \tag{1.33}$$

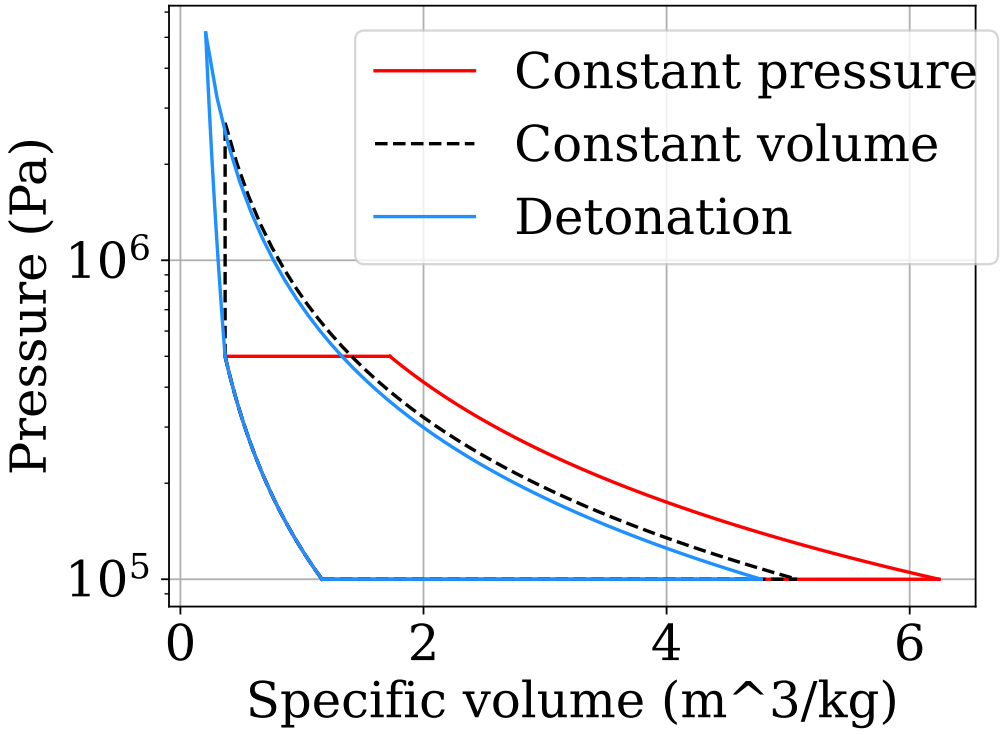

Figure 1.10: Thermodynamic cycle of Figure 1.9 using a constant pressure combustion, a constant volume combustion and a detonation combustion represented in a P-v diagram

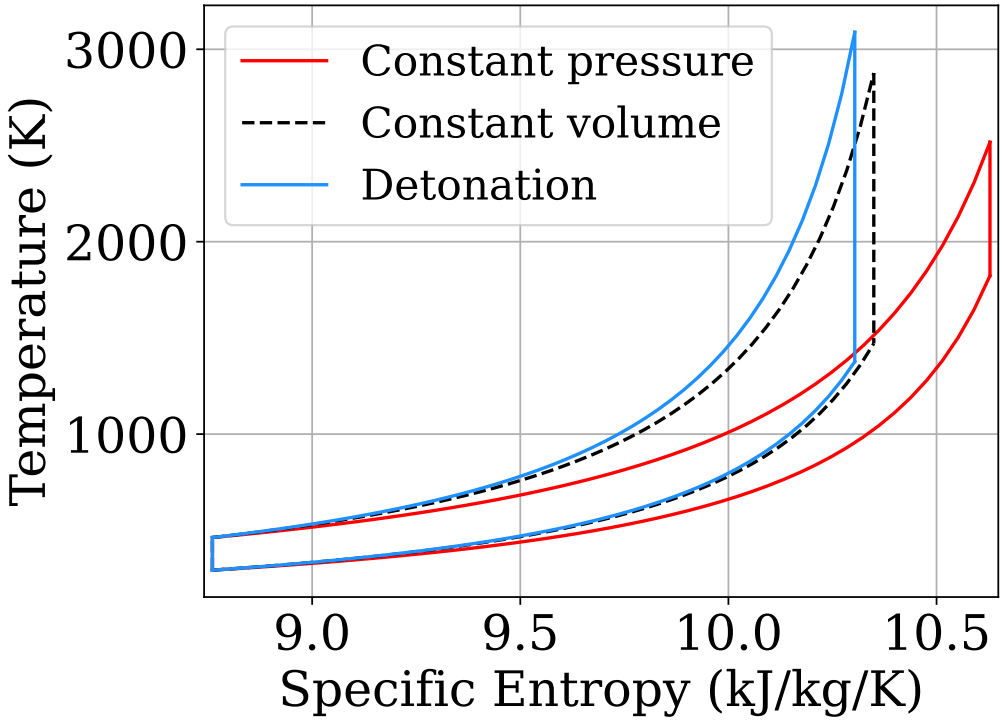

Figure 1.11: Thermodynamic cycle of Figure 1.9 using a constant pressure combustion, a constant volume combustion and a detonation combustion represented in a T-s diagram

With h being the specific enthalpy of the gas mixture in the point expressed by its subscript.

Finally, it is possible to compute the cycle efficiency as a ratio of specific useful work W_{useful} over the specific heat provided to the system through the combustion process. This heat, for an ideal gas, can be evaluated as:

$$q = c_p \Delta T \tag{1.34}$$

With c_p being the mixture gas specific heat at constant pressure, and ΔT the change in temperature between products and reactants. c_p anyway is not a constant, but is a function of temperature. So, considering that in the case taken into account there is a big change in temperature between reactants and products during the combustion process, the specific heat q has been evaluated considering an average c_p between states 2 and 3:

$$q = \frac{c_{p2} + c_{p3}}{2}(T_3 - T_2) \tag{1.35}$$

The cycle efficiency η, as described before, is then evaluated as:

$$\eta = \frac{W_{useful}}{q} = \frac{(h_3 - h_4) - (h_2 - h_1)}{\frac{c_{p2}+c_{p3}}{2}(T_3 - T_2)} \tag{1.36}$$

The following results have been obtained:

	Efficiency η
Constant pressure combustion	0.29
Constant volume combustion	0.56
Detonation combustion	0.64

Table 1.2: Efficiency comparison of the three cycles

As expected, the detonation cycle is the one presenting the highest efficiency. This is a consequence of the following two aspects:

- The detonation cycle, as it is possible to observe from figures 1.10 and 1.11, is the one that shows an higher pressure and temperature at the turbine inlet (point 3), allowing the turbine to extract more work from the gas mixture.
- The detonation cycle, as it is possible to notice from figure 1.11, is the one which implies the least amount of specific entropy change during the cycle. Since the change of entropy is associated with losses due to irreversibilities, the detonation cycle results to be the one with less irreversibility losses, and, as a consequence, with the highest efficiency.

Anyway, even if the detonation cycle seems to be really promising in terms of efficiency, there are some limitations to it. First of all, in case a turbine is used for expanding the

gas mixture and extracting useful work, there is no material, nowaday, that can tolerate a temperature as high as the one reached by the detonation combustion, which, as it is possible to notice in figure 1.11, exceeds the 3000 K. Therefore, a cooling of the gas mixture or a dilution of the reactants with some inert gas to reduce the turbine inlet temperature is necessary. This will result in a decrease of the efficiency. In addition, as it will be explained better in later chapters, in the case of a rotating detonation engine, the CJ conditions, which in this analysis are used as the point of inlet of the turbine, can be assumed to exist in a really thin layer just after the shock front. Before reaching the turbine inlet, therefore, the gas mixture undergoes an expansion, which reduces the mixture pressure and temperature, and consequently the cycle efficiency.

1.3.1 Literature review of detonation in thermodynamic cycle analysis

The story of detonation started with Abel, 1869, which measured the detonation velocities of different types of reactants. The mathematical model to evaluate the theoretical detonation velocity of a mixture was published by Chapman, 1899 and Jouguet et al., 1904, whose theory is known as the Chapman-Jouguet (CJ) theory. The CJ model, despite being still of great importance today, does not explain the structure of a detonation. This topic has been described lately by Zeldovich, 1940, Von Neumann, 1942 and Doring, 1943, which proposed the model of a detonation wave structured as the composition of a shock wave followed by a chemical reaction (ZND theory). The oblique shock on the detonation front has been observed for the first time by Campbell, 1927. Studies on the detonation cellular structure and detonation cell size have been conducted by A. A. Vasil'ev, 1982, A. A. Vasil'ev et al., 1987, Lee and Radulescu, 2005 and A. Vasil'ev, 2006. Research about the phenomenon of the deflagration-to-detonation transition has been conducted by Urtiew and Oppenheim, 1966, LEE et al., 1966, Lindstedt and Michels, 1989, Lee, 2008 and Ciccarelli and Dorofeev, 2008.

Due to its thermodynamic advantage compared to deflagration, detonation combustion has been implemented in many propulsion systems. One example is the oblique detonation wave engine, whose performance has been studied by Ostrander et al., 1987, Ashford and Emanuel, 1996, Valorani et al., 2001, Sislian et al., 2001,Miao et al., 2018. Another famous example of a detonation-based propulsion system is the pulse detonation engine (PDE), whose performance evaluation and thermodynamic cycle analysis are reported in Wintenberger and Shepherd, 2006. Other works related to the pulse detonation engine have been conducted by Hutchins and Metghalchi, 2003, Yan and Fan, 2005, Glaser et al., 2007, Shimo and Heister, 2008, Peng et al., 2013, George, Driscoll, Gutmark, and Munday, 2014, Lu and Zheng, 2016, Roux, 2015 and Xisto et al., 2018. Works related to the thermodynamic cycle analysis have been conducted by Alhussan et al., 2016, who performed a cycle comparison between detonation combustion, constant volume combustion and constant pressure combustion, first at the same values of pressure ratio and then at the same value of turbine inlet temperature. Vutthivithayarak et al., 2012 compared three possible cycles to model the thermodynamic performance of a PDE, concluding that the Zeldovich–von Neumann–Doring process is the most appropriate one. Wolanski, 2011

discussed the thermodynamic cycle of a detonation engine and described the advantages of a rotating detonation engine over a pulsed detonation engine. Finally, Assad and Tunik, 2023 compared the detonation cycle with the Otto and Brayton cycles at the same value of compressure ratio, concluding that the detonation cycle shows an advantage in terms of efficiency compared to the other two, but, at really high values of compression ratios (higher than 65), the Bryton cycle presents a higher value of efficiency. Also, if the three cycles are compared at the same values of limiting temperature of the equivalent Carnot cycle, the detonation cycle underperforms the other two in terms of efficiency and work output, but still shows a lower value of internal irreversibilities.

2. Theoretical fundamentals of rotating detonation combustors

2.1 Working principle

A three-dimensional representation of the flow field in a rotating detonation combustor is shown in figure 2.1, together with some geometrical properties.

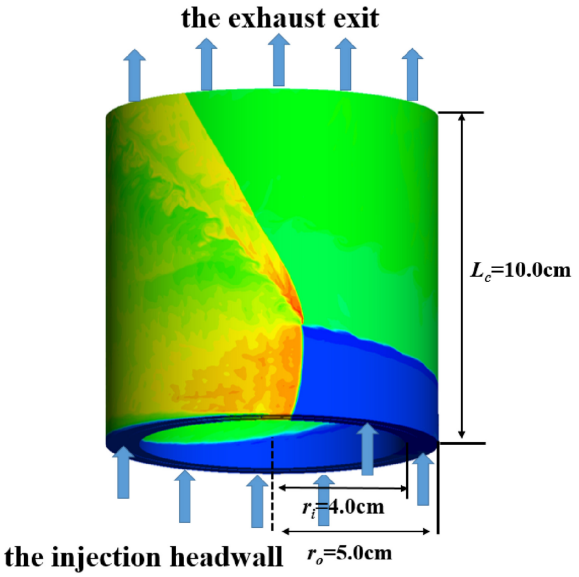

Figure 2.1: 3D view of a flow field in a rotating detonation combustor (Liu et al., 2020)

As it is possible to observe, the fuel and the oxidizer are injected axially in an annulus chamber, where a detonation wave is travelling around the combustor in the azimuthal direction. In figure 2.1 only one detonation wave is present, but experiments and numerical simulations have shown that the formation of more than one wave is possible. The continuous axial injection of the mixture assures the sustainability of the detonation wave and allows the combustor to work at high operational frequency, typically in the kilohertz range.

If the channel width of the chamber is negligible compared to its radius and height, the three-dimensional flow field inside a rotating detonation combustor can be simplified and visualized on a 2-D plot, with the combustor axial height on the vertical axis and the combustor circumferential length on the horizontal axis. Figure 2.2 shows an unwrapped flow field of a rotating detonation combustor.

As it is possible to observe, four main lines can be distinguished, all of which have a common point, called the 'triple point.' The first line is the line of the detonation front, and it separates the fresh injected mixture, on its right, from the post-detonation products (combustion products), on its left. From the interaction between the detonation wave and the surrounding flow, an oblique shock is formed, represented by the shock wave line. The third characteristic line is the slip line (contact surface b in figure 2.2), which

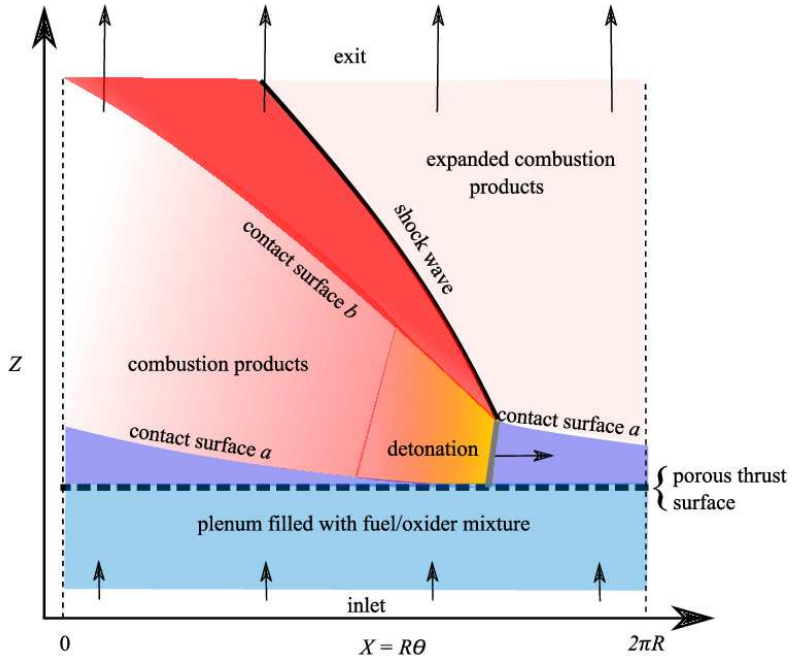

Figure 2.2: Flow field of a rotating detonation combustor in a 2D unwrapped view (Shepherd & Kasahara, 2017)

separates the detonation products from the previous cycle, which have just crossed the oblique shock, from the newly formed detonation products. In the post-detonation region, close to the detonation front, the fluid reaches extremely high temperatures and pressures, which prevents the mixture injection in the chamber. The fluid is then subject to a rapid expansion, which makes those two quantities gradually decrease. In particular, once the pressure in the combustor chamber is low enough to match the mixture injection pressure, the refill of the combustor can start again, and an almost-triangular refilling zone is formed, delimited by the injection slip-line (contact surface a in figure 2.2).

2.2 Main sources of losses

In this section, the main sources of losses of a rotating detonation combustor are reported and commented. For thermodynamic cycle analysis purposes, it is important to obtain the maximum possible temperature and pressure at the outlet of the combustor, together with the minimum possible entropy increase. Therefore, every physical mechanism that is decreasing temperature and pressure and/or increasing entropy will be considered as a loss. The highest conditions of temperature and pressure in a rotating detonation combustor, as explained before, are obtained just after the detonation front. For an ideal, planar, detonation front, assuming that the fuel has a speed perpendicular to the detonation front and equal to the CJ speed in the wave-fixed reference frame, they are represented respectively from the temperature and pressure at the CJ point.

The main loss mechanism, which leads to a decrease in temperature and pressure, is represented by the expansion of the detonation products. This expansion will lead the combustor to have temperature and pressure at the outlet that strongly differ from the CJ

ones. Therefore, for a rotating detonation engine, it is not possible to consider the CJ point as the outlet point of the combustor (and the inlet point of the turbine), since this will lead to an overestimation of the cycle thermal efficiency. For a more detailed thermodynamic cycle analysis of a rotating detonation cycle, detonation product expansion must be taken into account.

Another source of loss is represented by the oblique shock wave. Despite increasing temperature and pressure of the fluid that are crossing it, it also increases entropy and represents, therefore, a source of irreversibility. Anyway, the entropy increase generated by the oblique shock wave is small if compared to the one created by the detonation, as shown by Kaemming et al., 2017, and reported in figure 2.3.

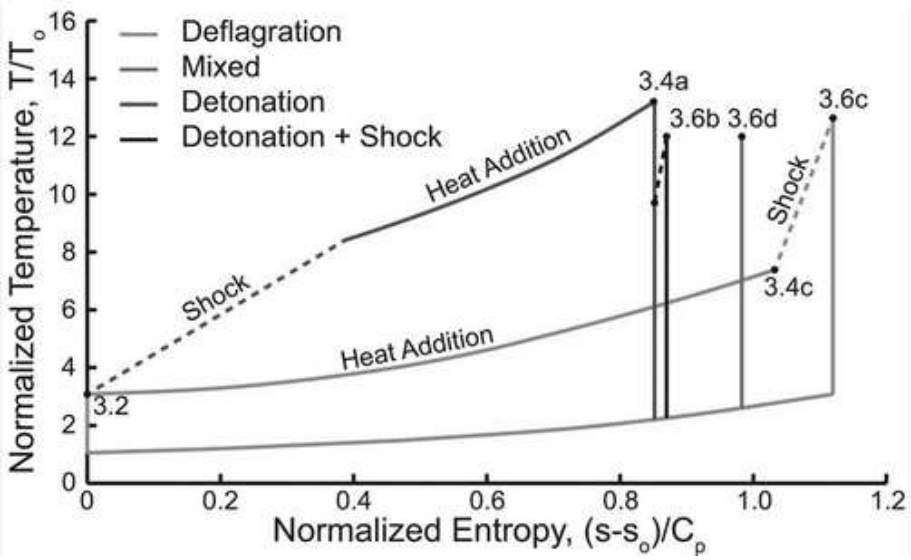

Figure 2.3: Main thermodynamic flow path of a rotating detonation combustor shown in a temperature-entropy diagram (Kaemming et al., 2017)

In figure 2.3, the point 3.2 represents the thermodynamic state that corresponds to the mixture injection conditions in the detonation chamber. The mixture is then subject to a detonation process, which, according to the ZND theory, can be treated as a shock wave followed by a chemical reaction. Therefore, point 3.4a, in case of ideal detonation, corresponds to the CJ point. The flow field, after the detonation, undergoes an expansion process, and a part of the flow field will then cross the oblique shock wave. Point 3.6b represents the post-oblique-shock thermodynamic condition. It is possible to observe that the entropy increase caused by the oblique shock wave is really small.

Always from figure 2.3, another source of loss can be visualized. In fact, at the beginning of the refilling zone, despite chamber pressure having reached the same value of the injection one, the chamber temperature is still significantly higher than the mixture injection temperature. Therefore, along the injection slip-line (contact surface a in figure 2.2), a part of the mixture is burned by deflagration. This represents a source of loss, since this means that not the totality of the injected fuel will be processed by the detonation wave. In addition to that, as previously discussed in sections 1.2 and 1.3, the deflagra-

tion generates a higher entropy increase compared to detonation and presents also lower values of final temperature and pressure. In figure 2.3, the deflagration is represented as a constant-pressure process, that starts from point 3.2 and ends in point 3.4*c*. Finally, also the deflagrated flow crosses the shock wave, creating another increase in entropy. The deflagrated flow post shock condition is represented by point 3.6*c*.

The last source of loss is represented by the backflow. In fact, in the region just after the detonation front, the high values of pressure of the detonation products are preventing mixture injection. Anyway, if the pressure of the products is too high, they can even enter the propellant injector manifold and potentially cause an explosion. In order to avoid that, fuel and oxidizer are usually injected separately in the combustion chamber, leading to additional losses due to the mixing mechanism.

2.3 Literature review of rotating detonation engine

The rotating detonation engine was patented by Wolanski, Fujiwara, and Mitsubishi in 2004. Studies about rotating detonation waves have been performed by F. A. Bykovskii, Zhdan, and Vedernikov, 2006, George, Driscoll, Anand, and Gutmark, 2017, Katta et al., 2019 and Wen, Xie, and Wang, 2019. Experimental research on RDE has been conducted by F. A. Bykovskii, Mitrofanov, and Vedernikov, 1997, F. A. Bykovskii and Vedernikov, 2003, F. A. Bykovskii, Zhdan, and Vedernikov, 2005, F. A. Bykovskii, Zhdan, and Vedernikov, 2006 and F. Bykovskii and Zhdan, 2013. In particular, these experiments were really useful for understanding the conditions in which the sustainability of rotating detonation waves is achieved.Russo, 2011 studied the influence of injection mass flow rate and equivalence ratio on rotating detonation engines fueled with hydrogen and air. Frolov, Aksenov, and Ivanov, 2015 experimentally demonstrated the Zeldovich cycle efficiency gain over cycle with constant pressure combustion for hydrogen-oxygen fuel mixture. Rankin et al., 2017 used chemiluminescence in order to visualize the propagation of the detonation waves around the annular channel. Numerical simulations of rotating detonation combustors have been performed by Hishida et al., 2009, Schwer and Kailasanath, 2010, Schwer and Kailasanath, 2011a, Kindracki et al., 2011, Schwer and Kailasanath, 2011b, Frolov, Aksenov, et al., 2015, Cocks et al., 2016, Tsuboi et al., 2015 and Sun et al., 2018.

Reduced order models to estimate performances of a rotaing detonation engine have been proposed by Braun et al., 2013, Nordeen et al., 2014, Kawashima et al., 2017, Shepherd and Kasahara, 2017, Kaemming et al., 2017, Sousa et al., 2017, Bach et al., 2021, Wen, Fan, and Wang, 2023 and Kanda and Inagaki, 2024.

3. Power plant configurations

In this section, the proposed power plant configurations will be described and analyzed, as well as fuel and oxidizer selection.

3.1 Fuel and oxidizer

In order to assure the integration of RDE power plants with renewable energy sources, hydrogen represents the perfect choice as far as fuel selection is concerned. In fact, one way to produce hydrogen is by using the excess energy produced by renewable sources through an electrolysis process. Hydrogen produced in this way is called 'green hydrogen,' since it represents a perfectly clean and CO2 emission-free process for its production. Hydrogen production represents, therefore, an energy storage mechanism, since, after being produced, it can be burned to have the energy back whenever it is needed. In addition to that, the combustion of hydrogen does not produce CO2, and therefore an RDE power plant, if run with this fuel, results in being carbon dioxide emission-free. Lastly, hydrogen is well-known for its detonability properties, which will facilitate RDE operation.

The oxidizer that will be explored is air, since detonation of hydrogen with pure oxygen leads to too extreme post-detonation conditions.

3.2 Configurations

3.2.1 Power plant A

The first proposed power plant configuration consists of the same components of a standard Joule-Brayton cycle, with the exception that in this case the deflagrative combustor is replaced with a rotating detonation combustor. A schematic representation of the power plant components can be visualized in figure 3.1.

Air and hydrogen are assumed to enter the compressor already premixed and at ambient conditions (pressure = 1 bar, temperature = 25°C), in point 1. Then after the compression, in stage 2, the mixture enters the rotating detonation combustor. After the combustion has occurred, the post-detonation products will be expanded in a gas turbine in order to extract useful work. Finally, the exhaust gases can be expelled into the atmosphere or, if possible, used as a heat source. In figure 3.1 only the thermodynamic-relevant components are shown, but, before being expelled into the atmosphere, it is important for exhaust gasses to undergo some NOx-removing treatment, since they represent an important source of pollution for the atmosphere.

3.2.2 Power plant B

The second proposed power plant configuration is shown in figure 3.2.

Its configuration is very similar to the previous one, since the only difference consists in the mixture cooling before the injection into the rotating detonation combustor. This, as

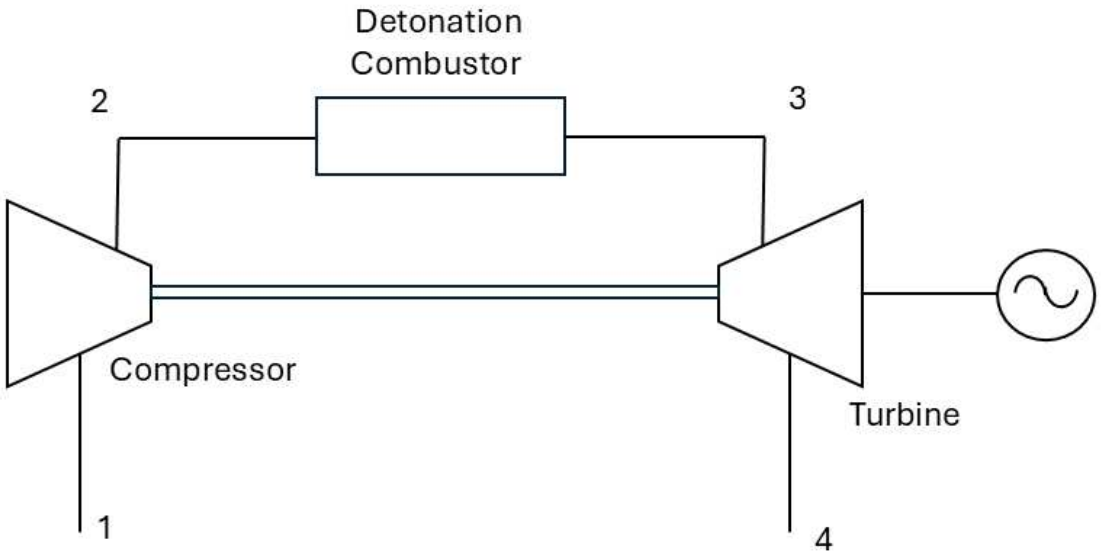

Figure 3.1: Schematic of components of power plant A

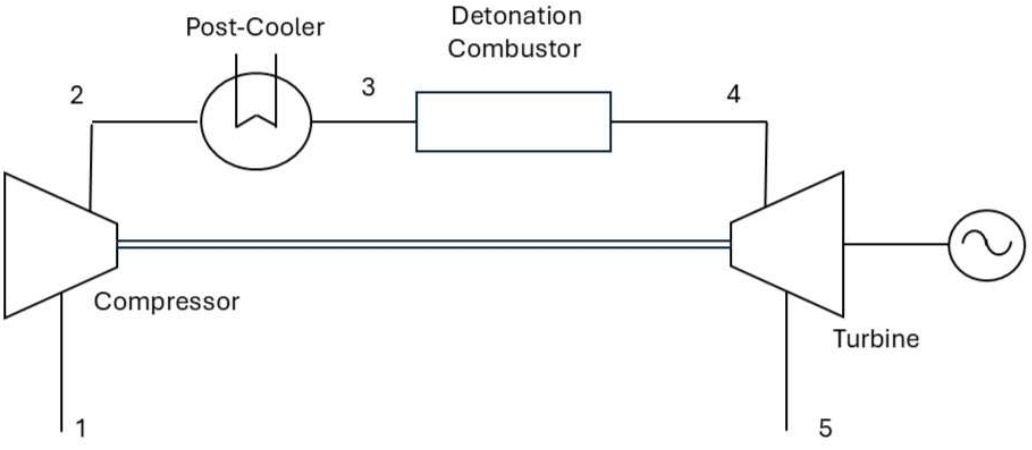

Figure 3.2: Schematic of components of power plant configuration B

it will be shown later, allows the system to increase its efficiency, since, at the same value of injection pressure, with lower injection temperature, higher values of post-detonation conditions are obtained. Also in this case, the fuel is hydrogen, while air is used as an oxidizer.

4. Methods

4.1 Simulation Tools

All the analysis carried out in this thesis has been performed in a Python environment with the usage of the library Cantera (Goodwin et al., 2018). Cantera consists of an open-source suite of tools that can be used to solve problems involving chemical kinetics, thermodynamics, and transport processes. In particular, in the context of this work, it is used to solve combustion problems and evaluate the gas thermodynamic properties in different conditions.

To start, an object gas must be created, specifying the desired mechanism file to use. This operation is done with the `Solution` function, as reported here:

```
[gas]=Solution(mech)
```

Where `[gas]` is the Cantera gas object created, and `mech` is a string that contains the name of the mechanism file.

Cantera already provides a lot of different mechanism files from which it is possible to choose, but at the same time the user can also create its own personalized mechanism file. In this work, the mechanism '`gri30_highT.yaml`', already present in Cantera, has been used for all the analysis.

Once a gas object has been created, it is important to set its state. To do that, two thermodynamic properties must be assigned to the gas mixture, as well as the mixture's chemical composition. An example is here reported:

```
gas.TPX = 300, 1e5, 'H2:2 O2:1 N2:3.76'
```

In this case, the state of the gas has been set equal to the one of a stoichiometric hydrogen-air mixture at a pressure of 1 bar and a temperature of 300 K.

As an alternative to temperature and pressure, it is possible to set the state using the following combinations of thermodynamic quantities:

- `gas.TD` to set temperature and density
- `gas.HP` to set specific enthalpy and pressure
- `gas.UV` to set specific internal energy and specific volume
- `gas.SP` to set specific entropy and pressure
- `gas.SV` to set specific entropy and specific volume

Once the thermodynamic state has been set, it is possible to extract all the other thermodynamic quantities from the gas object. For example:

```
h=gas.h
```

will extract the specific enthalpy.

Cantera can also be used to compute the chemical equilibrium of a mixture. For doing that, the method `equilibrate` is used. For example, the chemical equilibrium of a gas mixture at constant pressure and temperature is found in the following way:

```
gas.equilibrate('TP')
```

Other chemical equilibrium options are:

- `equilibrate('HP')`: computes the chemical equilibrium at constant specific enthalpy and pressure
- `equilibrate('UV')`: computes the chemical equilibrium at constant specific internal energy and specific volume
- `equilibrate('SV')`: computes the chemical equilibrium at constant specific entropy and specific volume
- `equilibrate('SP')`: computes the chemical equilibrium at constant specific entropy and pressure

In addition to Cantera, the library Shock and Detonation Toolbox (Browne et al., 2023) has been used. This library consists of a set of functions based on Cantera specifically developed for solving shock and detonation equations. In particular, the CJ speed of a mixture can be computed with the `CJspeed` function, as follows:

```
U_cj= CJspeed(P1,T1,q,mech)
```

Where:

- `U_cj` = CJ speed
- P1 = mixture initial pressure (Pa)
- T1 = mixture initial temperature (K)
- q = string describing the mixture chemical composition
- mech = Cantera mechanism file

The post-detonation conditions are obtained from the function `PostShock_eq`, which presents the following syntax:

```
[gas] = PostShock_eq(U1,P1,T1,q,mech)
```

Where:

- `[gas]` = Cantera gas object
- U1 = detonation wave velocity
- P1 = mixture initial pressure (Pa)
- T1 = mixture initial temperature (K)
- q = string describing the mixture chemical composition

- mech = Cantera mechanism file

Another function used from the Shock and Detonation Tolbox is the function `soundspeed_fr`, which computes the sound of speed relative to the input gas. It presents the following syntax:

```
a = soundspeed_fr(gas)
```

Where `a` is the sound of speed and `gas` is a Cantera gas object.

4.2 Detonation maps

As a preliminary investigation, the post-detonation conditions as a function of different values of inlet pressure and temperature are explored. This work has been performed in order to determine if there are any 'more convenient' inlet characteristics, which can potentially lead to higher power-plant efficiencies.

The following analysis has been carried out in a Python environment. The thermodynamic properties and the chemical composition of the post-detonation products are evaluated using the library 'Cantera' (Goodwin et al., 2018), while the solution of the detonation equations is provided by the 'Shock and Detonation Toolbox' (Browne et al., 2023), which is a library based on Cantera specifically developed for solving shock and detonation equations.

First, a stoichiometric mixture of hydrogen and air is considered. The injection pressure has been varied between 1 and 20 bar, while the range of the inlet temperature is 300-500 K. The post-detonation solutions provided are the ones corresponding to the CJ state, and the analysis has been carried out with the Cantera gas mechanism file `gri30_highT.yaml`.

In figure 4.1, the post-detonation temperature as a function of different mixture initial temperature and pressure is shown.

What is observed is that the highest values of post-detonation temperature are obtained in correspondence with the maximum values of injection temperature and pressure. Therefore, post-detonation temperature is increasing for increasing values of injection temperature and pressure. Anyway, at the same time, it is worth noticing that the post-detonation temperature is varying by roughly 300 K, for a quite wide range of different inlet conditions. In particular, it is possible to observe that it is quite significantly affected by inlet pressure, but only slightly influenced by inlet temperature. Therefore, increasing the initial temperature does not result in significant changes in post-detonation temperature.

Figure 4.2 shows the post-detonation pressure.

In figure 4.2, it is first of all possible to observe that, unlike post-detonation temperature, which can be considered almost independent from initial conditions, post-detonation pressure is strongly affected by mixture inlet conditions, particularly by inlet pressure. The higher the inlet pressure, the higher the post-detonation pressure, with values that can reach up to 320 bar for an inlet value of 20 bar. Another interesting result from figure 4.2 is

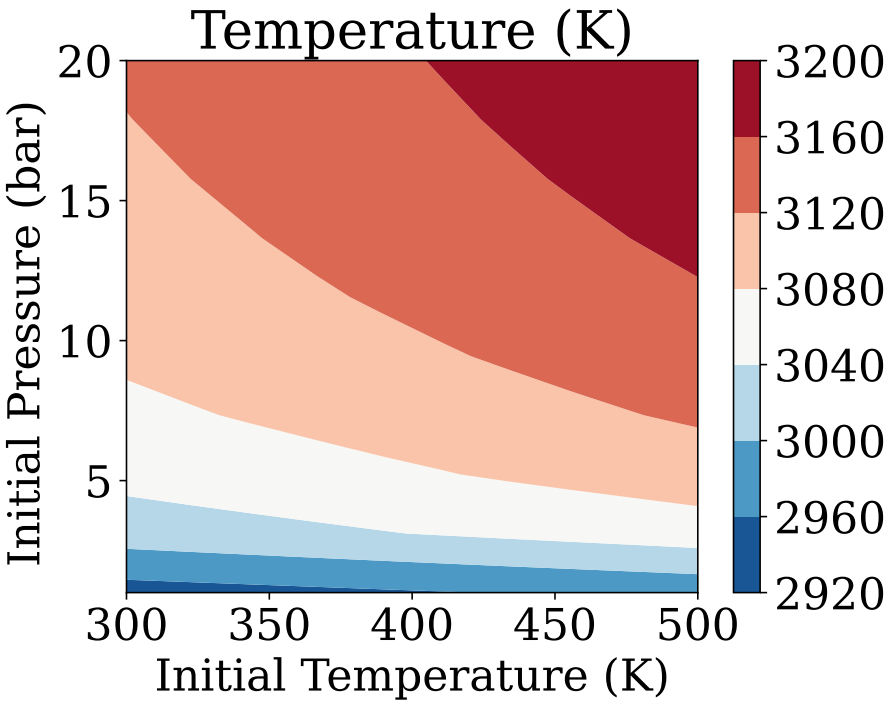

Figure 4.1: Post-detonation temperature of a stoichiometric hydrogen-air mixture for different values of inlet pressure and temperature

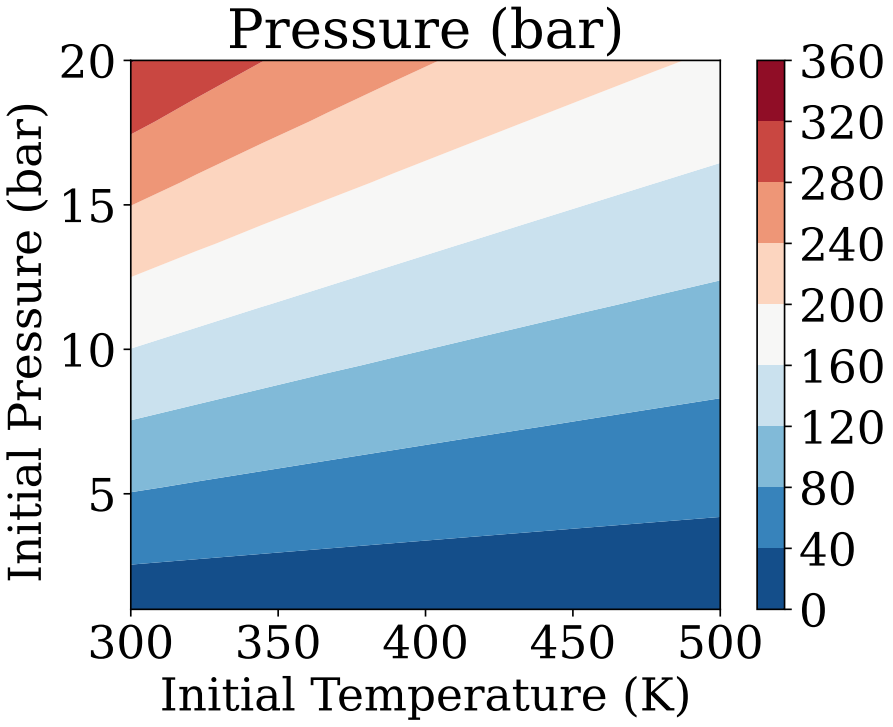

Figure 4.2: Post-detonation pressure of a stoichiometric hydrogen-air mixture for different values of inlet pressure and temperature

that post-detonation pressure decreases with the increase of inlet temperature. Therefore, the highest values of post-detonation pressure are produced by a high inlet pressure and a low inlet temperature of the mixture.

As a consequence, since post-detonation temperature is almost independent from the initial temperature, while post-detonation pressure is significantly higher in correspondence with lower mixture temperatures, the addition of a post-cooler after the compressor, as in power plant B (figure 3.2), is expected to increase the cycle efficiency compared to power plant A (figure 3.1).

In figure 4.3, the CJ speed of a stoichiometric hydrogen-air mixture is represented for different values of mixture inlet temperature and pressure.

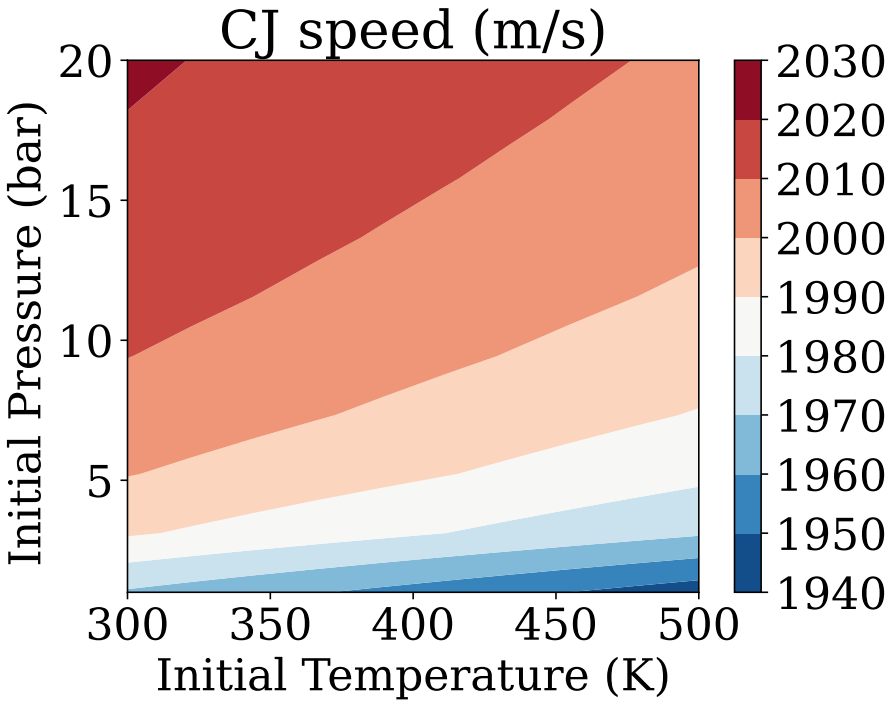

Figure 4.3: CJ speed of a stoichiometric hydrogen-air mixture for different values of inlet pressure and temperature

It is possible to notice that the CJ speed follows the same trend as the post-detonation pressure, being higher for high injection pressures and low injection temperatures. Since, as explained in section 1.2, the CJ speed corresponds almost to the detonation wave speed, the combustor is expected to operate at a higher frequency in correspondence with higher values of CJ speed.

Finally, figure 4.4 shows the NOx mass fraction in the post-detonation products of a stoichiometric hydrogen-air mixture for different values of inlet pressure and temperature.

Figure 4.4 shows that the higher mass fraction value of NOx in the detonation products is found for high values of inlet temperature and low values of inlet pressure.

In conclusion, the detonation maps have not found a more convenient injection point, since all the quantities taken into consideration present maximum and minimum values

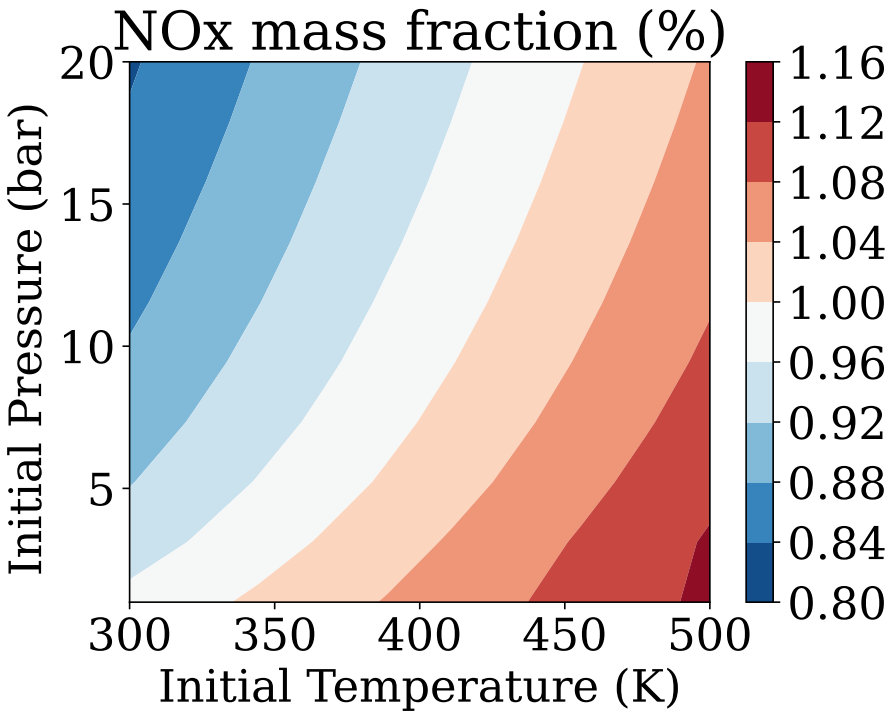

Figure 4.4: NOx mass fraction produced by a stoichiometric hydrogen-air mixture for different values of inlet pressure and temperature.

always in correspondence with the corners of the detonation maps. Therefore, it is always possible to increase or decrease the considered post-detonation conditions by increasing or decreasing the inlet conditions.

Anyway, the addition of a post-cooler, as in Power Plant B (3.2), is expected to have the following impacts:

- higher post-detonation pressure
- slightly lower post-detonation temperature
- higher thermodynamic cycle efficiency
- higher combustor operational frequency
- lower NOx emissions

4.3 Rotating detonation combustor 1D model

In this section, a one-dimensional model for predicting outlet temperature and pressure of a rotating detonation combustor is presented and discussed. The described model is based on works of Sousa et al., 2017 and Kanda and Inagaki, 2024. The Python code of the model is reported in section 0.1.

4.3.1 Model description

The model starts with acquiring the following input values:

- Mixture injection temperature
- Mixture injection pressure
- Detonation front height
- Combustor Radius

The first two parameters represent the thermodynamic conditions of the mixture before entering the combustor. In particular, in the case of power plant A, they correspond to the conditions in point 2 of figure 3.1, which also coincides with the compressor outlet condition. In the case of power plant B, these values are the ones obtained after the mixture post-cooling, represented schematically as point 3 of figure 3.2.

Regarding the detonation front height, F. A. Bykovskii, Zhdan, and Vedernikov, 2006 demonstrated that it is in the range of $(12 \pm 5)\lambda$, with λ being the detonation cell size, which, for a stoichiometric hydrogen-air mixture, is about 15 mm . As an alternative, if results from a computational fluid dynamic simulation are known, it is possible to estimate the detonation height directly from the flow field picture, as in figure 2.2.

In the same way, if known, the combustor radius can be set directly. As an alternative, F. A. Bykovskii, Zhdan, and Vedernikov, 2006 found that, for a stable operation of the rotating detonation combustor, the ratio between the detonation front height and the circumferential length should be in the range 0.14 ± 0.04. Therefore, the combustor circumferential length can be estimated from the detonation front height and, as a consequence, the combustor radius.

The first step of the model consists of evaluating the CJ speed and the post-detonation condition. For calculating those quantities, the Shock and Detonation Toolbox (Browne et al., 2023) has been used. In particular, the CJ speed has been evaluated with the `CJspeed` function, and the post-detonation conditions are obtained from the function `PostShock_eq`.

The `PostShock_eq` function returns a Cantera gas object, from which it is possible to obtain all the thermodynamic properties as well as the post-detonation chemical composition of the detonation products at the equilibrium state. The detonation wave velocity U1, in the case of a planar and ideal detonation, corresponds to the CJ speed. A detailed description of the numerical methods implemented in these two functions is described in Browne et al., 2023, as well as simplified analytical expressions derived for an ideal gas model.

Suddenly, an iterative process to evaluate the mixture injection velocity and the refilling angle is initiated. In figure 4.5, the speed triangle at the inlet of the combustor is represented, together with the injection refilling zone.

The injection velocity V_{inj} is assumed to be completely axial, while the detonation wave speed U is assumed to be completely azimuthal. Since the detonation is presumed to be ideal, in the wave-fixed reference frame, the mixture is entering the detonation front at a speed equal to the CJ one W_{cj}. The angle β between the detonation wave speed U and the CJ speed W_{cj} is assumed to be equal to the refilling angle, and only one detonation wave

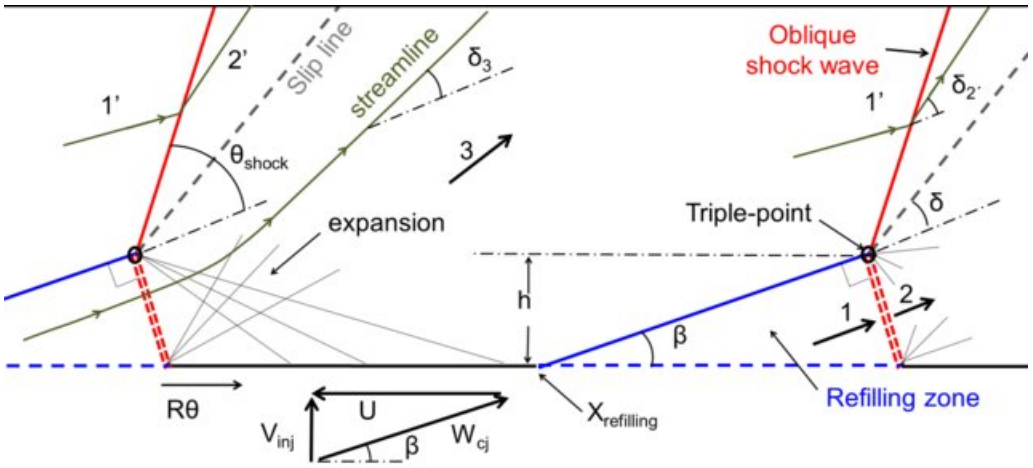

Figure 4.5: Injection refilling zone and speed triangle at the inlet of the combustor (Sousa et al., 2017)

per cycle is considered. The iterative process starts with guessing the injection velocity. After that, knowing the CJ speed, the injection refilling angle can be evaluated as:

$$\beta = sin^{-1}\left(\frac{V_{inj}}{W_{cj}}\right) \tag{4.1}$$

Since only one detonation wave per cycle is considered, the distance between the two triple points of figure 4.5 is equal to the combustor circumferential length, $2\pi R$, with R being the combustor radius.

Therefore, the position of X_{ref}, which is the point where the refilling of the combustor starts, can be found as:

$$X_{ref} = 2\pi R - \frac{h}{tan(\beta)} \tag{4.2}$$

considering a reference system that has its origin in the correspondence of the orthogonal projection of the first triple point on the horizontal line, with h being the detonation front height.

The next step consists of evaluating the total pressure in the wave-fixed reference frame after the detonation front. The static pressure after the detonation, as well as the value of the specific heat ratio in the post-detonation state, can be extracted from the Cantera gas object resulting from the `PostShock_eq` function. The total pressure can then be evaluated from the isentropic flow equation as follows:

$$P_{t_2} = P_2\left(1 + \frac{\gamma_2 - 1}{2}\right)^{\frac{\gamma_2}{\gamma_2 - 1}} \tag{4.3}$$

With P_2 being the static pressure and γ_2 the specific heat ratio in the post-detonation point. This expression is valid for an ideal detonation, where the speed of the post-detonation

products at the CJ point is exactly sonic and therefore presents a Mach number equal to 1.

The flow field after the detonation front is resolved in an analytical way, as proposed by Kanda and Inagaki, 2024. In figure 4.6, a schematic representation of all the quantities considered for the analytical solution is shown.

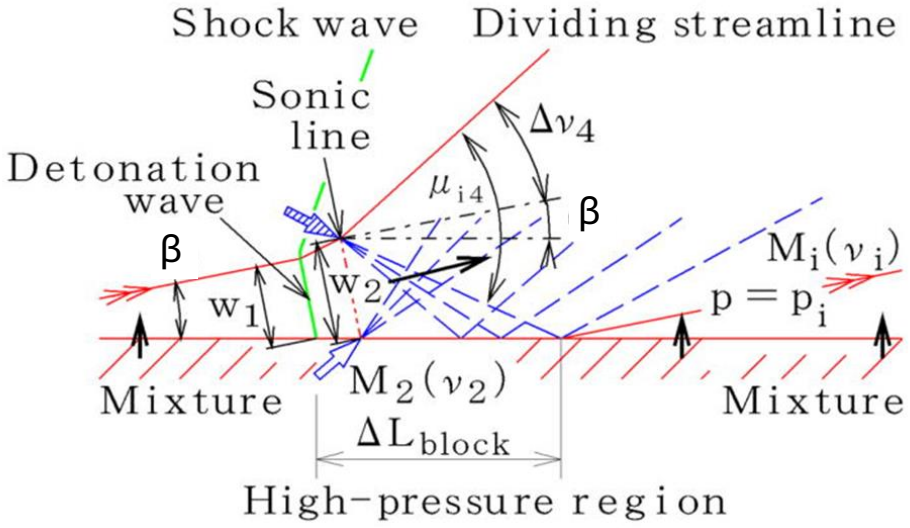

Figure 4.6: Schematic representation of characteristic lines after the detonation front (Kanda & Inagaki, 2024)

After the detonation front, where the sonic point is reached, the flow undergoes an expansion, which lowers its temperature and especially pressure, and accelerates the flow from sonic to supersonic in the wave-fixed reference frame. The expansion in this case is assumed to be isentropic, and the backflow of the detonation products in the injection manifold is neglected. As it is possible to observe from figure 4.6, two main Prandtl-Meyer expansion fans turn the flow from an initial angle of β, one to zero degrees and the other to the angle of the dividing streamline, which is the line that separates the detonation products just formed from the ones of the previous cycle that crossed the oblique shock. Anyway, for this analytical solution, the strength of the two expansion fans is considered to be the same, and the Mach lines of the expansion fans are assumed to be straight lines.

Applying the isentropic flow equation, it is possible to evaluate the Mach number of the detonation products in the wave-fixed reference frame at the point where the mixture injection of the following cycle starts, as follows:

$$M_i = \sqrt{\frac{2}{\gamma_2 - 1}\left(\left(\frac{P_{t_2}}{P_1}\right)^{\frac{\gamma_2 - 1}{\gamma_2}} - 1\right)} \tag{4.4}$$

With P_1 being the mixture injection pressure, since mixture refilling can only start if the

detonation products reach a pressure equal to the injection one.

Knowing the Mach number of the detonation products at the start of the injection, it is possible to evaluate the Prandtl-Meyer function at this point with the following formula:

$$\nu_i = \sqrt{\frac{\gamma_2+1}{\gamma_2-1}} tan^{-1}\left(\sqrt{\frac{\gamma_2-1}{\gamma_2+1}(M_i^2-1)}\right) - tan^{-1}\left(\sqrt{M_i^2-1}\right) \tag{4.5}$$

The Prandtl-Meyer function represents the angle through which you must expand a sonic flow to obtain a given Mach number.

According to gas dynamic theory, when a flow is subjected to two expansion fans of the same strength, which are turning it in opposite directions, it will have a final value of the Prandtl-Meyer function that is double the one caused by a single expansion fan. After having crossed the two initial expansion fans, the flow also undergoes the reflection of the expansion waves on the walls, as represented in figure 4.6, before reaching the point where the mixture injection restarts. Therefore, after crossing the wave reflected on the wall, the value of the Prandtl-Meyer function doubles again, and the angle $\Delta\nu_4$, represented in figure 4.6, can be computed as:

$$\Delta\nu_4 = \frac{\nu_i}{4} \tag{4.6}$$

Knowing the values of $\Delta\nu_4$, the corresponding Mach number M_4 must be evaluated by reversing the Prandtl-Meyer function in a numerical way or in an analytical way, as described by Hall, 1975. The angle μ_{i4} of figure 4.6 represents the Mach angle associated with M_4, and is evaluated as follows:

$$\mu_{i4} = sin^{-1}(M_4) \tag{4.7}$$

By knowing the angles μ_{i4}, ν_4, the detonation front height h, and with the estimation of β, it is possible to calculate the position where the refilling of the fresh mixture starts:

$$X_{ref} = \frac{h}{tan(\mu_{i4} - (\beta + \nu_4))} \tag{4.8}$$

If the value of X_{ref} is coherent with the one found before, then the iteration ends. If not, by using the value of X_{ref} obtained from equation 4.8, a new value of the injection angle β is computed as:

$$\beta = tan^{-1}\frac{h}{2\pi R - X_{ref}} \tag{4.9}$$

With R being the combustor radius. With this new angle, a new value of the injection velocity is obtained as:

$$V_{inj} = W_{cj} sin(\beta) \tag{4.10}$$

And the process is started again.

The combustor outlet conditions are computed assuming an axial and sonic (in the laboratory reference frame) flow at the combustor outlet.

The mixture injection velocity, in the laboratory reference frame, presents only an axial component. This means that there is no angular momentum in the combustor's bottom part, and, consequently, angular momentum should be zero also at the combustor outlet. Therefore, the velocity of the detonation gas products at the outlet of the combustor is assumed to be only axial. Experiments and more detailed numerical simulations show that the outlet velocity of a rotating detonation combustor is not purely axial; anyway, it is observed that some parts of the outlet flow present a circumferential velocity in one direction and some others in the opposite one. As a consequence, the hypothesis of purely axial outlet flow results to be acceptable if the average outlet velocity is considered.

The other hypothesis made for evaluating the outlet conditions consists of assuming the outlet flow to be choked. This has been shown to be representative of a really wide range of rotating detonation combustor operating conditions. Anyway, simulation results showing both subsonic and supersonic exit flow are present.

With these two assumptions, the combustor outlet temperature is easily found from the post-detonation total temperature, which is evaluated as:

$$T_{t_2} = T_2 \left(1 + \frac{\gamma_2 - 1}{2}\right) \tag{4.11}$$

With T_2 being the post-detonation temperature given by the `PostShock_eq` function. This value is relative to the wave-fixed reference frame, in which the post-detonation products have a sonic speed.

Since to pass from the laboratory reference frame to the wave-fixed one it is necessary to subtract vectorially the detonation wave tangential velocity U, which is evaluated, with reference to figure 4.5, as:

$$U = W_{cj} cos(\beta) \tag{4.12}$$

the outlet velocity in the laboratory reference frame results in being equal to the axial component of the outlet velocity in the wave-fixed reference frame.

Assuming conservation of energy between the post-detonation conditions and the combustor outlet, the exit temperature is evaluated as:

$$T_e = T_{t_2} - \frac{1}{2cp_2} u_e^2 \tag{4.13}$$

With cp_2 being the specific heat at constant pressure of the detonation products and u_e the outlet velocity of the gas in the wave-fixed reference frame.

In particular, the outlet velocity is formed by its axial and circumferential components. Therefore:

$$u_e^2 = u_{ea}^2 + u_{ec}^2 \tag{4.14}$$

The axial component, as discussed before, is equal to the one of the laboratory reference system, that is assumed to be sonic. It can therefore be replaced with the following expression:

$$u_{ea} = \sqrt{\gamma_2 R_2 T_e} \tag{4.15}$$

With R_2 being the gas costant divided for the detonation gas products molar mass.

Also, since the only difference between the two reference systems is the detonation wave velocity U, it follows that:

$$u_{ec} = U \tag{4.16}$$

By substituting equations 4.15 and 4.16 in 4.13, the following expression for evaluating the outlet temperature is found:

$$T_e = \frac{T_{t_2} - \frac{1}{2cp_2}U^2}{1 + \frac{1}{2cp_2}\gamma_2 R_2} \tag{4.17}$$

By knowing the exit temperature T_e, the exit velocity can then be evaluated using equation 4.15.

The outlet pressure of the combustor is found by imposing mass conservation and axial momentum conservation between the combustor inlet and outlet. In particular, knowing the point where the fresh mixture injection starts X_{ref}, the injection area per unit of thickness can be computed as:

$$A_{inj} = 2\pi R - X_{ref} \tag{4.18}$$

Consequently, the inlet mass flow rate is evaluated as:

$$\dot{m}_{inj} = \rho_{inj} V_{inj} A_{inj} \tag{4.19}$$

To evaluate the inlet axial momentum, a pressure equal to the post-detonation pressure P_2 is assumed to act on the part between the detonation front and the start of the refilling zone. In reality, the pressure on the lower wall gradually decays from the post-detonation value to the injection one. Therefore, this assumption can lead to higher values of combustor outlet pressure and pressure gain.

The inlet axial momentum is composed of the sum of the contribution of the inlet mass flow rate, of the injection pressure acting on the injection refilling zone, and of the pressure acting on the blocked area. The expression to evaluate it is the following:

$$F_i = \dot{m}_{inj} V_{inj} + P_1 A_{inj} + P_2 X_{ref} \tag{4.20}$$

Finally, by equating inlet and outlet mass flow rates and axial momentum, the outlet pressure is found with the following expression:

$$P_e = \frac{F_i}{2\pi R} - \dot{m}_{inj} u_{ea} \tag{4.21}$$

4.3.2 Advantages and limitations of the model

The presented model is based on the equations of mass conservation and momentum conservation between the combustor inlet and outlet. Also, the outlet flow is assumed to be sonic and completely axial in the laboratory reference frame. The main advantages of this model are:

- Simplicity of implementation
- Execution time

Regarding the simplicity, the model consists of really few equations that are describing the main features of the rotating detonation combustor, while providing, at the same time, a good estimation of the combustor average outlet temperature and pressure. As a consequence, the model results in being both easy to implement and computationally fast, providing results in a matter of seconds even on an average computer.

This simplicity, anyway, is counterbalanced by the following limitations:

- The model assumes sonic exit flow. Despite this condition having been shown to be appropriate for a wide range of combustor operating conditions, it does not represent the totality of the possible cases.
- The model assumes perfectly axial outlet flow. This condition is justified by the fact that rotating detonation combustors show an exit flow that presents positive components of circumferential velocity in some points and negatives in others, which leads to a net swing close to zero. Anyway, also in this case, this condition is not true for the totality of the cases.
- The model does not consider the effect of the oblique shock wave
- The model does not consider the backflow of the detonation products in the injection manifold
- The model does not consider the burning by deflagration of the fresh mixture along the injection slip line, assuming all the fuel to be combusted by detonation
- Injection losses are neglected
- The presented model is suited for just one detonation wave per cycle.

4.4 Rotating detonation combustor 2D model

In this section, a two-dimensional model for predicting outlet temperature and pressure of a rotating detonation combustor is presented and discussed. The described model is based on the work of Sousa et al., 2017. The Python code relative to the 2D model is reported in section 0.2.

4.4.1 Model description

The input parameters needed for the model are the following:

- Mixture injection total pressure
- Mixture injection total temperature
- Detonation front height
- Combustor Radius
- Combustor axial length

Regarding those quantities, the same considerations reported in section 4.3 are still valid.

The present model is quite similar to the previous one, but, instead of finding the outlet conditions by imposing an axial sonic outlet gas velocity, the entire post-detonation flow field is solved with a method of characteristics.

As in the previous model, an iterative process to calculate the combustor injection velocity is set up as described below.

The model starts by estimating an initial mixture injection velocity V_{inj}, which, as in the previous model, is assumed to be completely axial. From this guess value, the static values of injection pressure and temperature can be found from the stagnation ones with the following equations:

$$T_{inj} = T_0 - \frac{1}{2}\frac{V_{inj}^2}{cp_1} \tag{4.22}$$

$$P_{inj} = P_0 \left(\frac{T_{inj}}{T_0}\right)^{\frac{\gamma_1}{\gamma_1 - 1}} \tag{4.23}$$

With T_0 and P_0 being the total inlet temperature and pressure respectively, cp_1 being the mixture constant-pressure specific heat and γ_1 the mixture specific heats ratio.

By knowing the two static injection thermodynamic properties, the function `CJspeed` by the Shock and Detonation Toolbox is used to evaluate the CJ speed. The syntax, inputs, and output of this function have already been reported in section 4.1.

By reference to figure 4.5, knowing the CJ speed and with the guess value of V_{inj}, the refilling angle β can be computed as:

$$\beta = sin{-}1\left(\frac{V_{inj}}{W_{cj}}\right) \tag{4.24}$$

With W_{cj} being the CJ speed.

Also in this case, the refilling angle is assumed to be the same as the angle between the CJ speed and the detonation wave speed U. As in the previous model, only one detonation wave per cycle is assumed to be present. Therefore, a first estimation of the injection refilling X_{ref} can be obtained with the following expression:

$$X_{ref} = 2\pi R - \frac{h}{tan(\beta)} \tag{4.25}$$

with h being the detonation front height and R the combustor radius.

Suddenly, the post-detonation conditions are evaluated. This is done with the `PostShock_eq` function from the Shock and Detonation Toolbox. This function has already been commented in section 4.1. Also in this case, the detonation is assumed to be ideal; therefore, as input parameter regarding the detonation wave speed, the CJ speed W_{cj} is used. In particular, the post-detonation pressure P_2, temperature T_2 and specific heat ratio γ_2 need to be extracted from the Cantera gas object resultants from the `PostShock_eq` function, since they will be needed for the following analysis.

Before starting with the resolution of the flow field with the method of characteristics, the oblique shock angle θ_{shock} and the slip line angle δ of figure 4.5 need to be evaluated. For this model, both the oblique-shock line and slip-line are assumed to be straight lines. Anyway, while for the oblique-shock line this assumption results in being accurate, for the slip-line numerical simulations have shown that it presents a curvature.

Sichel and Foster, 1980, proposed an analytical model to evaluate those two angles, which can also be adapted and used in the case of a rotating detonation combustor. In particular, their model was presented for analyzing a gaseous detonation bounded by an inert gas, like, for example, air. The analysis starts by assuming a plane detonation wave, which is moving at a speed equal to the CJ one in a cloud of mixture and oxidizer, as shown in figure 4.7.

All the analysis is carried out in the wave-fixed reference frame; therefore, the vector C of figure 4.7 represents the velocity of the mixture, which is travelling towards the detonation wave with a speed equal to the CJ one. The interaction of the detonation with the surrounding bounding layer has the effect of generating the oblique shock line and the interface line, which separates the inert gas that crossed the oblique shock wave from the detonation products. The main idea of the model consists in assuming an equilibrium condition along the interface line. This can be expressed, in a mathematical way, with the following two expressions:

$$P_{i2} = P_{e3} \tag{4.26}$$

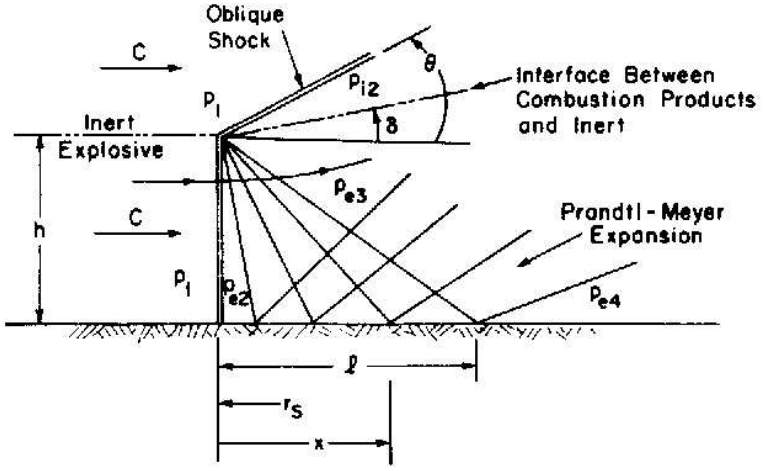

Figure 4.7: Schematic representation of a plane detonation bounded by an inert gas (Sichel & Foster, 1980)

$$\delta_{i2} = \delta_{e3} = \delta \tag{4.27}$$

In particular, equation 4.26 states that the pressure on both sides of the interface line must be the same, with P_{i2} being the pressure on the side of the inert gas and P_{e3} the one on the side of the detonation products. Equation 4.27 says that the flow direction of the inert gas along the slip-line must equal the one of the detonation products, with δ_{i2} being the flow angle of the inert gas, δ_{e3} the one of the detonation products, and δ the slip-line angle represented in figure 4.7.

Since, in this analysis, the detonation front is assumed to be planar and the fuel is moving towards the wave front with a speed equal to the CJ one, the pressure reached in the post-detonation condition, P_{e2}, will be equal to the CJ pressure. This value of pressure is present just in the very proximity of the detonation front since, after, the detonation products are subjects to an expansion process. This expansion process is assumed to be isentropic, and it is modelled through a Prandtl-Meyer expansion fan. Under those assumptions, total pressure is conserved, and the static pressure of the detonation products after the expansion fan, P_{e3}, can be evaluated from the post-detonation pressure P_{e2} with the following formula:

$$P_{e3} = P_{e2} \left(\frac{1 + \frac{\gamma_{e2}-1}{2}}{1 + \frac{\gamma_{e2}-1}{2} M_{e3}^2} \right)^{\frac{\gamma_{e2}}{\gamma_{e2}-1}} \tag{4.28}$$

With M_{e3} being the Mach number of the detonation products after the expansion fan and γ_{e2} the specific heat ratio of the detonation products, which is assumed to be the same both in the post-detonation zone and after the expansion fan. In particular, equation 4.28 is expressing the conservation of total pressure between the post-detonation zone and the end of the Prandtl-Meyer expansion fan, with the knowledge that, in the post-detonation zone, since the detonation is ideal, the Mach number is equal to 1.

According to the shock expansion theory, a flow crossing a Prandtl-Meyer expansion fan is turned by a certain angle δ, which can be computed with the following expression:

$$\delta = \nu(M_2) - \nu(M_1) \tag{4.29}$$

Where ν is the Prandtl-Meyer function, whose value depends on the Mach number of the flow, and point 1 represents the condition upstream of the expansion fan, while point 2 the one downstream.

The Prandtl-Meyer function has the following expression:

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}} tan^{-1}\left(\sqrt{\frac{\gamma-1}{\gamma+1}(M^2-1)}\right) - tan^{-1}\left(\sqrt{M^2-1}\right) \tag{4.30}$$

and, in the case of Mach=1, has a value equal to zero. Therefore, the flow angle of the detonation products after the expansion fan δ_{e3}, is related to the Mach number M_{e3} by the following equation:

$$\delta_{e3} = \sqrt{\frac{\gamma_{e2}+1}{\gamma_{e2}-1}} tan^{-1}\left(\sqrt{\frac{\gamma_{e2}-1}{\gamma_{e2}+1}(M_{e3}^2-1)}\right) - tan^{-1}\left(\sqrt{M_{e3}^2-1}\right) \tag{4.31}$$

Since, before the expansion fan, due to the CJ sonic condition, $\nu(M_{e2}) = 0$.

The pressure and the flow angle relative to the inert gas in correspondence with the slip-line are found by applying the oblique shock equations. In particular, the pressure jump across an oblique shock is expressed with the following equation:

$$\frac{P_{i2}}{P_1} = 1 + \frac{2\gamma_{i1}}{\gamma_{i1}+1}(M_{i1}^2 sin^2(\theta) - 1) \tag{4.32}$$

Where P_1 and P_{i2} are, respectively, the pressure of the inert gas before and after the oblique shock, γ_{i1} is the specific heat ratio of the inert gas (assumed to be the same before and after the oblique shock), M_{i1} is the Mach number of the inert gas before the oblique shock wave in the wave-fixed reference frame, and θ is the oblique shock angle.

Finally, the flow angle of the inert gas δ_{i2} along the interface line is computed using the following equation:

$$\delta_{i2} = tan^{-1}\left(2cot(\theta)\frac{M_{i1}^2 sin^2(\theta) - 1}{M_{i1}^2(\gamma_{i1} + cos(2\theta)) + 2}\right) \tag{4.33}$$

Which expresses the relation between the flow deflection angle after the crossing of the oblique shock and the angle of the oblique shock inclination.

By solving equations 4.26, 4.27, 4.28, 4.31, 4.32 and 4.33 simultaneously, the values of the unknowns P_{i2}, P_{e3}, δ_{i2}, δ_{i3}, M_{e2} and θ can be found.

Applying the analytical model developed by Sichel and Foster to the case of a rotating detonation combustor, with reference to the nomenclature of figure 4.5, the following equations are obtained:

$$P_3 = P_{2'} \tag{4.34}$$

$$\delta_3 = \delta_{2'} = \delta \tag{4.35}$$

$$P_3 = P_2 \left(\frac{1 + \frac{\gamma_2 - 1}{2}}{1 + \frac{\gamma_2 - 1}{2} M_3^2} \right)^{\frac{\gamma_2}{\gamma_2 - 1}} \tag{4.36}$$

$$\delta_3 = \sqrt{\frac{\gamma_2 + 1}{\gamma_2 - 1}} tan^{-1} \left(\sqrt{\frac{\gamma_2 - 1}{\gamma_2 + 1}(M_3^2 - 1)} \right) - tan^{-1} \left(\sqrt{M_3^2 - 1} \right) \tag{4.37}$$

$$\frac{P_{2'}}{P_{1'}} = 1 + \frac{2\gamma_{1'}}{\gamma_{1'} + 1}(M_{1'}^2 sin^2(\theta_{shock}) - 1) \tag{4.38}$$

$$\delta_{2'} = tan^{-1} \left(2cot(\theta_{shock}) \frac{M_{1'}^2 sin^2(\theta_{shock}) - 1}{M_{1'}^2(\gamma_{1'} + cos(2\theta_{shock})) + 2} \right) \tag{4.39}$$

The difference between the case for which the model was developed and a rotating detonation combustor consists in the fact that, in the latter case, the detonation is not bound by an inert gas, but from the detonation products of the previous cycle. In the case studied by Sichel and Foster, both the fuel-oxidizer mixture and the inert gas can be assumed to be at ambient condition, and, consequently, the value of M_{i1} can be evaluated as:

$$M_{i1} = \frac{C}{\sqrt{\gamma_{e1} R_{e1} T1}} \tag{4.40}$$

The same principle is not applicable to the case of a rotating detonation combustor, because the detonation products of the previous cycle present a temperature way much higher than the one of the fresh injected mixture. Since the value of the detonation products at the end of the cycle is not known a priori, the value of $M_{1'}$ consists of an additional unknown.

In order to evaluate it, the pressure of the detonation products, before the crossing of the oblique shock wave, is assumed to match the mixture injection pressure. The expansion from the post-detonation pressure P_2 to the injection one is assumed to be isentropic, and, consequently, the temperature of the detonation products at the end of the cycle can be evaluated as:

$$T_{1'} = T_2 \left(\frac{P_{1'}}{P_2} \right)^{\frac{\gamma_2 - 1}{\gamma_2}} \tag{4.41}$$

With T_2 being the CJ temperature.

Once the value of $T_{1'}$ is known, $M_{1'}$ can be evaluated, and equations from 4.34 to 4.39 can be solved in order to find the oblique shock angle and the slip-line angle.

At this point, all the quantities required for the application of the method of characteristics are known, and the solution of the flow field can start.

The method of characteristics consists of a mathematical technique used for solving partial differential equations. In particular, in the case of a two-dimensional, nonviscous, and irrotational flow, the equation of the flow motion has the following form (Liepmann and Roshko, 2001):

$$(u_x^2 - a^2)\frac{\partial u_x}{\partial x} + u_x u_y \left(\frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}\right) + (u_y^2 - a^2)\frac{\partial u_y}{\partial y} = 0 \tag{4.42}$$

$$\frac{\partial u_y}{\partial x} - \frac{\partial u_x}{\partial y} = 0 \tag{4.43}$$

Where x and y represents the two spatial coordinates, u_x the flow speed in the x direction, u_y the flow speed in the y direction, and a the speed of sound.

These equations present different solutions between the subsonic and supersonic cases. In particular, if:

$$\frac{u_x^2 + u_y^2}{a^2} < 1 \tag{4.44}$$

and, as a consequence, the flow is subsonic, equation 4.42 results in being of the elliptic type, and the application of the method of characteristics is not possible.

For supersonic cases, and, therefore, when

$$\frac{u_x^2 + u_y^2}{a^2} > 1 \tag{4.45}$$

equation 4.42 results in being hyperbolic, and a numerical solution with the method of characteristics can be found.

Therefore, the method of characteristics can be used to solve the flow field of a rotating detonation combustor only in the wave-fixed reference frame and only in the case of ideal detonation, where the detonation products have a speed exactly sonic. The two expansion fans present after the detonation front will then accelerate the flow from sonic to supersonic while, at the same time, reducing its temperature and pressure.

A hyperbolic equation is characterized by the presence of some particular directions, called characteristic lines. Along those characteristics lines, the dependent variables always satisfy a certain relation, which is called compatibility equation.

In the case of a supersonic flow with the previously made assumption, the characteristic lines present the two following directions:

$$\left(\frac{dy}{dx}\right)_{C_+} = tan(\theta + \mu) \tag{4.46}$$

$$\left(\frac{dy}{dx}\right)_{C_-} = tan(\theta - \mu) \tag{4.47}$$

Where θ represents the flow angle and μ is the Mach angle, which is a function of the flow local Mach number and is evaluated with the following expression:

$$\mu = sin^{-1}\left(\frac{1}{M}\right) \tag{4.48}$$

Therefore, for every point in the space, two characteristic lines can be distinguished, one with a positive and one with a negative slope. In particular, the positive-sloped characteristic line is called C_+, while the other one is referred to as the characteristic line C_-. A representation of the two characteristic lines of a generic point A is shown in figure 4.8

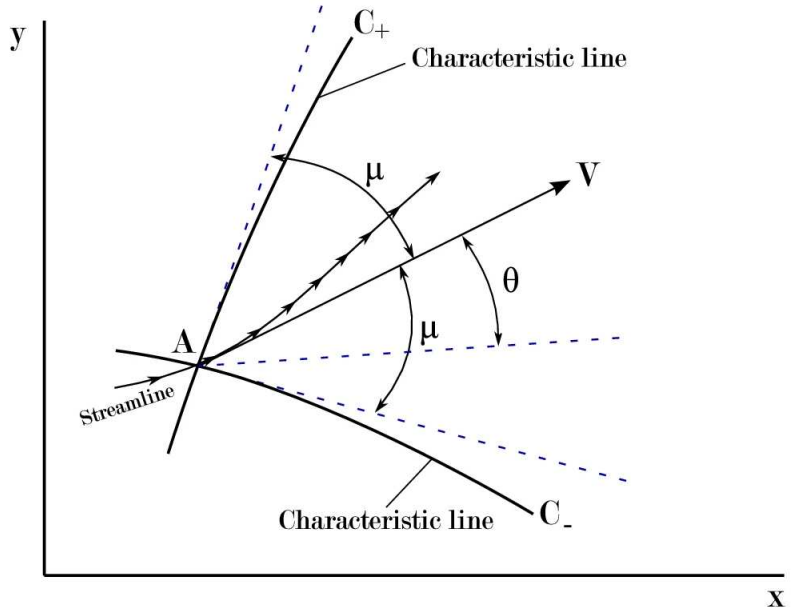

Figure 4.8: Representation of the two characteristic lines of a generic point A (Fernandes et al., 2023)

Regarding the compatibility equations, along a C_+ characteristic line, the following relation is always valid:

$$\theta - \nu(M) = K^+ \tag{4.49}$$

Where $\nu(M)$ is the value of the Prandtl-Meyer function, which depends on the local Mach number according to relation 4.30, and K^+ represents a constant value.

Along a C_- characteristic, the following compatibility equation is always satisfied:

$$\theta + \nu(M) = K^- \tag{4.50}$$

Where K^- is a constant value.

The method of characteristic computational method is based on the idea represented in figure 4.9

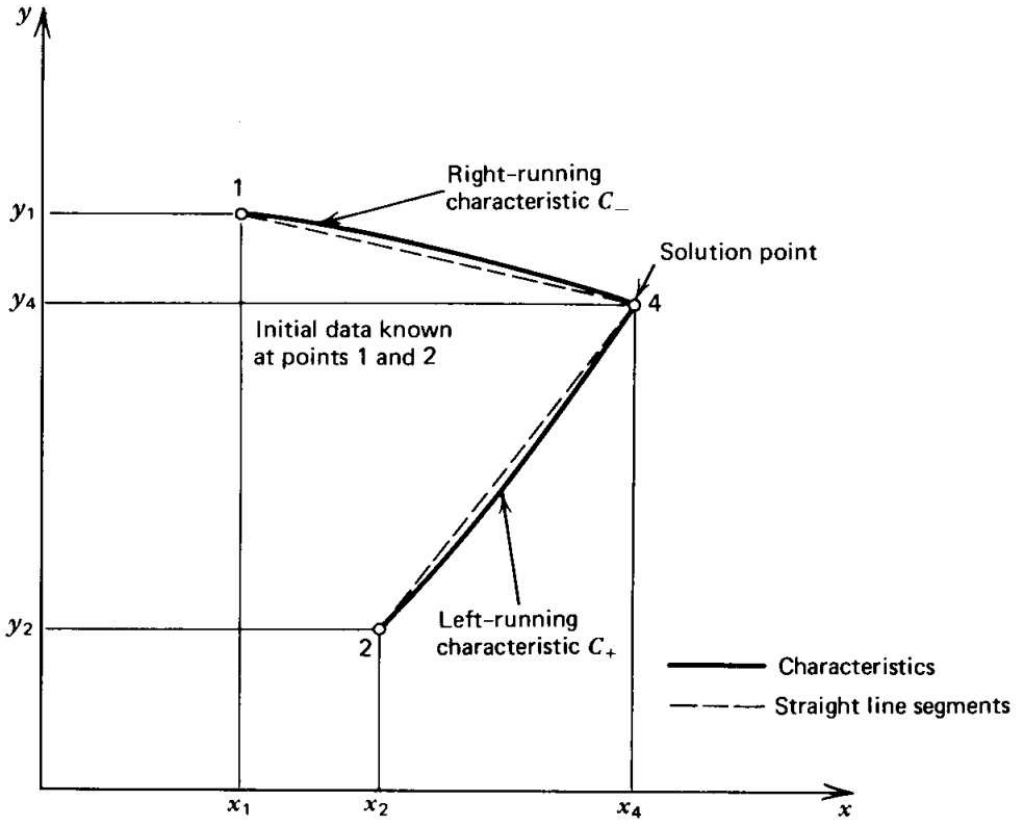

Figure 4.9: Representation of the characteristic network for three general points (Zucrow & Hoffman, 1977)

Starting from two points in which Mach number, spatial coordinates, and flow angle are known, with the method of characteristics it is possible to evaluate those quantities in a point correspondent to the intersection of a C_+ line of one of the known points with the C_- line of the other known one. With reference to figure 4.9, the known points are 1 and 2. It is possible to notice that points 1 and 4 lay on the same C_- characteristic; therefore, it is possible to say that:

$$\theta_1 + \nu_1 = \theta_4 + \nu_4 = K_1^- \tag{4.51}$$

At the same time, points 2 and 4 belong to the same C_+ characteristic line, and as a consequence:

$$\theta_2 - \nu_2 = \theta_4 - \nu_4 = K_2^+ \tag{4.52}$$

So, by solving equations 4.51 and 4.52, the flow angle and the Prandtl-Meyer function in point 4 are obtained as follows:

$$\theta_4 = \frac{K_1^- + K_2^+}{2} \tag{4.53}$$

$$\nu_4 = \frac{K_1^- - K_2^+}{2} \tag{4.54}$$

By reversing the Prandtl-Meyer function (equation 4.30), it is possible to obtain the value of the Mach number in point 4. Once this value is knonw, the Mach angle μ_4 can also be evaluated with equation 4.48.

Now, the spatial coordinates of point 4 need to be determined. It is important to mention that, as illustrated in figure 4.9, the characteristic lines are not perfectly straight. However, when the distance between the two known points and the unknown point is sufficiently small, these curves can be reasonably approximated using straight-line segments. This approximation, while practical for computational purposes, is responsible for introducing some degree of error in the solution. To mitigate this error and improve accuracy, the segments are computed by considering average values between the known points and the unknown point, as expressed by the following formulas:

$$\frac{y_4 - y_2}{x_4 - x_2} = tan\left(\frac{\theta_4 + \theta_2}{2} + \frac{\mu_4 + \mu_2}{2}\right) \tag{4.55}$$

$$\frac{y_4 - y_1}{x_4 - x_1} = tan\left(\frac{\theta_4 + \theta_1}{2} - \frac{\mu_4 + \mu_1}{2}\right) \tag{4.56}$$

By solving simultaneously equations 4.55 and 4.56, the spatial coordinates of point 4 are obtained. Then, once all the quantities of point 4 are known, the two characteristic lines C_+ and C_- can be drawn starting from point 4, which will then cross other characteristic lines from other known points, creating a characteristics net.

In the case of the rotating detonation combustor, the method of characteristics is initiated by the two Prandtl-Meyer expansion fans that take place just after the detonation front. In particular, in the wave-fixed reference frame, the upper one is turning the detonation products from their initial flow angle β to a final flow direction that will have the same inclination as the interface line, presenting an angle equal to $\beta + \delta$. Since backflow of the detonation products in the injection manifold is not considered in this model, in the part between the detonation front and the start of the mixture injection, the lower surface is assumed to behave like a wall. Therefore, the lower expansion fan is assumed to turn the flow from its initial angle β to a final flow direction of zero degrees, which corresponds to the horizontal slope of the wall of the lower surface.

For each expansion fan, the total turning angle is divided into n parts, such as:

$$\sum_{i=0}^{n} \delta_i = \delta \tag{4.57}$$

For each infinitesimal angle δ_i, the corresponding quantities necessary for the method of characteristics, such as the Prandtl-Meyer function, Mach number, and Mach angle, are evaluated. The Mach lines produced from the upper expansion fan are treated as C_- characteristic lines, and the ones from the lower expansion fan are treated as C_+ characteristic lines.

The characteristic network is then created, as shown in figure 4.10.

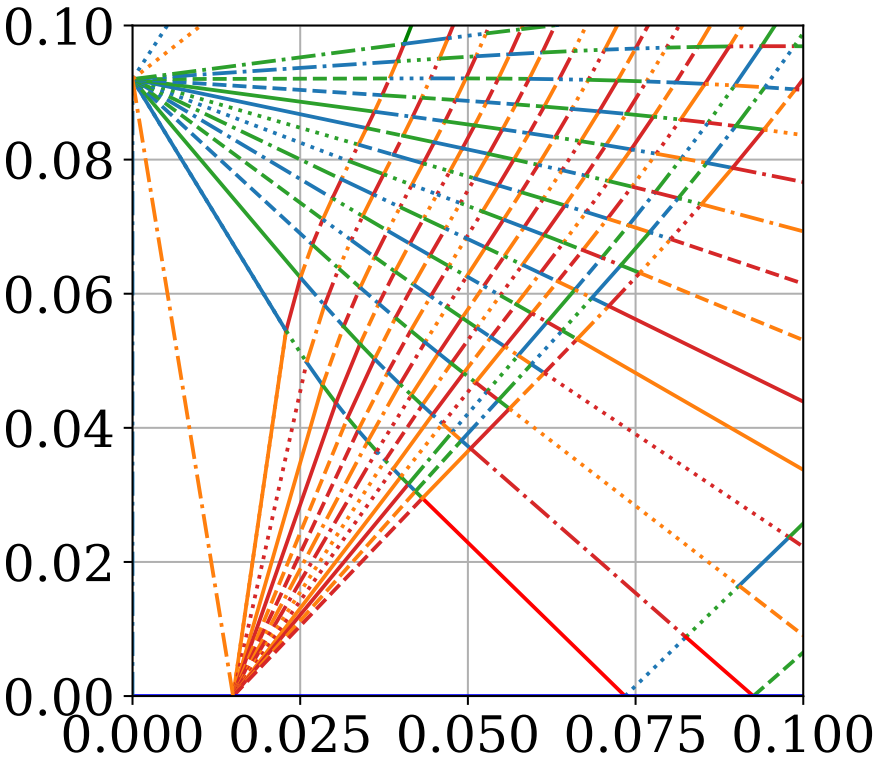

Figure 4.10: Representation of the characteristic network for the upper and lower expansion fans

In order to evaluate the pressure and temperature along the flow field, the total pressure and total temperature after the detonation front in the wave-fixed reference frame need to be computed. The total pressure is obtained from the following equation:

$$P_{t_2} = P_2 \left(1 + \frac{\gamma_2 - 1}{2}\right)^{\frac{\gamma_2}{\gamma_2 - 1}} \tag{4.58}$$

While the total temperature is evaluated with the following expression:

$$T_{t_2} = T_2 \left(1 + \frac{\gamma_2 - 1}{2}\right) \tag{4.59}$$

With the exception of the zone between the oblique shock line and the interface slip line, pressure and temperature are evaluated by assuming that the expansion of the detonation products is isentropic and, consequently, total properties are conserved. The pressure is therefore evaluated, in the points of intersection of two characteristic lines, with the following expression:

$$P = \frac{P_{t_2}}{\left(1 + \frac{\gamma-1}{2}M^2\right)^{\frac{\gamma}{\gamma-1}}} \tag{4.60}$$

Where M represents the local Mach number, obtained from the solution of the method of characteristics. Regarding the specific heat ratio γ, due to the significant change of temperature between the detonation front and the end of the cycle, an average value between those two points has been used.

In the same way, the temperature is computed as:

$$T = \frac{T_{t_2}}{1 + \frac{\gamma-1}{2}M^2} \tag{4.61}$$

Specific boundary conditions are imposed on the interface slip line, on the lower surface between the detonation front and the start of the mixture injection point and on the injection slip line.

On the interface line, the flow angle is imposed to be equal to the interface line inclination and, therefore, equal to $\beta + \delta$. Since the interface line is located in an upper position compared to the internal points of the flow field, it will intersect a C_+ characteristic line passing through the closest internal point, as schematically represented in figure 4.11.

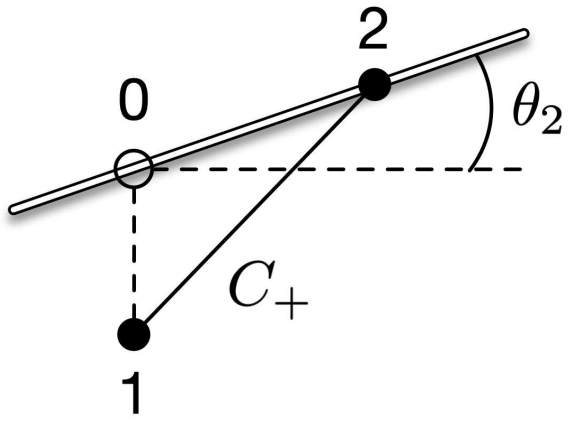

Figure 4.11: Schematic representation of a C_+ characteristic line in the case of an upper wall (Kyle Niemeyer, n.d.)

The interface slip-line is therefore treated as a wall, and the flow along it is assumed to have the same direction as the wall slope. In this case, point 1 is the known one, while point 2

has to be determined. By knowing that both points lay on the same C_+ characteristic and by knowing the flow angle in point 2, the value of the Prandtl-Meyer function in point 2 can be computed as:

$$\nu_2 = \theta_2 - \theta_1 + \nu_1 = \delta + \beta - \theta_1 + \nu_1 \tag{4.62}$$

Once the value of the Prandtl-Meyer function in point 2 is known, it is possible to compute the corresponding Mach number and, consequently, the associated Mach angle with equation 4.48. Then, spatial coordinates of point 2 can be computed with equations 4.55 and 4.56, as well as pressure and temperature, respectively, using equations 4.60 and 4.61. Once all the quantities of point 2 are known, a C_- characteristic line can be drawn from this point in order to proceed with the solution of the method of characteristics.

In an analogous way, the boundary condition on the lower surface is imposed. In this case, anyway, since no backflow in the injection manifold is considered, the flow is assumed to have an angle of zero degrees. The other difference with the previous case consists in the fact that this time the closest internal point and the point on the lower surface line will belong to the same C_- characteristic. Therefore, the value of the Prandtl-Meyer function on the lower wall is computed as:

$$\nu_2 = -\theta_2 + \theta_1 + \nu_1 = \theta_1 + \nu_1 \tag{4.63}$$

Also in this case, once the value of the Prandtl-Meyer function is known, all the other quantities, such as Mach number, Mach angle, spatial coordinates, temperature, and pressure, can be evaluated. After all the quantities are known, a C_+ characteristic line can be drawn, starting from this new point, in order to continue with the creation of the characteristic network.

In particular, this boundary condition is imposed on the lower surface until the value of the pressure, computed with equation 4.60, results in being higher than the mixture injection pressure P_{inj}. Once the pressure decays until this value, injection of the fresh mixture can begin. It is important, at this point, to save the horizontal spatial coordinates of the point where this happens, since this corresponds to the computed X_{ref}, which must later be compared with the estimated one from equation 4.25. In particular, since it is quite unlikely that the evaluated pressure on the lower surface will perfectly match the injection pressure P_{inj}, the value of X_{ref} is obtained by interpolation between the last point where $P > P_{inj}$ and the first one where $P < P_{inj}$.

The injection slip-line is assumed to be a straight line, and it is obtained by connecting the point where the refilling of the fresh mixture starts, X_{ref}, with the detonation front height of the next cycle, which is considered to be always equal to h.

Along this line, the boundary condition imposed is that the pressure of the flow is equal to the injection pressure. As in the case of the lower wall surface, the point on the line and the closest internal point will belong to the same C_- characteristic line. By imposing

the pressure to be equal to the injection one, by reversing equation 4.60, the value of the Mach number can be computed as:

$$M_2 = \sqrt{\frac{2}{\gamma - 1}\left(\left(\frac{P_{t_2}}{P_{inj}}\right)^{\frac{\gamma-1}{\gamma}} - 1\right)} \tag{4.64}$$

Once the Mach number is known, the value of the Prandtl-Meyer function can be evaluated with 4.30. After its evaluation, by knowing that the two points lay on the same C_- characteristics, the flow angle can be computed as:

$$\theta_2 = \theta_1 + \nu_1 - \nu_2 \tag{4.65}$$

With 2 being the point on the injection slip-line and 1 the closest point belonging to the internal flow field. Also in this case, all the missing quantities can be evaluated, and another C_+ characteristic line can be drawn starting from the point on the injection slip line.

Since, unlike what happens in other computational methods, in the method of characteristics the grid is not defined a priori but is constructed simultaneously with the solution procedure, it is really unlikely that the points found with the intersection of the characteristic lines will lie exactly on the combustor outlet line or combustor oblique shock line. Therefore, to evaluate the outlet conditions, as well as conditions on the oblique shock line, the solution of the flow field is continued also slightly outside of the combustor physical borders, as represented in figure 4.12.

In figure 4.12, point 1 represents the closest point of the characteristic network to the combustor outlet line. Point 2 also belongs to the characteristic network, but it is located outside of the combustor physical domain. Finally, point 3 does not belong to the characteristic network, but its position corresponds to the intersection of the C_+ characteristic line connecting points 1 and 2 and the combustor outlet line.

In order to evaluate pressure and temperature at the combustor outlet, the following procedure is adopted. Since they both belong to the characteristic net, all the quantities in points 1 and 2 are known. The slope of the C_+ line connecting the two points can be found from equation 4.56. At this point, by knowing the combustor axial length L, it is possible to find the spatial coordinate of point 3 by solving simultaneously the two following equations:

$$y_3 = L \tag{4.66}$$

$$y_3 = y_1 + Slope_{C_+}(x_3 - x_1) \tag{4.67}$$

Once y_3 and x_3 are known, the distance of point 3 with the other two points can be evaluated with the two following expressions:

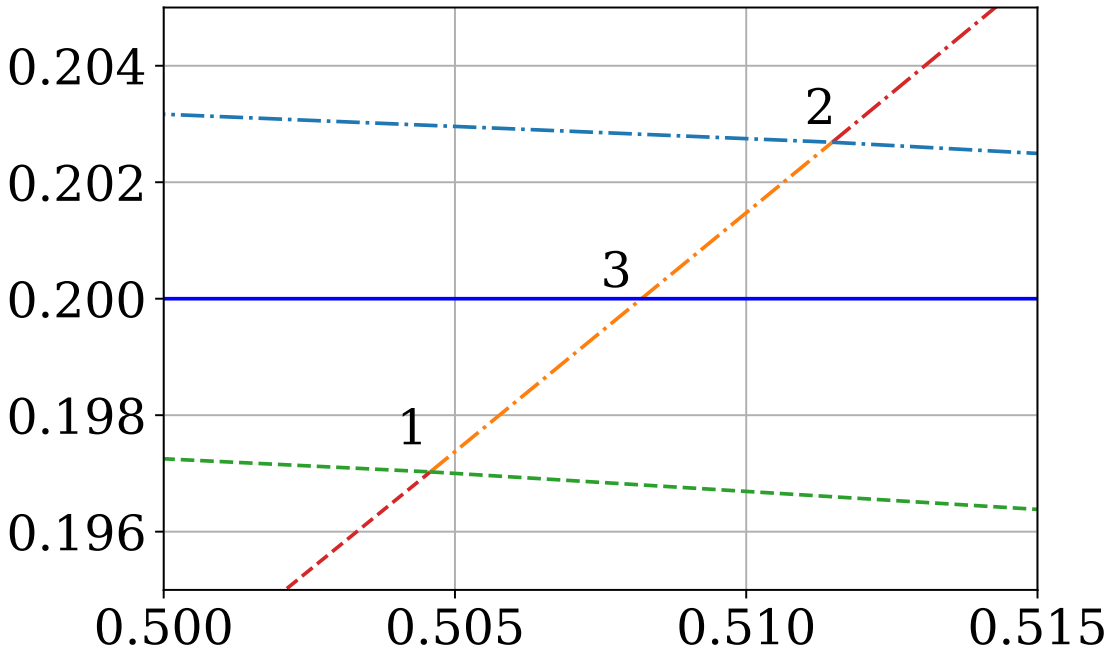

Figure 4.12: Representation of the characteristic net in a zone close to the combustor outlet line

$$D_{23} = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2} \tag{4.68}$$

$$D_{13} = \sqrt{(x_3 - x_1)^2 + (y_3 - y_1)^2} \tag{4.69}$$

Where D_{13} is the distance between point 1 and point 3, and D_{23} the one between point 2 and point 3.

By assuming the flow angle to change linearly between points 1 and 2, the flow angle in point 3 is evaluated as:

$$\theta_3 = \left(1 - \frac{D_{13}}{D_{12}}\right)\theta_1 + \left(1 - \frac{D_{23}}{D_{12}}\right)\theta_2 \tag{4.70}$$

Where D_{12} is the distance between points 1 and 2.

Then, since point 3 also belongs to a C_+ characteristic line, the value of the Prandtl-Meyer function in this point is computed as:

$$\nu_3 = \theta_3 - \theta_1 + \nu_1 \tag{4.71}$$

At this point, the Mach number can be computed by reversing the Prandtl-Meyer function, and temperature and pressure can be computed with equations 4.61 and 4.60.

The same procedure is applied to evaluate the points on the oblique shock line, with the difference that equation 4.66 is substituted with the equation of the oblique shock line:

$$y_3 = y_1 + tan(\theta_{shock} + \beta)(x_3 - x_1) \tag{4.72}$$

Then, considering all the points along the oblique shock line, an average value of the Mach number, pressure, and temperature obtained among them is evaluated. If the values are in agreement with the estimated ones, the solution of the model can proceed. Otherwise, another iteration is made, where the values of the average Mach number and pressure are used to solve equations from 4.34 to 4.39. Then, the flow field is solved again with the method of characteristics, but this time with the new values obtained for the oblique shock angle θ_{shock} and interface line angle δ . The process is repeated until the value of the Mach number and the pressure at the oblique shock line are in agreement with the one obtained in the previous iteration.

Once those conditions are satisfied, the solution procedure continues by confronting the obtained value of the fresh mixture injection starting point, X_{ref}, with the estimated one. If the two values are in agreement, it is possible to proceed with the next part. If not, based on the value of X_{ref} obtained from the solution of the method of characteristics, a new value of the injection angle β is evaluated as:

$$\beta = tan^{-1}\left(\frac{h}{2\pi R - X_{ref}}\right) \tag{4.73}$$

With h being the detonation front height and R the combustor radius.

Then, based on the new value of β, a new injection velocity is computed as:

$$V_{inj} = W_{cj} sin(\beta) \tag{4.74}$$

And the full process is started again and iterated until the predicted value of X_{ref} is in agreement with the one obtained from the method of characteristic solutions.

The final part of the model consists in evaluating the outlet condition in the zone between the oblique shock line and the interface line. It is in fact not possible to solve this part with the method of characteristics proposed above due to the entropy change introduced by the oblique shock, which is in contrast with the assumption of isentropic expansion. In order to evaluate the outlet conditions in this zone, the oblique shock equations are used. The value of Mach number, pressure, and temperature of the detonation products before the oblique shock are obtained from the solution of the method of characteristics, while the detonation products after the oblique shock are assumed to have a flow angle equal to the one of the interface line, as proposed in Sousa et al., 2017. Therefore, for each point, the pressure after the oblique shock is computed as:

$$P_{Post-Shock} = P\frac{2\gamma M^2 sin^2(s) - (\gamma - 1)}{\gamma + 1} \tag{4.75}$$

Where P and M are respectively the values of pressure and Mach number obtained from the method of characteristics, and s is the oblique shock angle relative to the flow angle before the oblique shock, which can be evaluated as:

$$s = \theta_{shock} + \beta - \theta \tag{4.76}$$

Where θ is the angle of the flow before crossing the oblique shock, and is obtained from the solution of the method of characteristics.

The temperature after the oblique shock is given by the following formula:

$$T_{Post-Shock} = T\frac{[2\gamma M^2 sin^2(s) - (\gamma - 1)][(\gamma - 1)M^2 sin^2(s) + 2]}{(\gamma + 1)^2 M^2 sin^2(s)} \tag{4.77}$$

And, finally, the Mach number after the oblique shock is evaluated from:

$$M^2_{Post-Shock} sin^2(s - a) = \frac{(\gamma - 1)M^2 sin^2(s) + 2}{2\gamma M^2 sin^2(s) - (\gamma - 1)} \tag{4.78}$$

Where a is the deflection angle, caused by the oblique shock, relative to the upstream flow direction. This angle is equal to:

$$a = \delta + \beta - \theta \tag{4.79}$$

4.4.2 Summary of the algorithm

In this section, the algorithm used is summarized. The model starts with acquiring the input parameters, which are total inlet pressure, total inlet temperature, detonation front height, combustor axial length and combustor radius. As an alternative to total inlet conditions, the static inlet conditions, if known, can also be used, with the difference that in the first case the system will stabilize at a certain value of injection velocity, which will then determine the value of the static injection conditions. In the latter case, the static injection condition will be held constant, and therefore, the value of the found injection velocity will determine the value of the total inlet condition corresponding to the stabilized system.

The model then proceeds with a first guess of the injection velocity, which, in case total conditions are used, sets the values of the corresponding inlet temperature and pressure. Once those two quantities are known, the Cj speed and the post-detonation conditions can be evaluated, and consequently, a first estimation of the point where the refilling of the fresh mixture starts, X_{ref}, can be performed.

For the first iteration, it is also necessary to estimate the values of temperature, pressure and Mach number before the oblique shock wave in order to perform the computation of the oblique shock angle and interface slip line angle. Those two angles are computed with an adaptation of the model proposed by Sichel and Foster, 1980.

After these angles are known, the flow field can be solved in the wave-fixed reference frame with the method of characteristics. Then, the results obtained are used to evaluate a new value of temperature, pressure and Mach number before the oblique shock line. If those results are in agreement with the one of the previous cycle, the solution can proceed. Otherwise, the newly obtained quantities are used for performing a new computation of the oblique shock angle and interface slip line angle.

Once this loop is solved, it is necessary to confront the estimated value of X_{ref} with the one obtained from the method of characteristics. If the two values are in agreement, the algorithm can continue with its last part, which consists of evaluating the outlet conditions in the zone between the oblique shock line and the interface line. Otherwise, based on the obtained value of X_{ref} from the method of characteristics, a new injection velocity has to be computed and used to restart the whole process. A schematic representation of the algorithm flow chart is shown in figure 4.13.

4.4.3 Advantages and limitations of the model

The presented model has the main advantage that, compared to a computational fluid dynamic (CFD) simulation, it can provide a quite accurate solution of the flow field and of the combustor outlet conditions in a much shorter computational time. In fact, while a CFD simulation can take several hours to provide the results, this algorithm just needs a couple of minutes with a standard computer. This fact makes that model perfectly suitable for thermodynamic cycle analysis, where lots of different inlet conditions have to be studied in order to maximize the cycle efficiency.

At the same time, this model presents the following limitation:

- The method of characteristics, with the assumption of straight characteristic lines, intrinsically introduces some errors in the flow field solution
- Since the method of the characteristics is valid only for supersonic flow field, it can be used only in the case of an ideal detonation, where the detonation products presents a sonic speed in the wave-fixed reference frame, which is then accelerated by the expansion fans
- Injection losses are neglected
- Backflow of the detonation products in the manifold is not considered
- Deflagration along the injection slip-line is not modeled, assuming all the mixture to burn by detonation
- The present model is suited for only one detonation wave per cycle, but adaptation with more than one wave is possible

4.5 Validation of the models

In this section, a comparison between the presented 2D model and the results from Sousa et al., 2017, on which the model is based, is firstly presented. Then, a comparison of the

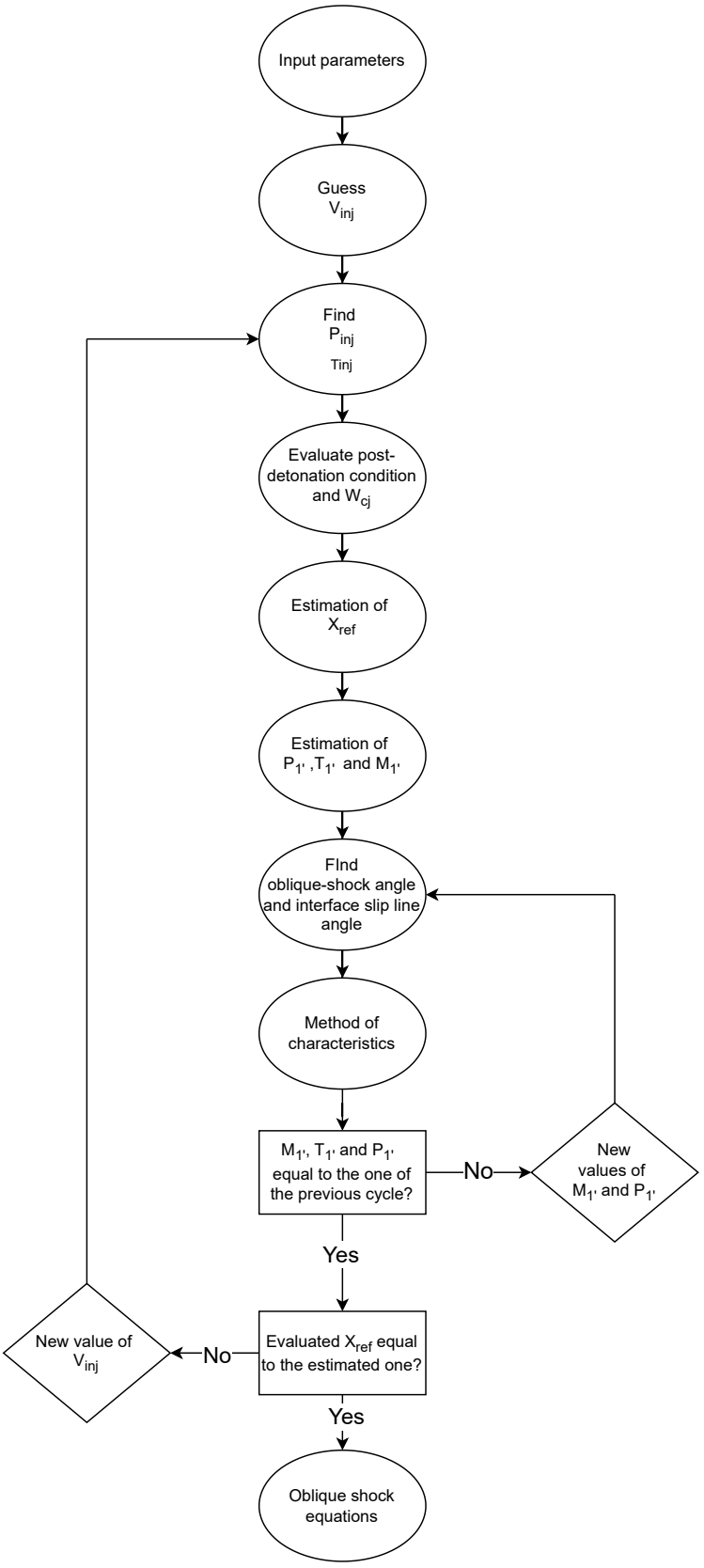

Figure 4.13: Schematic representation of the algorithm flow chart

total outlet pressure and temperature profiles estimated with the two models is confronted with results from the computational fluid dynamic (CFD) simulation performed also by Sousa et al., 2017.

Figure 4.14 represents the characteristic network obtained by Sousa et al., 2017, while figure 4.15 shows the final characteristics network obtained from the 2D model presented in this thesis.

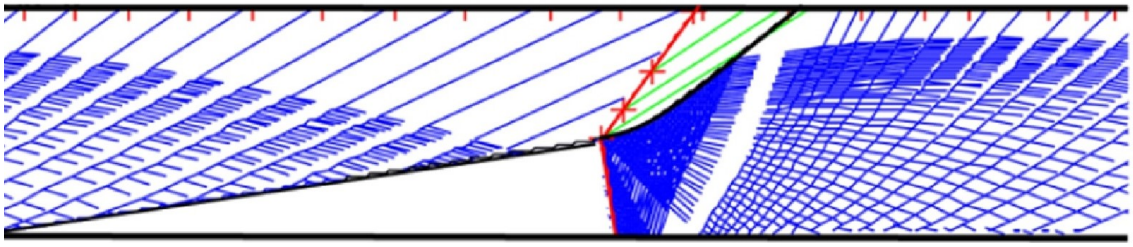

Figure 4.14: Characteristic network obtained from (Sousa et al., 2017)

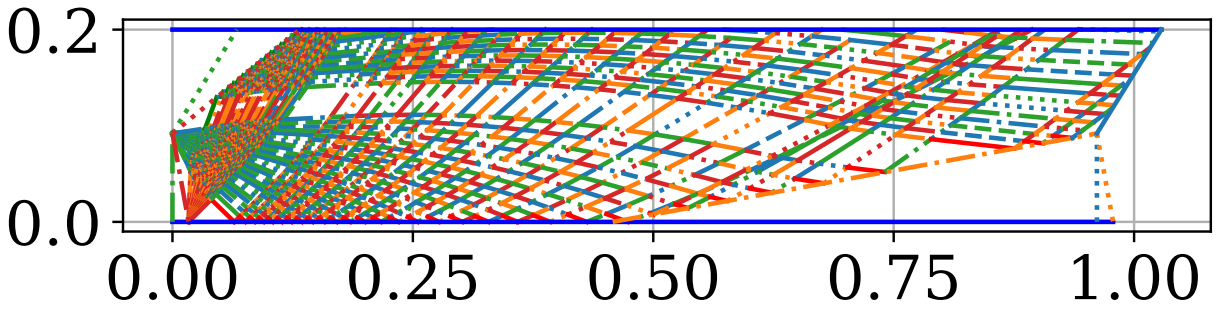

Figure 4.15: Characteristic network resulting from the 2D model presented in section 4.4

As it is possible to notice, one main difference is observable in the zone above the injection slip-line. In this area, in figure 4.14, the characteristic lines are showing a divergent behavior that, according to the shock expansion theory, symbolizes an expansion of the flow. On the other hand, in figure 4.15, the lines tend to converge, which is a sign that the flow is undergoing a compression. This phenomenon derives from the imposition of the injection pressure along the injection slip-line, which, as a consequence, prevents the detonation products from further expanding, showing the same behaviour as a compression.

Results obtained from the two presented combustor models have been compared with the CFD simulation performed in Sousa et al., 2017.

In the CFD simulation, a hydrogen-air stoichiometric mixture is injected in the combustor at a total pressure of 8 bar and a static temperature of 293 K. The combustor presents a radius equal to 0.153 m and an axial length of 0.2 m. The CFD solution has been performed with the software OpenFoam, using a two-dimensional unsteady Reynolds averaged Navier–Stokes solver. In addition to that, to simulate deflagration-to-detonation transitions of hydrogen-air, the solver ddtFoam, developed at TU Munich by Ettner, has been

used. In particular, this code uses a second-order accurate Harten-Lax-Van Leer-Contact Riemann Solver, and the turbulent closure was achieved with the k-omega-SST model. The rotating detonation combustion test case was performed on a 2D cylindrical numerical domain with a uniform mesh spacing of 1 mm in x, y and z directions, resulting in a computational domain of 1.2 million cells. The Courant– Friedrichs–Lewy was kept below 0.3 for stability reasons, which resulted in a time step of approximately $5 \times 10^{-8} s$. When the pressure close to the inlet exceeds the total injection pressure, the solver switched to supersonic outlet conditions allowing for reverse flow into the inlet. Periodic detonation was achieved after 15 detonation cycles.

The same geometrical properties have been used as input parameters for the two models, together with a total inlet pressure of 8 bar and a total inlet temperature of 330 K. The detonation front height has been set to the same value obtained from the CFD simulation, which corresponds to 92 mm.

In Sousa et al., 2017 the results reported are the total outlet pressure and total outlet temperature, both relative to the laboratory reference system. It is therefore necessary to convert the static outlet values obtained from the two models to the total ones.

In the case of the 1D model, the process is straightforward, since the value of the outlet flow velocity in the laboratory reference frame is already a known value. Therefore, referring to the nomenclature used in section 4.3, and assuming the detonation products to behave as an ideal gas, the value of the total pressure at the combustor outlet is evaluated as:

$$P_{t_{1D}} = P_e \left(1 + \frac{\gamma_2 - 1}{2} \frac{u_{ea}^2}{\gamma_2 R_2 T_e}\right)^{\frac{\gamma_2}{\gamma_2 - 1}} \tag{4.80}$$

Always using the isentropic flow equations, the value of the total outlet temperature of the 1D model is given by:

$$T_{t_{1D}} = T_e \left(1 + \frac{\gamma_2 - 1}{2} \frac{u_{ea}^2}{\gamma_2 R_2 T_e}\right) \tag{4.81}$$

Regarding the 2D model, the outlet conditions are provided by the solution of the method of characteristics, which is relative to the wave-fixed reference frame. Therefore, it is necessary to convert the obtained data in the laboratory reference frame.

The first step consists of evaluating the speed of sound at every point of the outlet. For this purpose, the function `soundspeed_fr` from the Shock and Detonation ToolBox (Browne et al., 2023) has been used. In particular, this function requires as an input a Cantera (Goodwin et al., 2018) gas object, and provides the corresponding speed of sound as an output. For each point of the outlet, a cantera gas object has been created using the `gri30_highT.yaml` model, and initialized with the values of pressure and temperature relative to the point taken in consideration. The chemical composition of the detonation products is assumed to be fixed for all the expansion process, and therefore, for each point, has been set equal to the one of the Cantera gas object resultant from the `PostShock_eq` function.

A simple alternative can consist of assuming ideal gas behavior at the combustor outlet and computing the sound speed with the classic formula:

$$c = \sqrt{\gamma RT} \tag{4.82}$$

Where T has to be substituted with the value of the temperature of the considered outlet point obtained from the method of characteristics.

Once the sound speed in every point has been evaluated, the flow speed in the wave-fixed reference frame can be computed by using the Mach number definition as:

$$W_i = M_i c_i \tag{4.83}$$

Where M_i is the Mach number, obtained from the solution of the method of characteristics, of a general outlet point i, and c_i its sound speed.

In order to pass from one reference system to the other one, it is necessary to add or subtract vectorially the speed of the detonation wave U. The latter is assumed to be constant, and, according to figure 4.5, can be evaluated as:

$$U = W_{cj} cos(\beta) \tag{4.84}$$

Since U is purely tangential, in each outlet point, the flow angle obtained from the method of characteristics, θ_i, represents the angle between the detonation wave speed U and the flow speed in the wave-fixed reference frame W_i. Therefore, the speed in the laboratory reference system is evaluated as:

$$V_i = \sqrt{V_x^2 + V_y^2} \tag{4.85}$$

Where

$$V_x = |W_i cos(\theta_i) - U| \tag{4.86}$$

And

$$V_y = W_i sin(\theta_i) \tag{4.87}$$

Finally, the Mach number in the laboratory reference system can be computed as:

$$M_{i_{lab}} = \frac{V_i}{c_i} \tag{4.88}$$

and total pressure and temperature are evaluated with the isentropic flow equations:

$$P_{t_{2D_i}} = P_i \left(1 + \frac{\gamma_2 - 1}{2} M_{i_{lab}}^2\right)^{\frac{\gamma_2}{\gamma_2 - 1}} \tag{4.89}$$

$$T_{t_{2D_i}} = T_i \left(1 + \frac{\gamma_2 - 1}{2} M_{i_{lab}}^2\right) \tag{4.90}$$

In figure 4.16, the comparison between the total outlet pressure profiles of the 1D model, 2D model and CFD simulation is shown. Figure 4.17 is representing instead the three total temperature profiles.

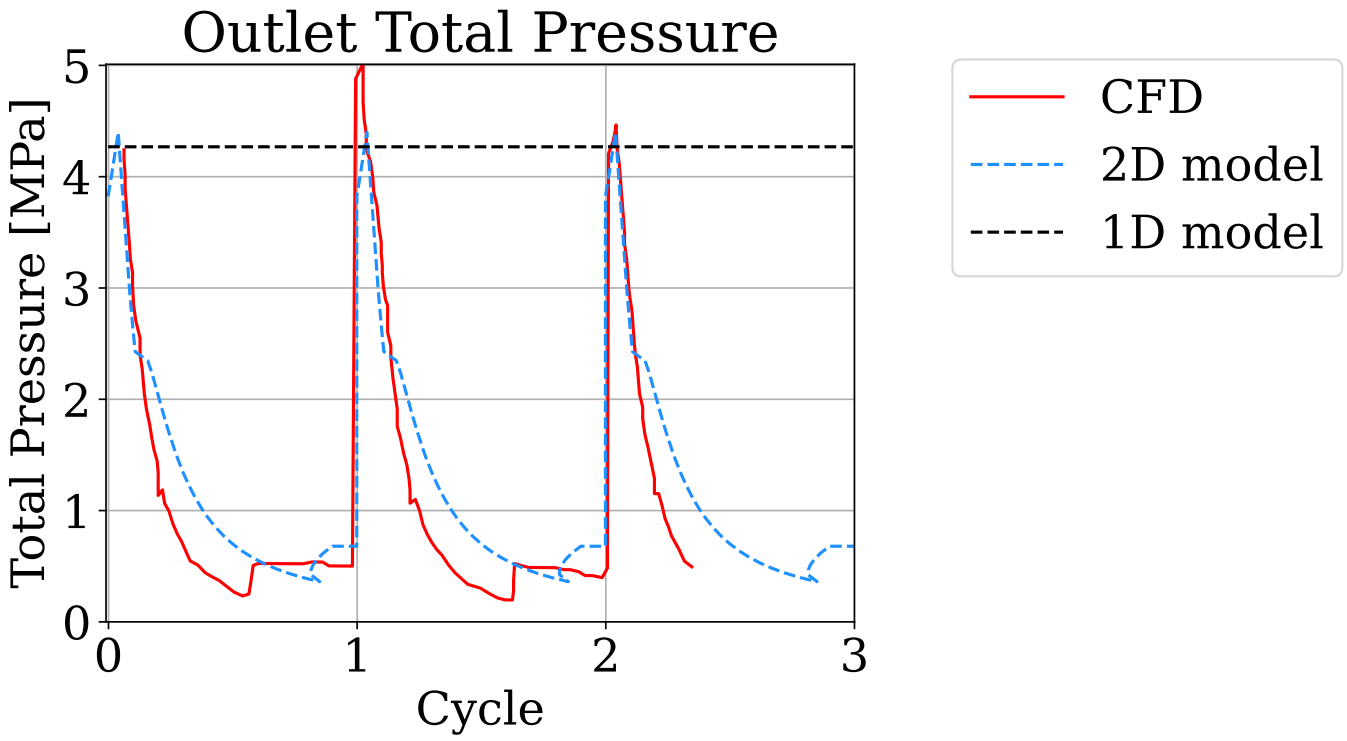

Figure 4.16: Comparison of the total outlet pressure profiles obtained from the two models and the CFD simulation of Sousa et al., 2017

As expected, the 1D model only returns one value of total temperature and pressure, and is therefore not able to reproduce the quantity oscillation along the azimuthal coordinate. Anyway, as it is possible to observe, the single provided value clearly overestimates the outlet pressure, but it is representative of the average combustor outlet temperature.

On the other hand, the 2D model is able to represent the spatial variation. In particular, it is perfectly suitable to reproduce the peak values as well as the physical behaviour of the outlet flow, such as the quantity decay and the small 'step' increase towards the end of the cycle. Anyway, it is clear that the decay predicted by the model results in being way much slower than the one shown in the CFD simulation. As a consequence of this, also the flow 'recompression' at the end of the cycle appears to have a certain delay with respect to the one predicted by the CFD solution. In fact, this phenomenon is a consequence of the fresh mixture injection that, as explained before, prevent the detonation products from further

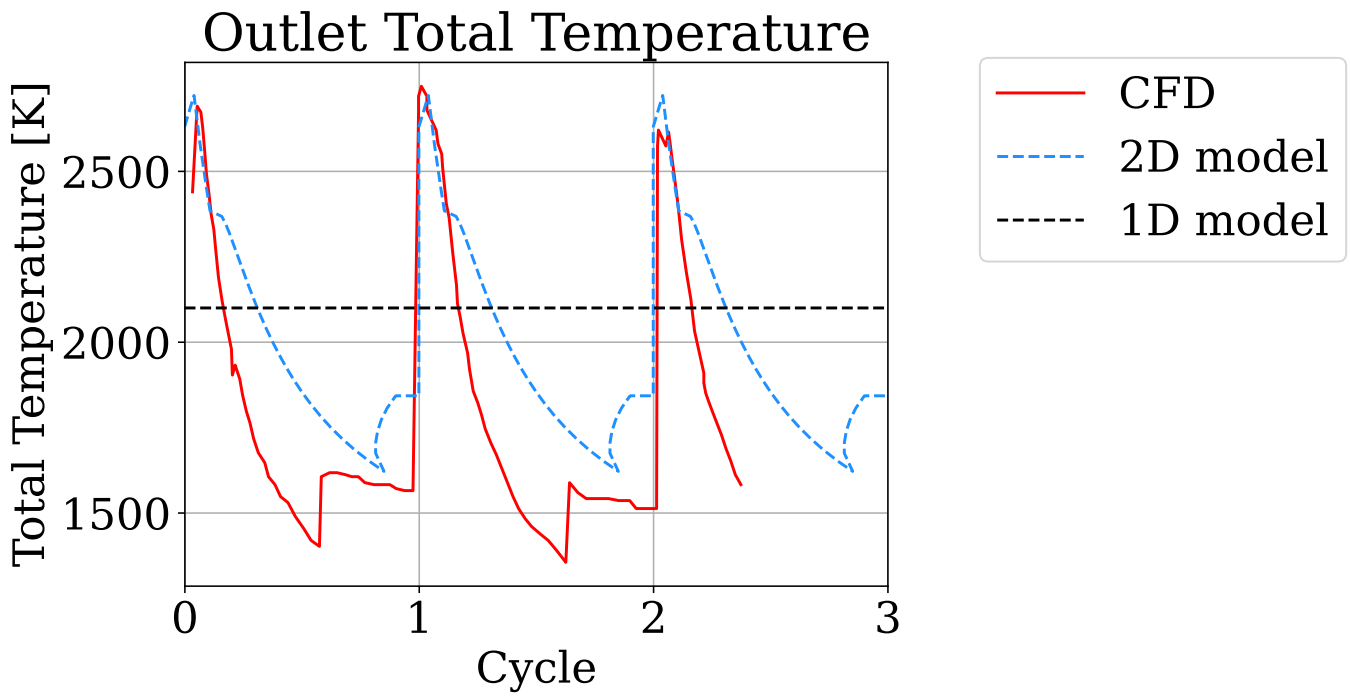

Figure 4.17: Comparison of the total outlet temperature profiles obtained from the two models and the CFD simulation of Sousa et al., 2017

expanding. Since the decay of pressure is faster in the CFD simulation, the detonation products are expected to match the injection pressure earlier, and, as a consequence, the fresh mixture injection will start sooner. A possible reason to explain the slow decay of the model could be the hypothesis of isentropic expansion from the post-detonation front.

Also, while for the total pressure the 2D model seems able to quite accurately reproduce the CFD profile, it is clearly less precise with the reproduction of temperature. The same phenomenon is also observable in the original version of the model, (Sousa et al., 2017), whose results are shown in figure 4.18.

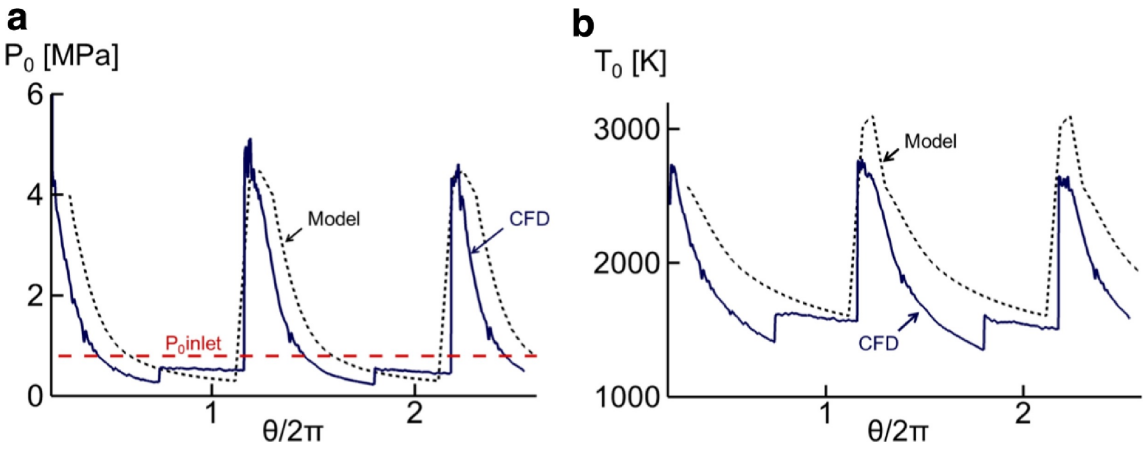

Figure 4.18: Comparison between the result obtained in Sousa et al., 2017 and the CFD simulation of Sousa et al., 2017

Of course, both the presented 2D model and the original model are compared to the same CFD simulation. Also, as expected from the comparison of figures 4.14 and 4.15, it is possible to notice that, in figure 4.18, the final 'recompression' of the flow is not present.

While, as it is shown in figure 4.17, the 2D model overestimates the outlet temperature, the values provided from the 1D model result in being very close to the arithmetical average (which, as it will be explained later, is lower than the mass flux average) of the CFD results, and as a consequence, it may underestimate the combustor outlet temperature. On the other hand, the 1D model clearly overestimates the combustor outlet pressure, while it is quite well approximated from the 2D model.

In conclusion, it is important to remember that the 1D model takes only a few seconds to provide the results, and the 2D model usually takes around 2 minutes. Nevertheless, they have proven to provide comparable results with a CFD simulation, which usually takes several hours to be completed. Therefore, the two presented models have been shown to be totally suitable if a quick estimation of the combustor outlet condition is needed, such as in the context of thermodynamic cycle analysis.

As a consequence, the two models will be used to perform the efficiency evaluation of the two proposed power plant configurations 3.1 and 3.2.

5. Results

5.1 Comparison of the two models

In this section, a comparison between the outlet conditions predicted by the 1D model and by the 2D model is shown.

The same parameters described in the validation section 4.5 have been used to perform the models comparison. In figure 5.1, the static pressure profile obtained at the combustor outlet using the 2D model is shown. Figure 5.2 represents instead the outlet static temperature.

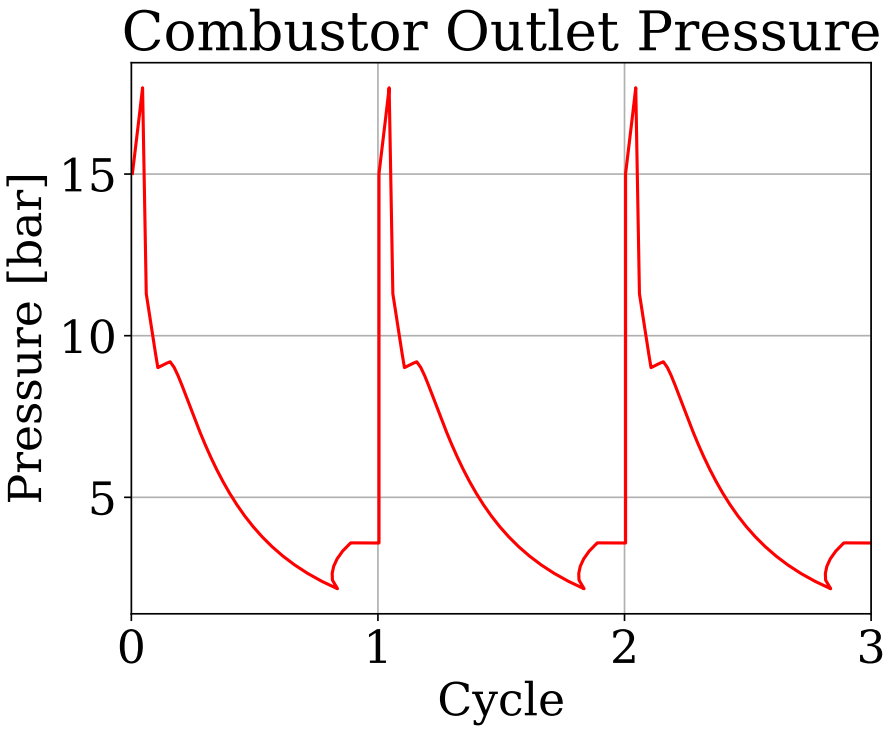

Figure 5.1: Combustor outlet pressure profile obtained from the 2D model

Anyway, in the thermodynamic cycle analysis of the power plants represented in figure 3.1 and figure 3.2, the turbine is treated with the standard thermodynamic approach, and, as a consequence, the thermodynamic state corresponding to the turbine inlet point is characterized by a single value of temperature and pressure. Therefore, from the 2D model, the value of the mass flux averaged temperature and pressure are extracted and used to determine the turbine inlet thermodynamic state.

In particular, the mass flux averaged temperature is computed with the following equation:

$$P_{avg} = \frac{\sum_{i=0}^{n} \rho_i V_{y_i} P_i}{\sum_{i=0}^{n} \rho_i V_{y_i}} \tag{5.1}$$

Where the pedix i is relative to the considered outlet point, n is the total number of outlet

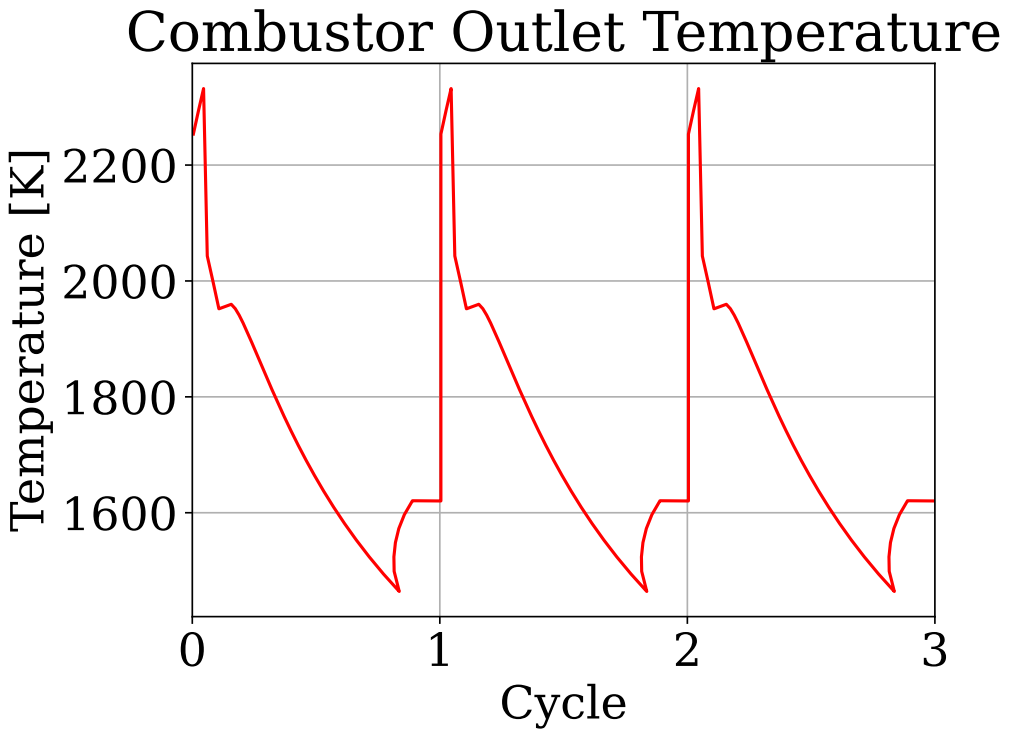

Figure 5.2: Combustor outlet temperature profile obtained from the 2D model

points, ρ_i is the density of the considered point i, V_{y_i} its axial velocity in the laboratory reference system and P_i its pressure.

The density ρ_i can be evaluated using a Cantera gas object initialized with the known value of temperature and pressure, with a procedure analogous to the one described in section 4.5 regarding the speed of sound. As an alternative, the equation of state of ideal gases can be used. The procedure to evaluate V_{y_i} is also explained in section 4.5.

In the same way, the mass flux averaged temperature is evaluated as:

$$T_{avg} = \frac{\sum_{i=0}^{n} \rho_i V_{y_i} T_i}{\sum_{i=0}^{n} \rho_i V_{y_i}} \tag{5.2}$$

Regarding the 1D model, the single returned values of pressure and temperature will be directly used to characterize the turbine inlet point.

Given the same initial conditions as the ones described in section 4.5, the two models have provided the results reported in table 5.1.

	1D model	2D model
Outlet temperature [K]	1737.17	1999.58
Outlet pressure [bar]	17.06	10.29

Table 5.1: Comparison of temperature and pressure provided from the two models with the initial conditions reported in 4.5

As expected, since also the axial flow speed and the outlet density in the 2D model present a similar profile to the one of pressure and temperature reported in figures 5.1 and 5.2, the mass flux averaged values, reported in the 2D model, result in being higher than the arithmetical values. Also, the 1D model returns values relative to the turbine inlet temperature that are lower than the one obtained from the 2D model, but shows higher values of combustor outlet pressure.

5.2 Power Plant A

In this section, the results obtained from the efficiency evaluation of the power plant A, shown in figure 3.1 are reported. Also, the efficiency of the power plant has been compared to that of a standard ideal Joule Brayton cycle under different circumstances.

In this analysis, the detonation front height has been estimated using the lower limit of the Bykowskii empirical correlation, which is:

$$h = 7\lambda = 0.1057 \quad m \tag{5.3}$$

With λ being the detonation cell width, that, for a hydrogen-air stoichiometric mixture, is 15.1 mm.

The combustor radius has been computed starting from another empirical correlation found always from F. A. Bykovskii, Zhdan, and Vedernikov, 2006. According to this relation, to ensure a stable operation of the rotating detonation combustor, the ratio between the detonation front height and the combustor circumferential length must be inside the range 0.10-0.18. Also in this case, the lower limit has been chosen, and the combustor radius has been computed as:

$$R = \frac{h}{0.1 \times 2\pi} \simeq 0.168 \quad m \tag{5.4}$$

In the case of the 2D model, the combustor axial length has been set equal to 0.2 m.

The compressor is treated as ideal, with a value of isentropic efficiency equal to 1. The mixture is assumed to be premixed, and enters the compressor at a pressure of 1 bar and a temperature of 300 K. The specific work of the compressor has been computed as:

$$W_{compressor} = h_2 - h_1 \tag{5.5}$$

With h being the enthalpy of the flow in point 1 and 2 of figure 3.1.

The thermodynamic properties of the various points have been evaluated by using Cantera (Goodwin et al., 2018).

The study has been performed at different values of compressor pressure ratio ($\frac{P_2}{P_1}$ of figure 3.1), ranging from 1 to 10. To estimate the rotating detonation combustor outlet

conditions, the two models previously described have been used. In case of the 1D model, the values obtained have been directly used to characterize the turbine inlet point, while in the case of the 2D model, the mass flux averaged values of the obtained quantities have been used.

The turbine is also treated as ideal, with unitary isentropic efficiency, and the specific work extracted from it has been evaluated as:

$$W_{turbine} = h_3 - h_4 \tag{5.6}$$

With h being the enthalpy in points 3 and 4, where in point 4 the flow has been expanded until the ambient pressure of 1 bar is reached.

Regarding the Joule-Brayton cycle used for comparison, its efficiency is evaluated at the same value of pressure ratio used for power plant A, and the constant pressure combustion has been performed using the function `equilibrate('HP')` by Cantera.

Both for the RDE and for the Joule-Brayton cycle, the efficiency has been evaluated as:

$$\eta = \frac{(\dot{m}_{air} + \dot{m}_{H2})(W_{turbine} - W_{compressor})}{\dot{m}_{H2} LHV_{H2}} \tag{5.7}$$

With LHV being the lower heating value of hydrogen and $\dot{m}$ the mass flow rate.

5.2.1 Comparison at the same initial conditions

The comparison of the efficiency curve as a function of the compressor pressure ratio between power plant A and a standard Joule-Brayton cycle is shown in figure 5.3. In this case, for both cycles, the same amount of mass flow rate has been used, and, in both cases, air and hydrogen are in a stoichiometric proportion.

5.2.2 Comparison at the same net power

In this section, the comparison between power plant A and a Joule-Brayton cycle has been performed at the same level of net power produced. The net power produced by the cycle corresponds to the numerator of equation 5.7. Also, while for a rotating detonation engine a stoichiometric mixture of air and fuel may represent a suitable working condition, for a Joule-Brayton cycle excess air is always used. Therefore, in this comparison, power plant A is runned with a stoichiometric air-to-fuel mass ratio, while for the Joule-Brayton cycle this value has been set equal to 70. Results of this comparison are reported in figure 5.4.

5.2.3 Comparison at the same turbine inlet temperature and initial conditions

The previous analyses have been performed only from a thermodynamic cycle efficiency point of view, without considering any possible external limitation in the cycle.

Anyway, nowadays, one of the main obstacles that is preventing efficiency enhancement in power cycles is related to the limit temperature at the turbine inlet. In fact, if the

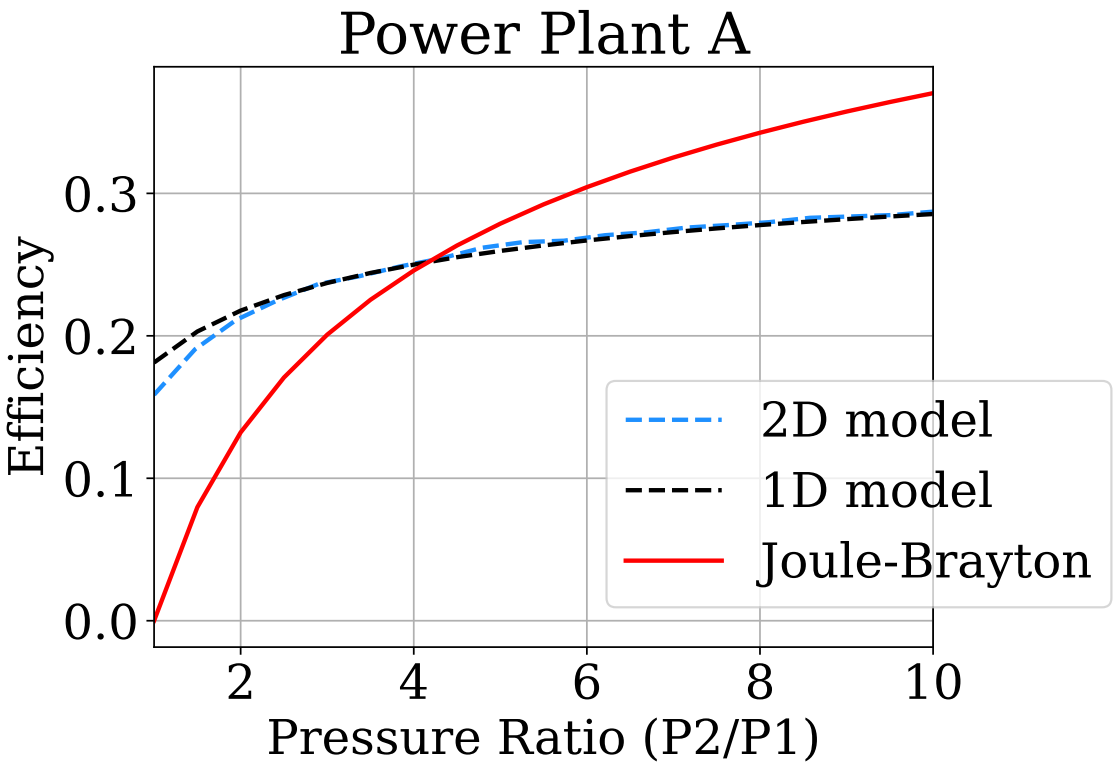

Figure 5.3: Efficiency of power plant A as a function of the compressor pressure ratio, compared with a Joule-Brayton cycle at the same initial conditions

temperature of the flow is higher than the limit temperature of the turbine material, the risk of melting the turbine blades may occur, and, as a consequence, this is something that in the design phase of a power plant must be totally avoided.

Also, since the power plant in question is working with a stoichiometric hydrogen-air mixture, the values of the turbine inlet temperature are expected to be higher than the turbine inlet limiting temperature.

Therefore, this time the two cycles have been compared at the same amount of mass flow rate, imposing a limitation on the turbine inlet temperature. In both cases, an initial stoichiometric mixture between air and fuel has been used.

In particular, in case the flow in point 3 of figure 3.1 shows a temperature lower than the limiting one, nothing changes, and the flow is expanded in the same way as previously done. In the case of a temperature higher than the limiting one, the flow is assumed to be cooled isobarically until the value of 1500 K is reached.

Results of the comparison of the two cycles under these conditions are reported in figure 5.5.

5.2.4 Comparison at the same turbine inlet temperature at different initial conditions

In this section, the comparison between power plant A and a Joule-Brayton cycle has been performed at the same turbine inlet temperature. Anyway, differently from the previous

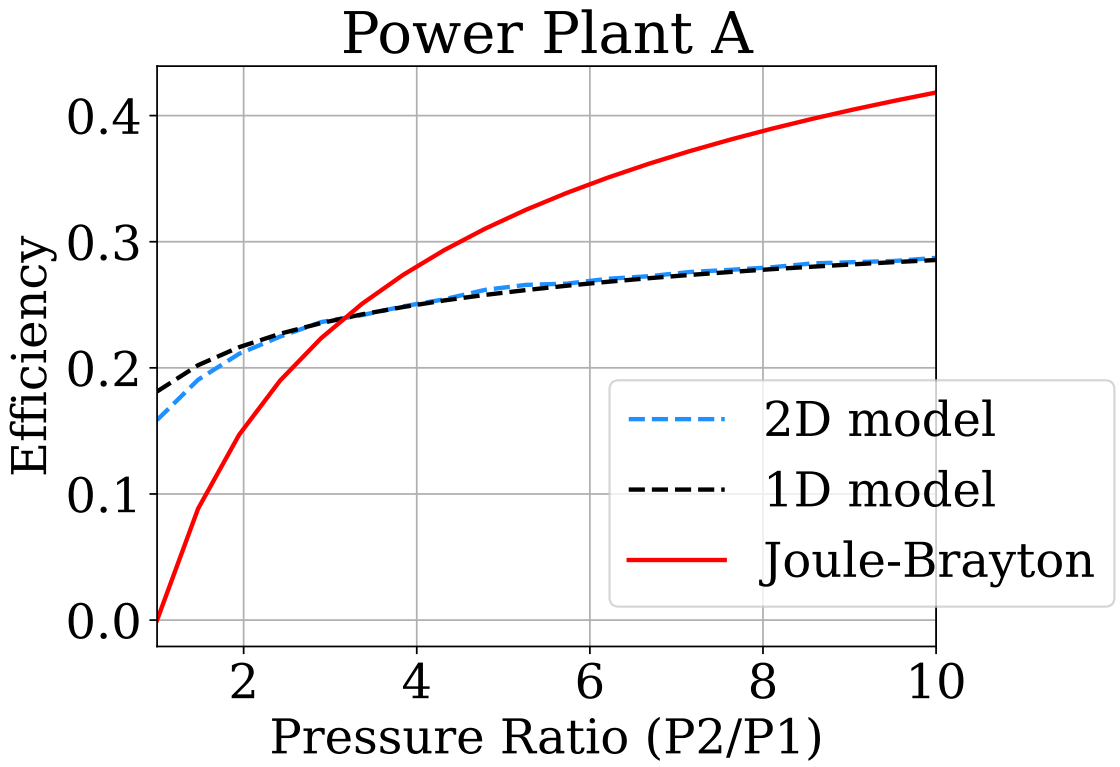

Figure 5.4: Efficiency of power plant A (stoichiometric mixture) as a function of the compressor pressure ratio, compared with a Joule-Brayton cycle (air-to-fuel mass ratio equal to 70) at the same value of net power produced

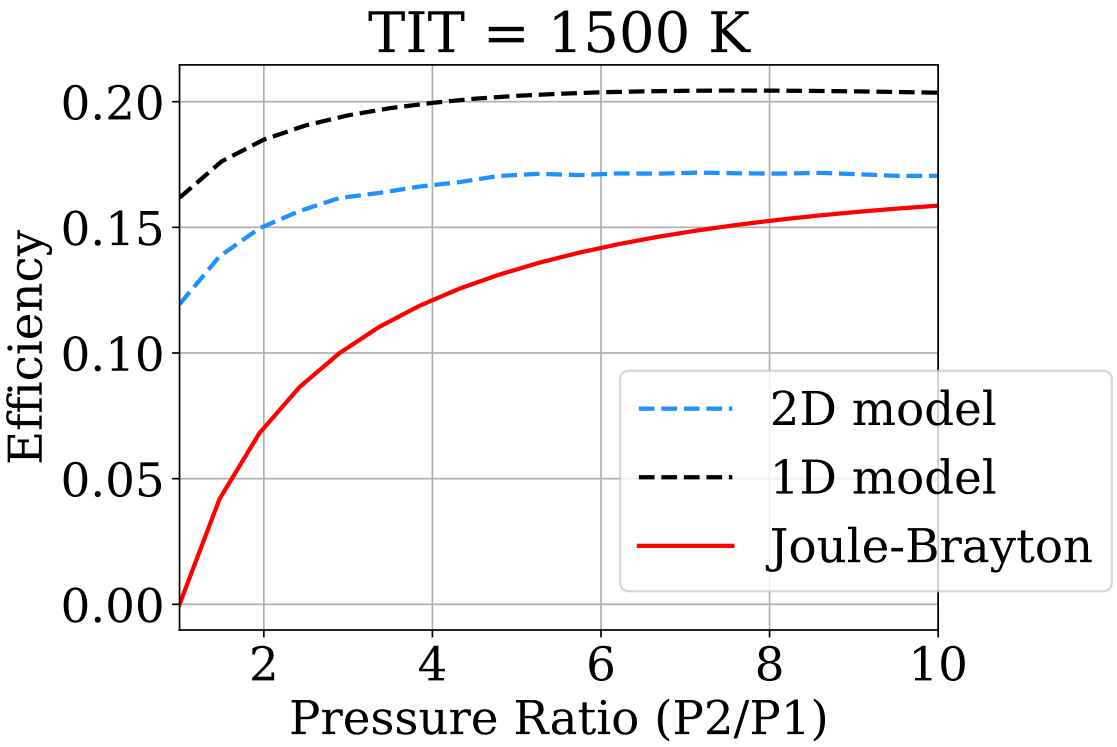

Figure 5.5: Efficiency of power plant A as a function of the compressor pressure ratio, compared with a Joule-Brayton cycle, in case of a limiting turbine inlet temperature of 1500 K

case, the desired turbine inlet temperature is not reached by an isobaric cooling of the flow but with the addition of air. Since also the added air will then expand in the turbine, this results in a more efficient way of cooling the stream. In particular, regarding the Joule-Brayton cycle, for every value of compressor pressure ratio considered, the value of the air-to-fuel mass ratio needed to obtain the desired turbine inlet temperature has been iteratively found. In the case of the RDE cycle, to ensure that the detonation will take place, a stoichiometric mixture between air and fuel has been used, and, starting from the results obtained at the combustion outlet, another stream of air at the same pressure is added to the detonation products, in order to lower their temperature. Also in this case, the amount of air needed to reach the desired turbine inlet temperature has been computed iteratively. Since the air is added at the same pressure of the detonation products at the exit of the combustor, the power needed to compress this air must be taken into account in the efficiency evaluation.

The results of this comparison are reported in figure 5.6

5.3 Power Plant B

In this section, the results obtained from the efficiency evaluation of the power plant B, shown in figure 3.2 are reported and compared first with the one obtained from the power plant A, and then with the one obtained from a standard Joule-Brayton cycle.

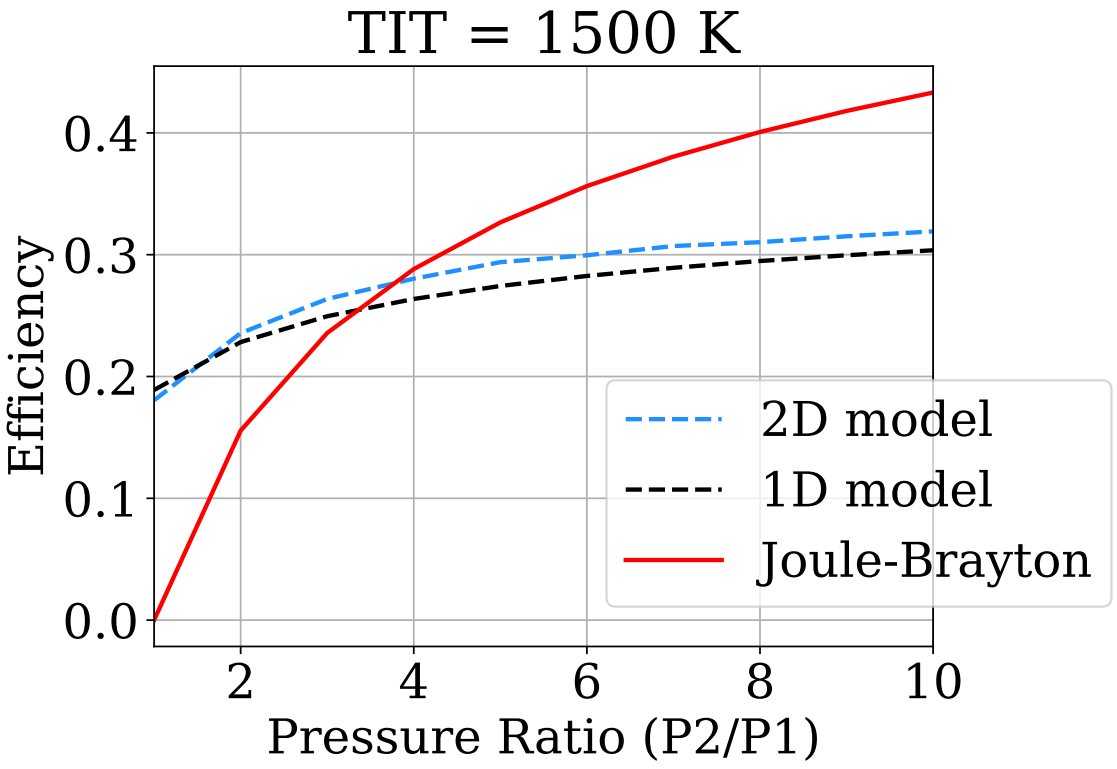

Figure 5.6: Efficiency of power plant A as a function of the compressor pressure ratio, compared with a Joule-Brayton cycle, in case of a limiting turbine inlet temperature of 1500 K, reached by increasing the amount of air

The same detonation front height and combustor radius as the one reported in section 5.2 have been used. Also, the same assumptions previously made regarding the turbine and the compressor are still valid in this section.

This time, the compressor pressure ratio considered ranges from a value of 1 to a value of 15, and the post-cooling after the flow compression is assumed to be ideal, with the flow being isobarically cooled until the temperature reaches again its initial value of 300 K.

The comparison between the efficiency of power plant A and power plant B, as a function of the compressor pressure ratio, is shown in figure 5.7.

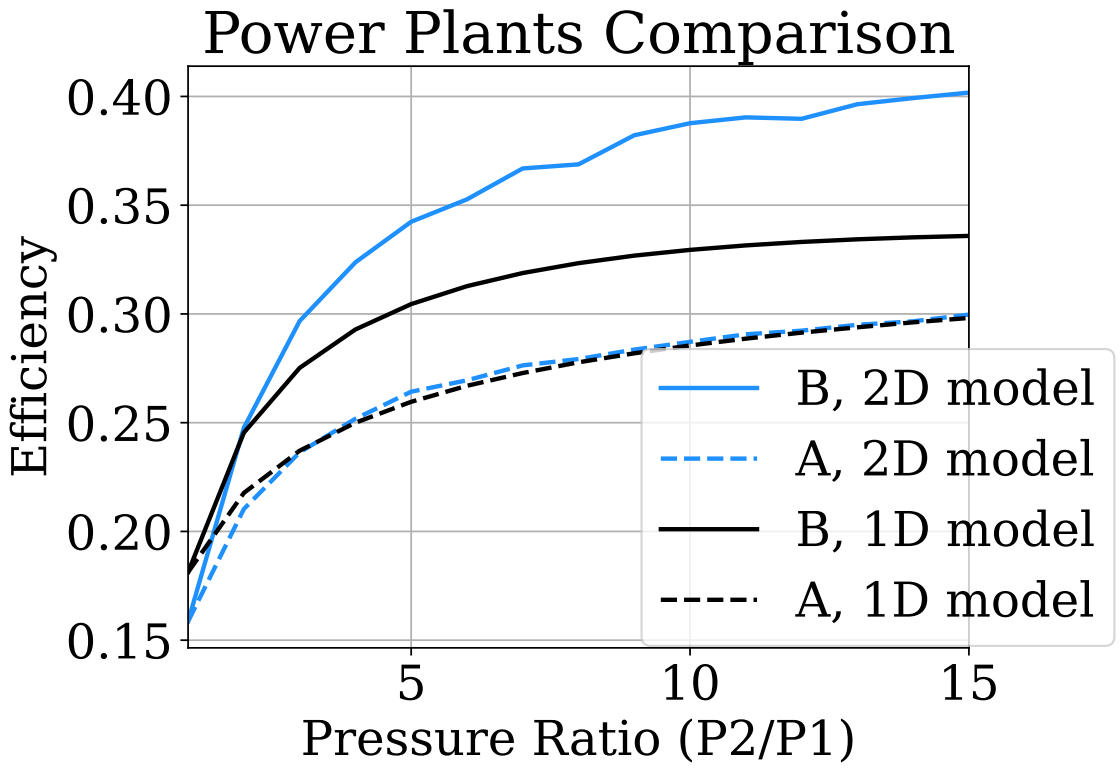

Figure 5.7: Comparison between efficiency of power plant A and efficiency of power plant B as a function of the compressor pressure ratio

The analysis has been carried out without imposing any limitation on the turbine inlet temperature, and the two cycles are compared at the same value of mass flow rate and compressor ratio. Efficiency has been evaluated with equation 5.7.

Also in this case, the results have been compared with the efficiency of a Joule-Brayton cycle, as shown in figure 5.8

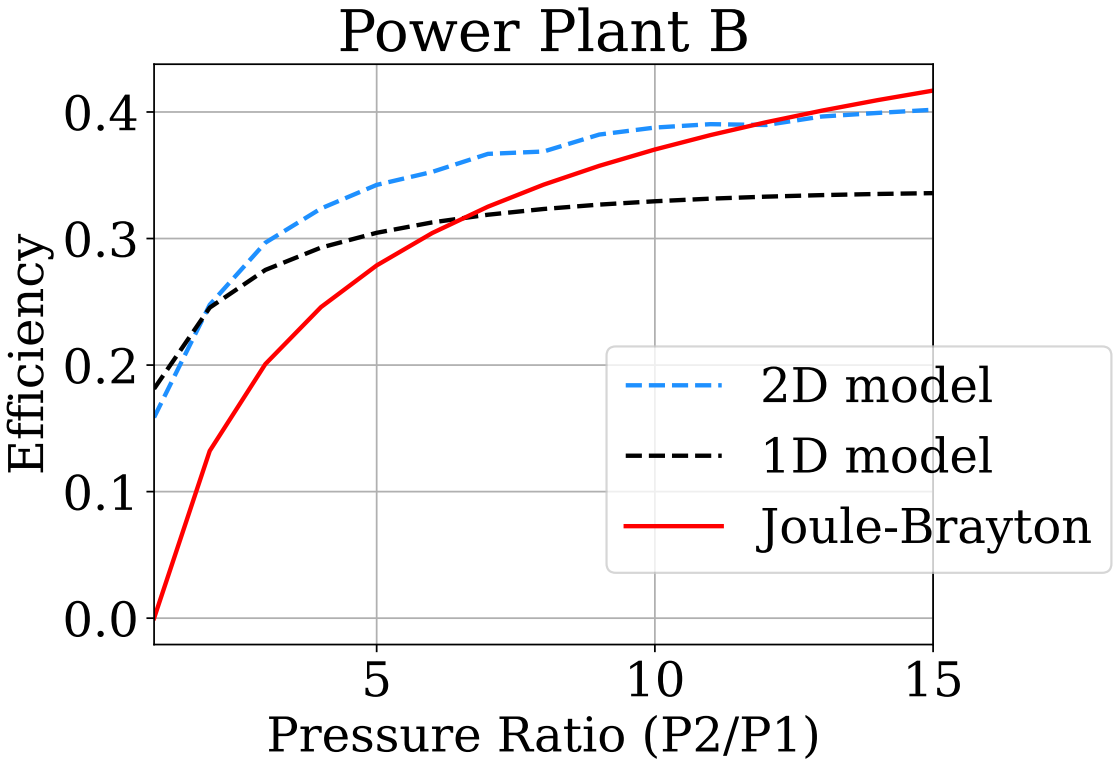

Figure 5.8: Comparison between efficiency of power plant A and efficiency of power plant B as a function of the compressor pressure ratio

6. Discussion

In this section, the results obtained and reported in chapter 5 are discussed and commented.

The first finding of the present thesis is represented in figure 5.3. Here, it is possible to observe that, unlike the detonation cycle, represented in figures 1.10 and 1.11, a power cycle featuring a rotating detonation combustor, from a thermodynamic point of view, is not always more efficient than a Joule-Brayton cycle.

This effect is explainable with the fact that, by reference to figure 1.11, in the case of a cycle featuring those types of combustors, starting from the post-detonation conditions, a first part of the expansion process happens inside the combustor itself. This expansion constitutes a 'parasitic' expansion, since the turbine is unable to recover work from it.

In particular, by looking at figure 5.3, it is possible to observe that the cycle represented in figure 3.1 shows higher efficiency than the Joule-Brayton at low values of pressure ratio. This is due to the fact that a rotating detonation combustor, according to the used models, presents a pressure gain combustion, which leads to a value of efficiency greater than zero even when a unitary pressure ratio is used. Then, until the pressure ratio remains low, the rotating detonation combustor cycle still shows an advantage in terms of efficiency. Anyway, at some point, when the pressure ratio became high enough, it is possible to observe that the trend is reversed. This happens because, while the turbine inlet pressure of the studied cycle remains higher than the one of the Joule-Brayton, its temperature results in being lower. In fact, the 'parasitic' expansion process that is happening inside the combustor is responsible for a temperature reduction from the post-detonation value to a value lower than the one obtainable from a constant-pressure stoichiometric hydrogen-air combustion. As a consequence of this fact, the Joule-Brayton cycle shows a higher efficiency at higher values of pressure ratio.

A similar trend is observed in figure 5.4. Anyway, in this case, the two cycles are compared at the same amount of net power produced, and, while the RDE cycle presents a stoichiometric air-to-fuel ratio, in the Joule-Brayton cycle this value has been set equal to 70. As a consequence, the observed turbine inlet temperatures result in being lower in the Joule-Brayton case due to the excess air used. For this reason, and also for the higher flow pressure at the turbine inlet, the RDE cycle shows higher values of specific turbine work $W_{turbine}$ at any value of compressor pressure ratio. Nevertheless, also in this case, when the pressure ratio is high enough, the efficiency of the Joule-Brayton cycle results in being higher than the one of the RDE cycle. In fact, considering the definition used for evaluating the cycle efficiency (equation 5.7), while the numerator is equal in both cases, the denominator results in being lower in the case of a Joule-Brayton cycle due to the used value of excess air.

Anyway, as expected, if a limiting turbine inlet temperature is considered and the two cycles present the same values of mass flow rate, the rotating detonation cycle shows a higher efficiency compared to the Joule-Brayton, as reported in figure 5.5. The higher

efficiency is a consequence of the pressure-gain combustion presented by the rotating detonation cycle, and, since the turbine inlet temperature has been saturated at the material limiting value, the Joule Brayton cycle has lost the advantage of entering the turbine at a higher temperature. At the same time, it is also possible to notice that, with the increasing of the pressure ratio, the efficiency difference between the two cycles is diminishing. Also, as expected, by limiting the cycle maximum temperature, the absolute values of the efficiencies are lower if compared to the previous cases.

However, if the cooling of the flow is performed with the addition of air, the Joule-Brayton cycle shows again higher values of efficiency at higher values of pressure ratio, as reported in figure 5.6. Also in this case, the RDE cycle shows higher turbine specific work, but the Joule-Brayton cycle requires a higher amount of excess air and, as a consequence, produces more useful power.

Another interesting aspect of the cycle featuring rotating detonation combustors is represented by the fact that the efficiency curve as a function of compressor pressure ratio tends to be more 'flat' than the one of a Joule-Brayton cycle. In fact, in all the figures reported in chapter 5, after a sufficiently high value of pressure ratio, the curve of efficiency shows a really small increase for a very large pressure ratio increase. The reason for this behavior can be found by looking at the detonation map reported in figure 4.1. As it is possible to observe, the post-detonation temperature is really lightly affected by the mixture inlet pressure, and presents an almost constant value. As a consequence, the efficiency curve quickly reaches a plateau.

Also, as expected from the detonation maps reported in figures 4.1 and 4.2, a possible way to increase the efficiency consists in reducing the combustor inlet temperature. This will have the effect of slightly reducing the post-detonation temperature, but substantially increasing the post-detonation pressure. As shown in figure 5.7, it is clear that the addition of a post-cooler after the compressor and before the rotating detonation combustor had the effect of increasing the thermodynamic cycle efficiency. Anyway, as shown in figure 5.8, despite the efficiency being increased with respect to the power plant A, also in the case of the second configuration, at a high enough value of pressure ratio, the Joule-Brayton cycle will present a higher efficiency. Of course, in this case, the Joule-Brayton cycle will start to show a higher efficiency value from a higher value of pressure ratio with respect to the case shown in figure 5.3.

Another important aspect regards the importance of the combustor modeling. In fact, while the 1D model and the 2D model show a good agreement regarding the performance evaluation of power plant A, as shown in figures 5.3 and 5.4, the efficiency predicted in the case of an imposed limiting turbine inlet temperature results in being higher if the 1D model is used, as reported in figure 5.5. This is a consequence of the model overestimating the combustor outlet pressure with respect to the 2D one. On the other hand, in the case of power plant B, it is the 2D model that shows an overall higher efficiency, as shown in figures 5.7 and 5.8. This is probably associated with the tendency of the 2D model to overestimate the combustor outlet temperature, with respect to the 1D model, which may underestimate it. A slightly higher value of efficiency is also predicted from the 2D

model in figure 5.6, which is also a consequence of the higher combustor outlet temperature predicted. In fact, for this reason, more air is needed to cool down the combustor outlet flow of the 2D model with respect to the 1D one, and, consequently, a higher value of mass flow rate is expanded in the turbine.

Another important aspect to consider is that the present analysis has been conducted only from a purely thermodynamical point of view, assuming, as is usually done in power cycle analysis, that the flow velocity at the inlet of the turbine is comparable with the one at the outlet. Anyway, in the case of a rotating detonation combustor, the flow velocity at the combustor outlet is quite important, and therefore, with a proper turbine design able to recover the flow kinetic component at its inlet, the efficiency values can be further increased.

Also, the interaction between the combustor outlet flow field, which results in being highly irregular and non-uniform, and the turbine blades represents a challenging topic that has to be further analyzed. This aspect is not considered in the present work, since the main focus has been set on the combustor modeling.

Another aspect to mention is that, even if the two presented rotating detonation combustor models were able to predict a pressure gain combustion, still nobody, according to the author's knowledge, was able to achieve it from an experimental point of view.

In conclusion, based on the results obtained from the current analysis, a cycle implementing a rotating detonation combustor can provide an advantage in terms of specific turbine work if compared to the Joule-Brayton cycle. Anyway, the efficiency of the latter seems to be higher in almost all the cases considered, except at low values of pressure ratio, where the pressure-gain combustion provided by the detonation makes the RDE cycle more efficient. Finally, the current analysis has established that the addition of a post-cooler after the compressor consists of a possible way to enhance the rotating detonation combustor cycle efficiency.

7. Conclusion

The current climate change is forcing the scientific community to find alternatives to the actual fuel-based economy. Anyway, the transition in the energy field, with a switch from the actual scenario to another where all the world is powered by energy from renewable sources, results in being challenging due to the volatility intrinsically related to these sources. Hydrogen is expected to have a huge role in the following years since it can be produced by electrolysis using excess energy from the renewable sources, and at the same time it can be used as fuel for power plants which have the role to constantly provide a base load for the electric grid.

In addition to that, new ways of enhancing the power plant efficiency are currently under research, and one of the most promising ideas consists of replacing the constant pressure combustion with detonation, which, thanks to the introduced pressure gain in the combustion stage, is expected to show better performance. In particular, the rotating detonation engine consists of one or more detonation waves constantly travelling around a cylindrical annulus, ensuring continuous operation and combustion of the fuel.

The present work focuses on the possible thermodynamic cycle analysis of power plants featuring hydrogen-fueled rotating detonation combustors.

In the beginning, the theoretical foundation of the detonation theory is summarized, in order to provide the reader with the necessary background to understand the following analysis. A comparison of the ideal detonation cycle with respect to the constant volume and constant pressure combustion is also represented, in order to further motivate the reasons behind the present work. Then, a general description of the working principles of a rotating detonation combustor has been provided, explaining all the sources of losses in these devices and, as a consequence, the reasons why their efficiency is really likely to not match the one of the ideal detonation cycle. After that, a preliminary analysis on post-detonation conditions as a function of different combustor inlet conditions has been performed. The results of this analysis showed that the post-detonation temperature is minimally affected by the inlet pressure and temperature, while, with the reduction of the flow inlet temperature, the combustor outlet pressure reported a substantial increase. Then, two power plant configurations have been presented. The first one consists of a modification of the Joule-Brayton cycle, where the standard combustor has been replaced with a rotating detonation combustor. Taking into account the results of the preliminary analysis, the second power plant configuration implemented a post-cooler after the compressor, in order to reduce the flow combustor inlet temperature. Two rotating detonation models have been proposed: a simpler 1D model and a 2D model. Both models share the same iteration process to find the combustor injection velocity, but, while in the 1D model the outlet conditions are found by assuming a sonic and completely axial outlet flow, in the 2D model the method of characteristics has been used to compute all the quantities relative to the post-detonation flow field. Both models have been validated by comparison with a CFD simulation performed by Sousa et al., 2017.

The present work has confirmed that the rotating detonation cycle, in the presence of a limiting turbine inlet temperature, shows higher turbine specific work compared to the Joule-Brayton cycle. Anyway, regarding the efficiency, the rotating detonation engine cycle results in overperforming the Joule-Brayton only at low values of pressure ratio, thanks to its pressure-gain combustion. The only case when the RDE cycle totally overperforms the Joule-Brayton is in the presence of a limiting turbine inlet temperature and when the same air-to-fuel ratio is used for both the cycles. In all other cases, the key advantage of the Joule-Brayton cycle is its ability to operate with high levels of excess air, which is something that is not feasible for the RDE cycle due to detonability limits. Also, the 'parasitic' expansion of the flow, which takes part in the combustor, is reducing the flow temperature to a value lower than the one obtainable from a constant pressure combustion, allowing the Joule-Brayton cycle to show higher values of efficiency at high values of pressure ratio even when no limitation on the turbine inlet temperature is imposed. Finally, the present work has confirmed that the addition of a post-cooler after the compressor results in enhancing the base-cycle efficiency.

In conclusion, the rotating detonation combustor represents a technology that can assure some advantages compared to the current power plant design. Anyway, its efficiency, according to the results obtained from the presented models, results in being way much lower than that of the ideal detonation cycle. Therefore, in order to get closer to this ideal value, new combustor design solutions to limit the 'parasitic' expansions need to be identified and developed.

Bibliography

Abel, F. A. (1869). Xiv. contributions to the history of explosive agents. *Philosophical transactions of the Royal Society of London*, (159), 489–516.

Alhussan, K., Assad, M., & Penazkov, O. (2016). Analysis of the actual thermodynamic cycle of the detonation engine. *Applied thermal engineering*, *107*, 339–344.

Ashford, S., & Emanuel, G. (1996). Oblique detonation wave engine performance prediction. *Journal of propulsion and power*, *12*(2), 322–327.

Assad, M., & Tunik, Y. (2023). Restrictions on the thermodynamic efficiency of the detonation cycle. *Applied Thermal Engineering*, *219*, 119548.

Bach, E., Paschereit, C. O., Stathopoulos, P., & Bohon, M. D. (2021). An empirical model for stagnation pressure gain in rotating detonation combustors. *Proceedings of the Combustion Institute*, *38*(3), 3807–3814.

Braun, E. M., Lu, F. K., Wilson, D. R., & Camberos, J. A. (2013). Airbreathing rotating detonation wave engine cycle analysis. *Aerospace Science and Technology*, *27*(1), 201–208.

Browne, S. T., Ziegler, J. L., Bitter, N. P., Schmidt, B. E., Lawson, J., & Shepherd. (2023). *Numerical solution methods for shock and detonation jump conditions* (tech. rep.). California Institute of Technology.

Bykovskii, F., & Zhdan, S. (2013). Continuously detonation engine. *Doklady Akademii Nauk*.

Bykovskii, F. A., Mitrofanov, V. V., & Vedernikov, E. F. (1997). Continuous detonation combustion of fuel-air mixtures. *Combustion, Explosion and Shock Waves*, *33*(3), 344–353.

Bykovskii, F. A., & Vedernikov, E. F. (2003). Continuous detonation of a subsonic flow of a propellant. *Combustion, Explosion and Shock Waves*, *39*, 323–334.

Bykovskii, F. A., Zhdan, S. A., & Vedernikov, E. F. (2005). Continuous spin detonation in annular combustors. *Combustion, Explosion and Shock Waves*, *41*, 449–459.

Bykovskii, F. A., Zhdan, S. A., & Vedernikov, E. F. (2006). Continuous spin detonations. *Journal of propulsion and power*, *22*(6), 1204–1216.

Campbell, C. (1927). The ignition of gases by an explosion wave. part 1. carbon monoxide and hydrogen mixtures. *J. Chem. Soc*, *1927*, 1572–1578.

Chapman, D. L. (1899). Vi. on the rate of explosion in gases. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, *47*(284), 90–104.

Ciccarelli, G., & Dorofeev, S. (2008). Flame acceleration and transition to detonation in ducts. *Progress in energy and combustion science*, *34*(4), 499–550.

Cocks, P. A., Holley, A. T., & Rankin, B. A. (2016). High fidelity simulations of a non-premixed rotating detonation engine. *54th AIAA aerospace sciences meeting*, 0125.

Doring, W. (1943). Detonation waves. *Ann. Phys. 5e Folge*, *43*, 421–436.

Fernandes, T., Souza, A., & Afonso, F. (2023). A shape design optimization methodology based on the method of characteristics for rocket nozzles. *CEAS Space Journal*, *15*(6), 867–879.

Frolov, S. M., Aksenov, V. S., Dubrovskii, A. V., Ivanov, V. S., & Shamshin, I. (2015). Energy efficiency of a continuous-detonation combustion chamber. *Combustion, Explosion, and Shock Waves*, *51*, 232–245.

Frolov, S. M., Aksenov, V., & Ivanov, V. (2015). Experimental proof of zel'dovich cycle efficiency gain over cycle with constant pressure combustion for hydrogen–oxygen fuel mixture. *international journal of hydrogen energy*, *40*(21), 6970–6975.

George, A. S., Driscoll, R., Anand, V., & Gutmark, E. (2017). On the existence and multiplicity of rotating detonations. *Proceedings of the Combustion Institute*, *36*(2), 2691–2698.

George, A. S., Driscoll, R., Gutmark, E., & Munday, D. (2014). Experimental comparison of axial turbine performance under steady and pulsating flows. *Journal of Turbomachinery*, *136*(11), 111005.

Glaser, A., Caldwell, N., & Gutmark, E. (2007). Performance of an axial flow turbine driven by multiple pulse detonation combustors. *45th AIAA Aerospace Sciences Meeting and Exhibit*, 1244.

Goodwin, D. G., Speth, R. L., Moffat, H. K., & Weber, B. W. (2018). Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes. *Zenodo*.

Hall, I. (1975). Inversion of the prandtl-meyer relation. *The Aeronautical Journal*, *79*(777), 417–418.

Hishida, M., Fujiwara, T., & Wolanski, P. (2009). Fundamentals of rotating detonations. *Shock waves*, *19*, 1–10.

Hutchins, T. E., & Metghalchi, M. (2003). Energy and exergy analyses of the pulse detonation engine. *J. Eng. Gas Turbines Power*, *125*(4), 1075–1080.

IEA. (2023). *Greenhouse gas emissions from energy* (tech. rep.). IEA, Paris.

Jouguet, E., et al. (1904). Sur l'onde explosive. *CR Acad. Sci. Paris*, *140*, 1211.

Kaemming, T., Fotia, M. L., Hoke, J., & Schauer, F. (2017). Thermodynamic modeling of a rotating detonation engine through a reduced-order approach. *Journal of Propulsion and Power*, *33*(5), 1170–1178.

Kanda, T., & Inagaki, H. (2024). Analytical study of rotating detonation and engine operating conditions. *Physics of Fluids*, *36*(11).

Katta, V. R., Cho, K. Y., Hoke, J. L., Codoni, J. R., Schauer, F. R., & Roquemore, W. M. (2019). Effect of increasing channel width on the structure of rotating detonation wave. *Proceedings of the Combustion Institute*, *37*(3), 3575–3583.

Kawashima, R., Funaki, I., Fujii, J., Matsuo, A., Ishihara, K., & Kasahara, J. (2017). A quasi-one-dimensional analytic model of rotating detonation combustors. *arXiv preprint arXiv:1712.02915*.

Kindracki, J., Kobiera, A., Wolański, P., Gut, Z., Folusiak, M., & Swiderski, K. (2011). Experimental and numerical study of the rotating detonation engine in hydrogen-air mixtures. *Progress in propulsion physics*, *2*, 555–582.

Kuo, K. K. (1986, January). *Principles of combustion*. Elsevier Science Pub. Co. Inc.,New York, NY.

Kyle Niemeyer. (n.d.). Gas dynamics notes [Online]. https://kyleniemeyer.github.io/gas-dynamics-notes/compressible-flows/method-characteristics.html

LEE, B. H., LEE, J. H., & KNYSTAUTAS, R. (1966). Transmission of detonation waves through orifices. *AIAA Journal*, *4*(2), 365–367.

Lee, J. H. (2008). *The detonation phenomenon*. Cambridge University Press.

Lee, J. H., & Radulescu, M. I. (2005). On the hydrodynamic thickness of cellular detonations. *Combustion, Explosion and Shock Waves*, *41*, 745–765.

Liepmann, H. W., & Roshko, A. (2001). *Elements of gasdynamics*. Courier Corporation.

Lindstedt, R., & Michels, H. (1989). Deflagration to detonation transitions and strong deflagrations in alkane and alkene air mixtures. *Combustion and Flame*, *76*(2), 169–181.

Liu, P., Guo, Q., Sun, D., Li, C., & Zhang, H. (2020). Wall effect on the flow structures of three-dimensional rotating detonation wave. *International Journal of Hydrogen Energy*, *45*(53), 29546–29559.

Lu, J., & Zheng, L. (2016). Investigations on key technologies of the pulse detonation turbine engine. *Northwestern Polytechnical University*.

Miao, S., Zhou, J., Lin, Z., Cai, X., & Liu, S. (2018). Numerical study on thermodynamic efficiency and stability of oblique detonation waves. *AIAA Journal*, *56*(8), 3112–3122.

Nordeen, C. A., Schwer, D., Schauer, F., Hoke, J., Barber, T., & Cetegen, B. (2014). Thermodynamic model of a rotating detonation engine. *Combustion, Explosion, and Shock Waves*, *50*, 568–577.

Ostrander, M., Hyde, J., Young, M., Kissinger, R., & Pratt, D. (1987). Standing oblique detonation wave engine performance. *23rd Joint Propulsion Conference*, 2002.

Peng, C., Fan, W., Zheng, L., Wang, Z., & Yuan, C. (2013). Experimental investigation on valveless air-breathing dual-tube pulse detonation engines. *Applied Thermal Engineering*, *51*(1-2), 1116–1123.

Rankin, B. A., Richardson, D. R., Caswell, A. W., Naples, A. G., Hoke, J. L., & Schauer, F. R. (2017). Chemiluminescence imaging of an optically accessible non-premixed rotating detonation engine. *Combustion and Flame*, *176*, 12–22.

Rouser, K. P. (2012). *Unsteady specific work and isentropic efficiency of a radial turbine driven by pulsed detonations*. Air Force Institute of Technology.

Roux, J. (2015). Parametric cycle analysis of an ideal pulse detonation engine. *Journal of Thermophysics and Heat Transfer*, *29*(4), 671–677.

Russo, R. M. (2011). Operational characteristics of a rotating detonation engine using hydrogen and air.

Schwer, D., & Kailasanath, K. (2010). Numerical investigation of rotating detonation engines. *46th AIAA/ASME/SAE/ASEE joint propulsion conference & exhibit*, 6880.

Schwer, D., & Kailasanath, K. (2011a). Numerical investigation of the physics of rotating-detonation-engines. *Proceedings of the combustion institute*, *33*(2), 2195–2202.

Schwer, D., & Kailasanath, K. (2011b). Numerical study of the effects of engine size in rotating detonation engines. *49th AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition*, 581.

Shepherd, J. E., & Kasahara, J. (2017). Analytical models for the thrust of a rotating detonation engine. *GALCIT Report FM2017*, *1*.

Shimo, M., & Heister, S. D. (2008). Multicyclic-detonation-initiation studies in valveless pulsed detonation combustors. *Journal of Propulsion and Power*, *24*(2), 336–344.

Sichel, M., & Foster, J. (1980). The ground impulse generated by a plane fuel-air explosion with side relief. In *Gasdynamics of explosions and reactive systems* (pp. 243–256). Elsevier.

Sislian, J., Schirmer, H., Dudebout, R., & Schumacher, J. (2001). Propulsive performance of hypersonic oblique detonation wave and shock-induced combustion ramjets. *Journal of Propulsion and Power*, *17*(3), 599–604.

Sousa, J., Braun, J., & Paniagua, G. (2017). Development of a fast evaluation tool for rotating detonation combustors. *Applied Mathematical Modelling*, *52*, 42–52.

Sun, J., Zhou, J., Liu, S., & Lin, Z. (2018). Numerical investigation of a rotating detonation engine under premixed/non-premixed conditions. *Acta astronautica*, *152*, 630–638.

Tsuboi, N., Watanabe, Y., Kojima, T., & Hayashi, A. K. (2015). Numerical estimation of the thrust performance on a rotating detonation engine for a hydrogen–oxygen mixture. *Proceedings of the Combustion Institute*, *35*(2), 2005–2013.

Urtiew, P., & Oppenheim, A. (1966). Experimental observations of the transition to detonation in an explosive gas. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, *295*(1440), 13–28.

Valorani, M., Di Giacinto, M., & Buongiorno, C. (2001). Performance prediction for oblique detonation wave engines (odwe). *Acta Astronautica*, *48*(4), 211–228.

Vasil'ev, A. (2006). Cell size as the main geometric parameter of a multifront detonation wave. *Journal of propulsion and Power*, *22*(6), 1245–1260.

Vasil'ev, A. A. (1982). Geometric limits of gas detonation propagation. *Combustion, Explosion and Shock Waves*, *18*(2), 245–249.

Vasil'ev, A. A., Mitrofanov, V. V., & Topchiyan, M. E. (1987). Detonation waves in gases. *Combustion, Explosion and Shock Waves*, *23*(5), 605–623.

Von Neumann, J. (1942). Theory of detonation waves. *John von Neumann, collected works*, *6*, 203–218.

Vutthivithayarak, R., Braun, E., & Lu, F. (2012). On thermodynamic cycles for detonation engines. *28th International Symposium on Shock Waves: Vol 2*, 287–292.

Wen, H., Fan, W., & Wang, B. (2023). Theoretical analysis on the total pressure gain of rotating detonation systems. *Combustion and Flame*, *248*, 112582.

Wen, H., Xie, Q., & Wang, B. (2019). Propagation behaviors of rotating detonation in an obround combustor. *Combustion and Flame*, *210*, 389–398.

Wintenberger, E., & Shepherd, J. E. (2006). Thermodynamic cycle analysis for propagating detonations. *Journal of propulsion and power*, *22*(3), 694–698.

Wolanski, P. (2011). Detonation engines. *Journal of KONES Powertrain and Transport*, *18*, No.3.

World Meteorological Organization. (2024). *State of the global climate 2023* (tech. rep.). WMO.

Xisto, C., Petit, O., Grönstedt, T., Rolt, A., Lundbladh, A., & Paniagua, G. (2018). The efficiency of a pulsed detonation combustor–axial turbine integration. *Aerospace Science and Technology*, *82*, 80–91.

Yan, C., & Fan, W. (2005). Principle and key technology of pulse detonation engine. *Northwestern Polytechnical University, Xi'an, China*.

Zeldovich, Y. B. (1940). Zh. exp. teor. fiz. *NACA TN*, *1261*, 542–568.

Zucrow, M. J., & Hoffman, J. D. (1977). Gas dynamics. volume 2-multidimensional flow. *New York*.

0.1 Python code of the 1D combustor model

```
import cantera as ct
from sdtoolbox.thermo import soundspeed_fr, soundspeed_eq
from sdtoolbox.postshock import CJspeed, PostShock_eq, PostShock_fr
from scipy.interpolate import pchip # creates a PCHIP interpolation function t
from scipy.optimize import fminbound
from sys import exit
import matplotlib.pyplot as plt
import numpy as np
from math import *
from pynverse import *
from numpy.polynomial import Polynomial as P
from scipy.optimize import fsolve

#Parameters


p_inj=5e5

T_inj=300

Radius=0.153

d=2*Radius

det_h=0.092

mech='gri30_highT.yaml'
X1='H2:2 O2:1 N2:3.76'
```

```
#RDE model



#mixture properties


gas=ct.Solution(mech)
gas.TPX=T_inj,p_inj,X1


cp1=gas.cp_mass
R1=gas.cp_mass-gas.cv_mass
gamma1=cp1/gas.cv_mass


#guess values


v_inj=250


check=0

while check==0:



    #detonation properties

    cj_speed=CJspeed(p_inj, T_inj, X1, mech)
    gas=PostShock_eq(cj_speed, p_inj, T_inj, X1, mech)

    p2=gas.P
    T2=gas.T

    cp2=gas.cp_mass
    R2=gas.cp_mass-gas.cv_mass
    gamma2=cp2/gas.cv_mass


    #functions
```

```
PM= lambda M: sqrt((gamma2+1)/(gamma2-1))*atan(sqrt((gamma2-1)/(gamma2+1)*
invPM=inversefunc(PM, domain=[1,30])


mu= lambda M: asin(1/M)
invmu=lambda mu: 1/sin(mu)


#finding v_inj and teta0


teta0=asin(v_inj/cj_speed)

L1=det_h/tan(teta0)

Xref=pi*d-L1

#print('Xref ',Xref)


pt2=p2*(1+(gamma2-1)/2)**(gamma2/(gamma2-1))

Minj=sqrt(2/(gamma2-1)*((pt2/p_inj)**((gamma2-1)/gamma2)-1))

PM4=PM(Minj)/4

mu4=mu(invPM(PM4))

Xref2=det_h/(tan(mu4-(teta0+PM4)))


if abs(Xref-Xref2)/Xref2<=1e-5:

    check=2

else:

    L1=pi*d-Xref2

    teta0=atan(det_h/L1)

    v_inj=cj_speed*sin(teta0)
```

```
gas.TPX=T_inj, p_inj, X1
rho_inj=gas.density_mass

Ainj=pi*d-Xref

mi=rho_inj*v_inj*Ainj

Fi=mi*v_inj+Ainj*p_inj+p2*Xref

Tt2=T2+(1+(gamma2-1)/2)

uec=cj_speed*cos(teta0)

Te=(Tt2-1/(2*cp2)*uec**2)/(1+1/(2*cp2)*gamma2*R2)

uea=sqrt(gamma2*R2*T2)

pe=Fi/(pi*d)-mi*uea

pte=pe*(1+(gamma2-1)/2*(uea/sqrt(gamma2*R2*Te))**2)**(gamma2/(gamma2-1))
Tte=Te*(1+(gamma2-1)/2*(uea/sqrt(gamma2*R2*Te))**2)
```

0.2 Python code of the 2D combustor model

```
import cantera as ct
from sdtoolbox.thermo import soundspeed_fr, soundspeed_eq
from sdtoolbox.postshock import CJspeed, PostShock_eq, PostShock_fr
from scipy.interpolate import pchip # creates a PCHIP interpolation function t
from scipy.optimize import fminbound
from sys import exit
import matplotlib.pyplot as plt
import numpy as np
from math import *
from pynverse import *
from scipy.optimize import fsolve

plt.rcParams['font.size'] = 22
plt.rcParams['font.family'] = 'serif'
plt.rcParams['font.weight'] = 'light'


#PARAMETERS

p0=8e5
```

```
T0=330


npoints=30

mol_H2=2
mol_O2=1
mol_N2=3.76

beta=mol_N2/mol_O2

det_cell=10**(-0.048*beta**2+0.46*beta+0.047)*10**(-3)

Radius=0.153

L=0.2



det_h=0.092


Vinj=200

#FUNCTIONS

def scaledM(M, gamma_med):

    M_p3_sc=sqrt(abs(2/((gamma_med+1)/M**2-gamma_med+1)))

    P_p3=P0_MOC/(1+0.5*(gamma_med-1)*M_p3_sc**2)**(gamma_med/(gamma_med-1))

    T_p3=T0_MOC/(1+0.5*(gamma_med-1)*M_p3_sc**2)

    return P_p3, T_p3




def center_MOC(p1,p2):
```

```
x_p1=p1[0]
y_p1=p1[1]
M_p1=p1[2]
PM_p1=p1[3]
mu_p1=p1[4]
teta_p1=p1[5]

x_p2=p2[0]
y_p2=p2[1]
M_p2=p2[2]
PM_p2=p2[3]
mu_p2=p2[4]
teta_p2=p2[5]

Km=teta_p1+PM_p1
Kp=teta_p2-PM_p2

teta_p3=(Km+Kp)/2
PM_p3=(Km-Kp)/2
M_p3=invPM(PM_p3)
mu_p3=mu(M_p3)
P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))
#P_p3=P0_MOC/((1+(gamma2-1)/2*M_p3**2)**(gamma2/(gamma2-1)))


slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2)
#slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2+beta_ref)

slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2)
#slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2+beta_ref)


'''
x_p3=(y_p1-slopeCm*x_p1+slopeCp*x_p2-y_p2)/(slopeCp-slopeCm)
y_p3=y_p2+slopeCp*(x_p3-x_p2)
'''
a=np.array([[1, -slopeCp], [1, -slopeCm]])
b=np.array([y_p2-slopeCp*x_p2, y_p1-slopeCm*x_p1])
c=np.linalg.solve(a,b)

y_p3=c[0]
x_p3=c[1]
```

```
p3=[x_p3,y_p3,M_p3,PM_p3,mu_p3,teta_p3,P_p3]

out1=np.zeros((1,7))
out2=np.zeros((1,7))

if y_p3>L:

    if y_p1>L and y_p2<L:

        x_outlet=x_p2+(L-y_p2)/slopeCp

        deltax2=x_outlet-x_p2
        deltax3=x_outlet-x_p3
        deltay2=L-y_p2
        deltay3=L-y_p3

        dist2=sqrt(deltax2**2+deltay2**2)
        dist3=sqrt(deltax3**2+deltay3**2)

        dist=dist2+dist3

        teta_out=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
        PM_out=teta_out-teta_p2+PM_p2
        M_out=invPM(PM_out)
        mu_out=mu(M_out)
        P_out=P0_MOC/((1+(gamma_med-1)/2*M_out**2)**(gamma_med/(gamma_med-

        out1[0,:]=[x_outlet,L, M_out, PM_out, mu_out, teta_out,P_out]



    if y_p1<L and y_p2<L:

        x_outlet2=x_p2+(L-y_p2)/slopeCp

        deltax2=x_outlet2-x_p2
        deltax3=x_outlet2-x_p3
        deltay2=L-y_p2
        deltay3=L-y_p3

        dist2=sqrt(deltax2**2+deltay2**2)
        dist3=sqrt(deltax3**2+deltay3**2)
```

```
            dist=dist2+dist3

            teta_out2=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
            PM_out2=teta_out2-teta_p2+PM_p2
            M_out2=invPM(PM_out2)
            mu_out2=mu(M_out2)
            P_out2=P0_MOC/((1+(gamma_med-1)/2*M_out2**2)**(gamma_med/(gamma_me

            x_outlet1=x_p1+(L-y_p1)/slopeCm

            deltax1=x_outlet1-x_p1
            deltax3=x_outlet1-x_p3
            deltay1=L-y_p1
            deltay3=L-y_p3

            dist1=sqrt(deltax1**2+deltay1**2)
            dist3=sqrt(deltax3**2+deltay3**2)

            dist=dist1+dist3

            teta_out1=(1-dist1/dist)*teta_p1+(1-dist3/dist)*teta_p3
            PM_out1=-teta_out1+teta_p1+PM_p1
            M_out1=invPM(PM_out1)
            mu_out1=mu(M_out1)
            P_out1=P0_MOC/((1+(gamma_med-1)/2*M_out1**2)**(gamma_med/(gamma_me

            out1[0,:]=[x_outlet1,L, M_out1, PM_out1, mu_out1, teta_out1,P_out1
            out2[0,:]=[x_outlet2,L, M_out2, PM_out2, mu_out2, teta_out2,P_out2



    return p3,out1,out2




def print_MOC(p1,p2,p3):

    x_p1=p1[0]
    y_p1=p1[1]
```

```
M_p1=p1[2]
PM_p1=p1[3]
mu_p1=p1[4]
teta_p1=p1[5]

x_p2=p2[0]
y_p2=p2[1]
M_p2=p2[2]
PM_p2=p2[3]
mu_p2=p2[4]
teta_p2=p2[5]

x_p3=p3[0]
y_p3=p3[1]
M_3=p3[2]
PM_p3=p3[3]
mu_p3=p3[4]
teta_p3=p3[5]

slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2)
#slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2+beta_ref)

slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2)
#slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2+beta_ref)




'''
if y_p3<L:


    x=np.linspace(x_p2,x_p3,5)
    y=y_p2+slopeCp*(x-x_p2)

    plt.plot(x,y,'b-')

    if y_p1<=L:

        x=np.linspace(x_p1,x_p3,5)
        y=y_p1+slopeCm*(x-x_p1)

        plt.plot(x,y,'b-')
```

```
        else:
            x_outlet=x_p1+(L-y_p1)/slopeCm

            x=np.linspace(x_outlet,x_p3,3)
            y=L+slopeCm*(x-x_outlet)

            plt.plot(x,y,'b-')

    else:


        if y_p1>L and y_p2<L:

            x_outlet=x_p2+(L-y_p2)/slopeCp

            x=np.linspace(x_p2,x_outlet,3)
            y=y_p2+slopeCp*(x-x_p2)

            plt.plot(x,y,'b-')

        if y_p1<L and y_p2<L:

            x_outlet2=x_p2+(L-y_p2)/slopeCp
            x_outlet1=x_p1+(L-y_p1)/slopeCm

            x=np.linspace(x_p1,x_outlet1,3)
            y=y_p1+slopeCm*(x-x_p1)

            plt.plot(x,y,'b-')

            x=np.linspace(x_p2,x_outlet2,3)
            y=y_p2+slopeCp*(x-x_p2)

            plt.plot(x,y,'b-')

    '''

    #outlet conditions

    if y_p3>L:

        if y_p1>L and y_p2<L:
```

```
            x_outlet=x_p2+(L-y_p2)/slopeCp

            if x_outlet<=x_lim:

                x=np.linspace(x_p2,x_outlet,3)
                y=y_p2+slopeCp*(x-x_p2)

                plt.plot(x,y)


        if y_p1<L and y_p2<L:

            x_outlet2=x_p2+(L-y_p2)/slopeCp
            x_outlet1=x_p1+(L-y_p1)/slopeCm

            if x_outlet2<=x_lim:

                x=np.linspace(x_p2,x_outlet2,3)
                y=y_p2+slopeCp*(x-x_p2)

                plt.plot(x,y)

            if x_outlet1<=x_lim:

                x=np.linspace(x_p1,x_outlet1,3)
                y=y_p1+slopeCm*(x-x_p1)

                plt.plot(x,y)


    #obl shock conditions

    if det_h+tan(beta_ref+teta)*(x_p3-2*pi*Radius)>y_p3:

        #entrambi i punti prima dello shock

        if det_h+tan(beta_ref+teta)*(x_p1-2*pi*Radius)<y_p1 and det_h+tan(beta

            x_shock1=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCm*x_p1-y_p1)/
            y_shock1=det_h+tan(beta_ref+teta)*(x_shock1-2*pi*Radius)
```

```
        x_shock2=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCp*x_p2-y_p2)/
        y_shock2=det_h+tan(beta_ref+teta)*(x_shock2-2*pi*Radius)


        if y_shock1<=L and y_shock1>=det_h:

            x=np.linspace(x_p1,x_shock1,3)
            y=y_p1+slopeCm*(x-x_p1)

            plt.plot(x,y)

        if y_shock2<=L and y_shock2>=det_h:

            x=np.linspace(x_p2,x_shock2,3)
            y=y_p2+slopeCp*(x-x_p2)

            plt.plot(x,y)

    #p1 prima e p2 dopo


    if det_h+tan(beta_ref+teta)*(x_p1-2*pi*Radius)<y_p1 and det_h+tan(beta


        x_shock1=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCm*x_p1-y_p1)/
        y_shock1=det_h+tan(beta_ref+teta)*(x_shock1-2*pi*Radius)

        if y_shock1>=det_h and y_shock1<=L:

            x=np.linspace(x_p1,x_shock1,3)
            y=y_p1+slopeCm*(x-x_p1)

            plt.plot(x,y)


    #p1 dopo e p2 prima


    if det_h+tan(beta_ref+teta)*(x_p1-2*pi*Radius)>y_p1 and det_h+tan(beta

        x_shock2=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCp*x_p2-y_p2)/
        y_shock2=det_h+tan(beta_ref+teta)*(x_shock2-2*pi*Radius)
```

```
            if y_shock2<=L and y_shock2>=det_h:

                x=np.linspace(x_p2,x_shock2,3)
                y=y_p2+slopeCp*(x-x_p2)

    #injection line conditions

    ##

    if y_p3<=L and det_h+tan(beta_ref+teta)*(x_p3-2*pi*Radius)<y_p3:

        if y_p1<=L and y_p2<=L:

            x=np.linspace(x_p1,x_p3,3)
            y=y_p1+slopeCm*(x-x_p1)

            plt.plot(x,y)

            x=np.linspace(x_p2,x_p3,3)
            y=y_p2+slopeCp*(x-x_p2)

            plt.plot(x,y)


        if y_p1>L and y_p2<L:

            x_outlet=x_p1+(L-y_p1)/slopeCm

            x=np.linspace(x_outlet,x_p3,3)
            y=L+slopeCm*(x-x_outlet)

            plt.plot(x,y)



def wallp_nord(p2):

    x_p2=p2[0]
    y_p2=p2[1]
    M_p2=p2[2]
    PM_p2=p2[3]
    mu_p2=p2[4]
```

```
teta_p2=p2[5]


teta_p3=delta+beta_ref
#teta_p3=delta
PM_p3=teta_p3-teta_p2+PM_p2
M_p3=invPM(PM_p3)
mu_p3=mu(M_p3)
P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))
#P_p3=P0_MOC/((1+(gamma2-1)/2*M_p3**2)**(gamma2/(gamma2-1)))

'''

M_p3=M
PM_p3=PM(M_p3)
mu_p3=mu(M_p3)
P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))


teta_p3=PM_p3-PM_p2+teta_p2
'''
slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2)
#slopeCp=tan(teta_p2+mu_p2)
#slopeCp=tan(teta_p2+mu_p2+beta_ref)


x_p3=(det_h-y_p2+slopeCp*x_p2)/(slopeCp-tan(beta_ref+delta))
y_p3=det_h+tan(beta_ref+delta)*x_p3

p3=[x_p3,y_p3,M_p3,PM_p3,mu_p3,teta_p3,P_p3]
out2=np.zeros((1,7))

if y_p3>L and y_p2<L:

    x_outlet2=x_p2+(L-y_p2)/slopeCp

    deltax2=x_outlet2-x_p2
    deltax3=x_outlet2-x_p3
    deltay2=L-y_p2
    deltay3=L-y_p3

    dist2=sqrt(deltax2**2+deltay2**2)
    dist3=sqrt(deltax3**2+deltay3**2)
```

```
        dist=dist2+dist3

        teta_out2=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
        PM_out2=teta_out2-teta_p2+PM_p2
        M_out2=invPM(PM_out2)
        mu_out2=mu(M_out2)
        P_out2=P0_MOC/((1+(gamma_med-1)/2*M_out2**2)**(gamma_med/(gamma_med-1)

        out2[0,:]=[x_outlet2,L, M_out2, PM_out2, mu_out2, teta_out2,P_out2]

    return p3,out2



def wallp_nord2(p2):

    x_p2=p2[0]
    y_p2=p2[1]
    M_p2=p2[2]
    PM_p2=p2[3]
    mu_p2=p2[4]
    teta_p2=p2[5]

    x_p1=save_points[wall_index_nord[-1],0]
    y_p1=save_points[wall_index_nord[-1],1]
    teta_p1=save_points[wall_index_nord[-1],5]


    teta_p3=teta_p2
    PM_p3=teta_p3-teta_p2+PM_p2
    M_p3=invPM(PM_p3)
    mu_p3=mu(M_p3)
    P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))


    ,,,
    M_p3=M
    PM_p3=PM(M_p3)
    mu_p3=mu(M_p3)
    P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))
```

```
teta_p3=PM_p3-PM_p2+teta_p2

,,,
#slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2)
slopeCp=tan(teta_p2+mu_p2)

slopeWall=0.5*(teta_p1+teta_p3)


a=np.array([[1,-slopeCp],[1,-slopeWall]])
b=np.array([y_p2-slopeCp*x_p2,y_p1-slopeWall*x_p1])

c=np.linalg.solve(a, b)

y_p3=c[0]
x_p3=c[1]


p3=[x_p3,y_p3,M_p3,PM_p3,mu_p3,teta_p3,P_p3]
out2=np.zeros((1,7))

if y_p3>L and y_p2<L:

    x_outlet2=x_p2+(L-y_p2)/slopeCp

    deltax2=x_outlet2-x_p2
    deltax3=x_outlet2-x_p3
    deltay2=L-y_p2
    deltay3=L-y_p3

    dist2=sqrt(deltax2**2+deltay2**2)
    dist3=sqrt(deltax3**2+deltay3**2)

    dist=dist2+dist3

    teta_out2=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
    PM_out2=teta_out2-teta_p2+PM_p2
    M_out2=invPM(PM_out2)
    mu_out2=mu(M_out2)
    P_out2=P0_MOC/((1+(gamma_med-1)/2*M_out2**2)**(gamma_med/(gamma_med-1)

    out2[0,:]=[x_outlet2,L, M_out2, PM_out2, mu_out2, teta_out2,P_out2]
```

```
    return p3,out2



def plot_pnord(p2,p3):

    x_p2=p2[0]
    y_p2=p2[1]
    M_p2=p2[2]
    PM_p2=p2[3]
    mu_p2=p2[4]
    teta_p2=p2[5]

    x_p3=p3[0]
    y_p3=p3[1]
    M_3=p3[2]
    PM_p3=p3[3]
    mu_p3=p3[4]
    teta_p3=p3[5]

    slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2)
    #slopeCp=tan(teta_p2+mu_p2)
    #slopeCp=tan(teta_p2+mu_p2+beta_ref)


    if y_p3<L:

        x=np.linspace(x_p2,x_p3,5)
        y=y_p2+slopeCp*(x-x_p2)

        plt.plot(x,y,'g-')


    else:


        if y_p2<L:

            x_outlet=x_p2+(L-y_p2)/slopeCp

            x=np.linspace(x_p2,x_outlet,3)
            y=y_p2+slopeCp*(x-x_p2)
```

```
            plt.plot(x,y,'g-')



def plot_pnord2(p2,p3):

    x_p2=p2[0]
    y_p2=p2[1]
    M_p2=p2[2]
    PM_p2=p2[3]
    mu_p2=p2[4]
    teta_p2=p2[5]

    x_p3=p3[0]
    y_p3=p3[1]
    M_3=p3[2]
    PM_p3=p3[3]
    mu_p3=p3[4]
    teta_p3=p3[5]


    x_p1=save_points[wall_index_nord[-1],0]
    y_p1=save_points[wall_index_nord[-1],1]
    teta_p1=save_points[wall_index_nord[-1],5]

    #slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2)
    slopeCp=tan(teta_p2+mu_p2)

    slopeWall=0.5*(teta_p1+teta_p3)

    if y_p3<L:

        if y_p2<L:

            x=np.linspace(x_p2,x_p3,5)
            y=y_p2+slopeCp*(x-x_p2)

            plt.plot(x,y,'g-')

        if y_p1<L:
```

```
            x=np.linspace(x_p1,x_p3,3)
            y=y_p1+slopeWall*(x-x_p1)

            plt.plot(x,y,'g-')


    else:


        if y_p2<L:

            x_outlet=x_p2+(L-y_p2)/slopeCp

            x=np.linspace(x_p2,x_outlet,3)
            y=y_p2+slopeCp*(x-x_p2)

            plt.plot(x,y,'g-')

        if y_p1<L:

            x_outlet2=x_p1+(L-y_p1)/slopeWall

            x=np.linspace(x_p1,x_outlet2,3)
            y=y_p1+slopeWall*(x-x_p1)

            plt.plot(x,y,'g-')




def wallp_sud(p1):

    x_p1=p1[0]
    y_p1=p1[1]
    M_p1=p1[2]
    PM_p1=p1[3]
    mu_p1=p1[4]
    teta_p1=p1[5]

    teta_p3=0
    #teta_p3=-beta_ref
```

```
    PM_p3=-teta_p3+teta_p1+PM_p1
    M_p3=invPM(PM_p3)
    mu_p3=mu(M_p3)
    P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))
    #P_p3=P0_MOC/((1+(gamma2-1)/2*M_p3**2)**(gamma2/(gamma2-1)))

    slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2)
    #slopeCm=tan(teta_p1-mu_p1)
    #slopeCm=tan(teta_p1-mu_p1+beta_ref)

    y_p3=0
    x_p3=x_p1-y_p1/slopeCm

    #x_p3=(slopeCm*x_p1-tan(beta_ref_new)*Xref_new-y_p1)/(slopeCm-tan(beta_ref_
    #y_p3=tan(beta_ref_new)*(x_p3-Xref_new)

    p3=[x_p3,y_p3,M_p3,PM_p3,mu_p3,teta_p3,P_p3]


    return p3




def wallp_sud2(p1):

    x_p1=p1[0]
    y_p1=p1[1]
    M_p1=p1[2]
    PM_p1=p1[3]
    mu_p1=p1[4]
    teta_p1=p1[5]

    #slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2)
    slopeCm=tan(teta_p1-mu_p1)

    #y_p3=0
    #x_p3=x_p1-y_p1/slopeCm

    x_p3=(slopeCm*x_p1-tan(beta_ref_new)*Xref_new-y_p1)/(slopeCm-tan(beta_ref_n
    y_p3=tan(beta_ref_new)*(x_p3-Xref_new)
```

```
    teta_p3=0
    teta_p3=beta_ref_new
    PM_p3=-teta_p3+teta_p1+PM_p1
    M_p3=invPM(PM_p3)
    mu_p3=mu(M_p3)
    P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))
    p3=[x_p3,y_p3,M_p3,PM_p3,mu_p3,teta_p3,P_p3]


    return p3



def wallp_sud3(p1):

    x_p1=p1[0]
    y_p1=p1[1]
    M_p1=p1[2]
    PM_p1=p1[3]
    mu_p1=p1[4]
    teta_p1=p1[5]

    #slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2)
    slopeCm=tan(teta_p1-mu_p1)

    gamma=0.5*(gamma_med+gamma1)

    #y_p3=0
    #x_p3=x_p1-y_p1/slopeCm

    x_p3=(slopeCm*x_p1-tan(beta_ref_new)*Xref_new-y_p1)/(slopeCm-tan(beta_ref_n
    y_p3=tan(beta_ref_new)*(x_p3-Xref_new)

    P_p3=p
    #M_p3_tr=sqrt(2/(gamma-1)*((P0_MOC/P_p3)**((gamma-1)/gamma)-1))
    #M_p3=sqrt(abs((gamma+1)/(2/M_p3_tr**2+gamma-1)))

    M_p3=sqrt(2/(gamma_med-1)*((P0_MOC/P_p3)**((gamma_med-1)/gamma_med)-1))

    PM_p3=PM(M_p3)
    mu_p3=mu(M_p3)
    teta_p3=-PM_p3+teta_p1+PM_p1
```

```
    p3=[x_p3,y_p3,M_p3,PM_p3,mu_p3,teta_p3,P_p3]

    return p3



def plot_psud(p1,p3):

    x_p1=p1[0]
    y_p1=p1[1]
    M_p1=p1[2]
    PM_p1=p1[3]
    mu_p1=p1[4]
    teta_p1=p1[5]

    x_p3=p3[0]
    y_p3=p3[1]
    M_3=p3[2]
    PM_p3=p3[3]
    mu_p3=p3[4]
    teta_p3=p3[5]

    slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2)
    #slopeCm=tan(teta_p1-mu_p1)
    #slopeCm=tan(teta_p1-mu_p1+beta_ref)


    if x_p3<=2*pi*Radius:

        x=np.linspace(x_p1,x_p3,5)
        y=y_p1+slopeCm*(x-x_p1)

        plt.plot(x,y,'r-')



def MOC_3(p1,p2):


    x_p1=p1[0]
```

```
y_p1=p1[1]
M_p1=p1[2]
PM_p1=p1[3]
mu_p1=p1[4]
teta_p1=p1[5]

x_p2=p2[0]
y_p2=p2[1]
M_p2=p2[2]
PM_p2=p2[3]
mu_p2=p2[4]
teta_p2=p2[5]

Km=teta_p1+PM_p1
Kp=teta_p2-PM_p2

teta_p3=(Km+Kp)/2
PM_p3=(Km-Kp)/2
M_p3=invPM(PM_p3)
mu_p3=mu(M_p3)
P_p3=P0_MOC/((1+(gamma_med-1)/2*M_p3**2)**(gamma_med/(gamma_med-1)))
#P_p3=P0_MOC/((1+(gamma2-1)/2*M_p3**2)**(gamma2/(gamma2-1)))


slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2)
#slopeCp=tan((teta_p3+teta_p2+mu_p3+mu_p2)/2+beta_ref)

slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2)
#slopeCm=tan((teta_p3+teta_p1-mu_p3-mu_p1)/2+beta_ref)


x_p3=(y_p1-slopeCm*x_p1+slopeCp*x_p2-y_p2)/(slopeCp-slopeCm)
y_p3=y_p2+slopeCp*(x_p3-x_p2)

p3=[x_p3,y_p3,M_p3,PM_p3,mu_p3,teta_p3,P_p3]

out1=np.zeros((1,7))
out2=np.zeros((1,7))
shock1=np.zeros((1,7))
shock2=np.zeros((1,7))

if y_p3>L:
```

```
if y_p1>L and y_p2<L:

    x_outlet=x_p2+(L-y_p2)/slopeCp

    if x_outlet<x_lim:

        deltax2=x_outlet-x_p2
        deltax3=x_outlet-x_p3
        deltay2=L-y_p2
        deltay3=L-y_p3

        dist2=sqrt(deltax2**2+deltay2**2)
        dist3=sqrt(deltax3**2+deltay3**2)

        dist=dist2+dist3

        teta_out=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
        PM_out=teta_out-teta_p2+PM_p2
        M_out=invPM(PM_out)
        mu_out=mu(M_out)
        P_out=P0_MOC/((1+(gamma_med-1)/2*M_out**2)**(gamma_med/(gamma_

        out1[0,:]=[x_outlet,L, M_out, PM_out, mu_out, teta_out,P_out]



if y_p1<L and y_p2<L:

    x_outlet2=x_p2+(L-y_p2)/slopeCp
    x_outlet1=x_p1+(L-y_p1)/slopeCm


    if x_outlet2<x_lim:

        deltax2=x_outlet2-x_p2
        deltax3=x_outlet2-x_p3
        deltay2=L-y_p2
        deltay3=L-y_p3

        dist2=sqrt(deltax2**2+deltay2**2)
        dist3=sqrt(deltax3**2+deltay3**2)

        dist=dist2+dist3
```

```
            teta_out2=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
            PM_out2=teta_out2-teta_p2+PM_p2
            M_out2=invPM(PM_out2)
            mu_out2=mu(M_out2)
            P_out2=P0_MOC/((1+(gamma_med-1)/2*M_out2**2)**(gamma_med/(gamm

            out2[0,:]=[x_outlet2,L, M_out2, PM_out2, mu_out2, teta_out2,P_

        if x_outlet1<x_lim:

            deltax1=x_outlet1-x_p1
            deltax3=x_outlet1-x_p3
            deltay1=L-y_p1
            deltay3=L-y_p3

            dist1=sqrt(deltax1**2+deltay1**2)
            dist3=sqrt(deltax3**2+deltay3**2)

            dist=dist1+dist3

            teta_out1=(1-dist1/dist)*teta_p1+(1-dist3/dist)*teta_p3
            PM_out1=-teta_out1+teta_p1+PM_p1
            M_out1=invPM(PM_out1)
            mu_out1=mu(M_out1)
            P_out1=P0_MOC/((1+(gamma_med-1)/2*M_out1**2)**(gamma_med/(gamm

            out1[0,:]=[x_outlet1,L, M_out1, PM_out1, mu_out1, teta_out1,P_


if det_h+tan(beta_ref+teta)*(x_p3-2*pi*Radius)==y_p3:

    if y_p3<=L and y_p3>=det_h:

        shock1[0,:]=p3

if det_h+tan(beta_ref+teta)*(x_p3-2*pi*Radius)>y_p3:

    #entrambi i punti prima dello shock

    if det_h+tan(beta_ref+teta)*(x_p1-2*pi*Radius)<y_p1 and det_h+tan(beta
```

```
x_shock1=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCm*x_p1-y_p1)/
y_shock1=det_h+tan(beta_ref+teta)*(x_shock1-2*pi*Radius)


x_shock2=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCp*x_p2-y_p2)/
y_shock2=det_h+tan(beta_ref+teta)*(x_shock2-2*pi*Radius)

if y_shock1<=L and y_shock1>=det_h:

    deltax1=x_shock1-x_p1
    deltax3=x_shock1-x_p3
    deltay1=y_shock1-y_p1
    deltay3=y_shock1-y_p3

    dist1=sqrt(deltax1**2+deltay1**2)
    dist3=sqrt(deltax3**2+deltay3**2)

    dist=dist1+dist3

    teta_sh1=(1-dist1/dist)*teta_p1+(1-dist3/dist)*teta_p3
    PM_sh1=-teta_sh1+teta_p1+PM_p1
    M_sh1=invPM(PM_sh1)
    mu_sh1=mu(M_sh1)
    P_sh1=P0_MOC/((1+(gamma_med-1)/2*M_sh1**2)**(gamma_med/(gamma_

    shock1[0,:]=[x_shock1,y_shock1, M_sh1, PM_sh1, mu_sh1, teta_sh1


if y_shock2<=L and y_shock2>=det_h:

    deltax2=x_shock2-x_p2
    deltax3=x_shock2-x_p3
    deltay2=y_shock2-y_p2
    deltay3=y_shock2-y_p3

    dist2=sqrt(deltax2**2+deltay2**2)
    dist3=sqrt(deltax3**2+deltay3**2)

    dist=dist2+dist3

    teta_sh2=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
    PM_sh2=teta_sh2-teta_p2+PM_p2
    M_sh2=invPM(PM_sh2)
```

```
            mu_sh2=mu(M_sh2)
            P_sh2=P0_MOC/((1+(gamma_med-1)/2*M_sh2**2)**(gamma_med/(gamma_

            shock2[0,:]=[x_shock2,y_shock2, M_sh2, PM_sh2, mu_sh2, teta_sh2

        #p1 prima e p2 dopo

        if det_h+tan(beta_ref+teta)*(x_p1-2*pi*Radius)<y_p1 and det_h+tan(beta

            x_shock1=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCm*x_p1-y_p1)/
            y_shock1=det_h+tan(beta_ref+teta)*(x_shock1-2*pi*Radius)

            if y_shock1<=L and y_shock1>=det_h:

                deltax1=x_shock1-x_p1
                deltax3=x_shock1-x_p3
                deltay1=y_shock1-y_p1
                deltay3=y_shock1-y_p3

                dist1=sqrt(deltax1**2+deltay1**2)
                dist3=sqrt(deltax3**2+deltay3**2)

                dist=dist1+dist3

                teta_sh1=(1-dist1/dist)*teta_p1+(1-dist3/dist)*teta_p3
                PM_sh1=-teta_sh1+teta_p1+PM_p1
                M_sh1=invPM(PM_sh1)
                mu_sh1=mu(M_sh1)
                P_sh1=P0_MOC/((1+(gamma_med-1)/2*M_sh1**2)**(gamma_med/(gamma_

                shock1[0,:]=[x_shock1,y_shock1, M_sh1, PM_sh1, mu_sh1, teta_sh1

        #p1 dopo e p2 prima

        if det_h+tan(beta_ref+teta)*(x_p1-2*pi*Radius)>y_p1 and det_h+tan(beta

            x_shock2=(det_h-tan(beta_ref+teta)*2*pi*Radius+slopeCp*x_p2-y_p2)/
```

```
            y_shock2=det_h+tan(beta_ref+teta)*(x_shock2-2*pi*Radius)


            if y_shock2<=L and y_shock2>=det_h:

                deltax2=x_shock2-x_p2
                deltax3=x_shock2-x_p3
                deltay2=y_shock2-y_p2
                deltay3=y_shock2-y_p3

                dist2=sqrt(deltax2**2+deltay2**2)
                dist3=sqrt(deltax3**2+deltay3**2)

                dist=dist2+dist3

                teta_sh2=(1-dist2/dist)*teta_p2+(1-dist3/dist)*teta_p3
                PM_sh2=teta_sh2-teta_p2+PM_p2
                M_sh2=invPM(PM_sh2)
                mu_sh2=mu(M_sh2)
                P_sh2=P0_MOC/((1+(gamma_med-1)/2*M_sh2**2)**(gamma_med/(gamma_

                shock2[0,:]=[x_shock2,y_shock2, M_sh2, PM_sh2, mu_sh2, teta_sh2


    return p3,out1,out2,shock1,shock2

m_tot=mol_O2*32+mol_N2*28+mol_H2*2

gas=ct.Solution('gri30_highT.yaml')
#gas.TPX=300,1e5,'H2:2 O2:1 N2:3.76'
gas.TPX=T0,p0,'H2:2 O2:1 N2:3.76'

cp=gas.cp_mass
cv=gas.cv_mass

R=cp-cv
gamma=cp/cv



check=0
```

```
#Xref_new=0


#inert-like gasses
P1=5e5
T1=1500
M1=3.5


while check==0:


    T=T0-0.5*Vinj**2/cp
    p=p0/((T0/T)**(gamma/(gamma-1)))

    #q='H2:' +str(mol_H2)+' O2:'+str(mol_O2)+' N2:'+str(mol_N2)
    q='H2:2 O2:1 N2:3.76'
    mech='gri30_highT.yaml'
    cj_speed = CJspeed(p, T, q, mech)
    gas = PostShock_eq(cj_speed, p, T, q, mech)


    #speed tringles
    U=sqrt((cj_speed**2-Vinj**2))
    beta_ref=atan(Vinj/U)
    L1=det_h/tan(beta_ref)
    L2=det_h*tan(beta_ref)

    Xref=Radius*2*pi-L1

    #post shock explosives
    X=gas.X
    P2=gas.P
    T2=gas.T
    rho2=gas.density_mass
    cp2=gas.cp_mass
    cv2=gas.cv_mass

    gamma2=cp2/cv2
    R2=gas.cp_mass-gas.cv_mass

    M2=1
```

```
a2=soundspeed_fr(gas)

W2=M2*a2
V2x=U-W2*cos(beta_ref)
V2y=W2*sin(beta_ref)
V2=sqrt(V2x**2+V2y**2)

M2_lab=V2/a2

#T0_MOC=T2*(1+(gamma2-1)/2*M2_lab**2)
#P0_MOC=P2*((1+(gamma2-1)/2*M2_lab**2)**(gamma2/(gamma2-1)))


T0_MOC=T2*(1+(gamma2-1)/2*M2**2)
P0_MOC=P2*((1+(gamma2-1)/2*M2**2)**(gamma2/(gamma2-1)))
#rho0_MOC=rho2*((1+(gamma2-1)/2*M2**2)**(1/(gamma2-1)))


check2=0


PM2= lambda M: sqrt((gamma2+1)/(gamma2-1))*atan(sqrt((gamma2-1)/(gamma2+1)
invPM2=inversefunc(PM2, domain=[1,30])


while check2==0:

    #inert-like gas

    gas1=ct.Solution(mech)
    gas1.TPX=T1,P1,X

    cp1=gas1.cp_mass
    cv1=gas1.cv_mass

    gamma1=cp1/cv1

    A=(gamma1+1)/(2*gamma1)
    P3_max=P1*(M1**2/A+1-1/A)
    B=(gamma2-1)/2
    M3_min=sqrt((1+B)/(B*(P3_max/P2)**((gamma2-1)/gamma2))-1/B)

    if M3_min<1:
```

```
    M3_min=1.2

#M3_min=round(M3_min,3)
print('M3 min = ',M3_min)

def equations(vars):
    P3, delta3, M3, teta, delta2  = vars

    eq1 = delta3-PM2(M3)
    eq2 = P3/P2-((1+0.5*(gamma2-1))/(1+0.5*(gamma2-1)*M3**2))**(gamma2
    eq3 = P3/P1-1-2*gamma1/(gamma1+1)*(M1**2*(sin(teta))**2-1)
    eq4 = tan(delta2)-(2*1/tan(teta)*(M1**2*(sin(teta))**2-1)/(M1**2*(
    eq5 = delta2-delta3


    return [eq1, eq2, eq3, eq4, eq5]

guess= [15e5, 35*pi/180, M3_min, 50*pi/180, 35*pi/180]

result =fsolve(equations, guess)

P3, delta, M3, teta, delta2 = result

x_lim=2*pi*Radius+(L-det_h)/tan(beta_ref+teta)

print('delta slip-line = ',delta*180/pi)
print('delta slip-line 2 = ',delta2*180/pi)
print('teta shock = ',teta*180/pi)
print('M3 = ',M3)
print('P3 = ',P3/1e5, ' bar')
print('')


#defining an average gamma

gamma_med=(gamma2+gamma1)/2
cp_med=(cp2+cp1)/2
cv_med=cp_med/gamma_med
R_med=cp_med-cv_med


PM= lambda M: sqrt((gamma_med+1)/(gamma_med-1))*atan(sqrt((gamma_med-1
#PM= lambda M: sqrt((gamma2+1)/(gamma2-1))*atan(sqrt((gamma2-1)/(gamma
```

```
invPM=inversefunc(PM, domain=[1,30])



#PM expansion fan nord

mu= lambda M: asin(1/M)
invmu=lambda mu: 1/sin(mu)

'''
dPM=np.linspace(0,delta,npoints)

dM=np.ones(npoints)

for i in range(npoints):
    dM[i]=invPM(dPM[i])

dmu=np.arcsin(1/dM)
#dteta=beta_ref+dPM
dteta=dPM
'''

M_max=invPM2(delta)

dM=np.linspace(M2,M_max,npoints)

dPM=np.zeros(npoints)
dmu=np.zeros(npoints)

for i in range(npoints):

    dPM[i]=PM2(dM[i])
    dmu[i]=mu(dM[i])

dteta=beta_ref+dPM
'''
dteta=np.ones(npoints)
dteta[0]=beta_ref

for i in range(npoints-1):

    dteta[i+1]=dteta[i]+dPM[i+1]-dPM[i]
'''
```

```
#PM expansion fan sud


'''
dPM2=np.linspace(0,beta_ref,npoints)

dM2=np.ones(npoints)

for i in range(npoints):
    dM2[i]=invPM(dPM2[i])

dmu2=np.arcsin(1/dM2)
#dteta2=beta_ref-dPM2
dteta2=-dPM2

'''

M_max=invPM2(beta_ref)

dM2=np.linspace(M2,M_max,npoints)

dPM2=np.zeros(npoints)
dmu2=np.zeros(npoints)

for i in range(npoints):

    dPM2[i]=PM2(dM2[i])
    dmu2[i]=mu(dM2[i])

dteta2=-dPM2+beta_ref


'''
dteta2=np.zeros(npoints)

dteta2[0]=beta_ref

for i in range(npoints-1):

    dteta2[i+1]=dteta2[i]-dPM2[i+1]+dPM2[i]
```

```
,,,



#MOC

plt.figure(1)

save_points=np.empty((0,7)) #x,y,M,PM,mu,teta,pressure

outlet=np.empty((0,7))
wall_index_sud=[]
wall_index_nord=[]

#exp fan nord

p2=[L2,0,dM2[1],dPM2[1],dmu2[1],dteta2[1]]

for i in range(npoints-1):

    p1=[0,det_h,dM[i+1],dPM[i+1],dmu[i+1],dteta[i+1]]

    [p3,out1,out2]=center_MOC(p1,p2)

    save_points=np.append(save_points,[p3],0)

    if out1[0,0]!=0:

        outlet=np.append(outlet,out1,0)

    if out2[0,0]!=0:

        outlet=np.append(outlet,out2,0)


    print_MOC(p1,p2,p3)

    p2=p3



#1st wall point nord
```

```
p2=save_points[-1,:]

[p3,out]=wallp_nord(p2)

save_points=np.append(save_points,[p3],0)

wall_index_nord.append(save_points.shape[0]-1)

if out[0,0]!=0:

    outlet=np.append(outlet,out,0)

plot_pnord(p2,p3)


#exp fan sud

p1=[0,det_h,dM[1],dPM[1],dmu[1],dteta[1]]

exp_fan_sud=np.ones((npoints-1,7))

for i in range(npoints-1):

    p2=[L2,0,dM2[i+1],dPM2[i+1],dmu2[i+1],dteta2[i+1]]

    [p3,out1,out2]=center_MOC(p1,p2)

    if i!=0:

        exp_fan_sud[i-1,:]=p3

    if out1[0,0]!=0:

        outlet=np.append(outlet,out1,0)

    if out2[0,0]!=0:

        outlet=np.append(outlet,out2,0)


    print_MOC(p1,p2,p3)
```

```
    p1=p3



#1st wall point sud

p1=exp_fan_sud[-2,:]

p3=wallp_sud(p1)

exp_fan_sud[-1,:]=p3

plot_psud(p1,p3)


#MOC part 1

imax=npoints-1
sud_index=[]
sud_index.append(0)

for j in range(npoints-1):

    save_points=np.append(save_points, [exp_fan_sud[j,:]], 0)
    sud_index.append(save_points.shape[0]-1)

    for i in range(imax):

        p1=save_points[sud_index[-2]+1+i,:]
        p2=save_points[sud_index[-1]+i,:]

        [p3,out1,out2]=center_MOC(p1,p2)

        save_points=np.append(save_points,[p3],0)

        if out1[0,0]!=0:

            outlet=np.append(outlet,out1,0)

        if out2[0,0]!=0:

            outlet=np.append(outlet,out2,0)
```

```
            print_MOC(p1,p2,p3)

        #wall point nord

        p2=save_points[-1,:]

        [p3,out]=wallp_nord(p2)

        save_points=np.append(save_points,[p3],0)

        wall_index_nord.append(save_points.shape[0]-1)

        if out[0,0]!=0:

            outlet=np.append(outlet,out,0)

        plot_pnord(p2,p3)

        imax=imax+1

'''
for i in range(save_points.shape[0]):
    print(i)
    print(save_points[i,:])
    print('')
'''

#creating next sud wall point

p1=save_points[sud_index[-1]+1,:]

p3=wallp_sud(p1)

save_points=np.append(save_points,[p3],0)

sud_index.append(save_points.shape[0]-1)

plot_psud(p1,p3)


#MOC part 2
```

```
imax=sud_index[-1]-sud_index[-2]-2

PP=p0+2e5

shock=np.empty((0,7))

j=0

while PP>p:


    for i in range(imax-j):

        p1=save_points[sud_index[-2]+2+i,:]
        p2=save_points[sud_index[-1]+i,:]

        #[p3,out1,out2]=center_MOC(p1,p2)

        [p3, out1, out2, sh1, sh2]=MOC_3(p1,p2)

        save_points=np.append(save_points,[p3],0)

        if out1[0,0]!=0:

            outlet=np.append(outlet,out1,0)

        if out2[0,0]!=0:

            outlet=np.append(outlet,out2,0)

        if sh1[0,0]!=0:

            shock=np.append(shock, sh1, 0)

        if sh2[0,0]!=0:

            shock=np.append(shock, sh2, 0)


        print_MOC(p1,p2,p3)


    #wall point nord
```

```
    p2=save_points[-1,:]

    [p3,out]=wallp_nord(p2)

    save_points=np.append(save_points,[p3],0)

    wall_index_nord.append(save_points.shape[0]-1)

    if out[0,0]!=0:

        outlet=np.append(outlet,out,0)

    plot_pnord(p2,p3)


    #wall points sud

    p1=save_points[sud_index[-1]+1,:]

    p3=wallp_sud(p1)

    save_points=np.append(save_points,[p3],0)

    sud_index.append(save_points.shape[0]-1)

    plot_psud(p1,p3)

    PP=p3[6]



'''
for i in range(save_points.shape[0]):
    print(i)
    print(save_points[i,:])
    print('')

'''


#Evaluating new Xref
```

```
x_p2=save_points[sud_index[-2],0]
x_p1=save_points[sud_index[-1],0]

P_p2=save_points[sud_index[-2],6]
P_p1=save_points[sud_index[-1],6]

dP1=p-P_p1
dP2=P_p2-p
dPtot=P_p2-P_p1

Xref_new=(1-dP2/dPtot)*x_p2+(1-dP1/dPtot)*x_p1

beta_ref_new=atan((det_h)/(2*pi*Radius-Xref_new))

print(Xref_new)


#MOC part 3

xx=0
z=0

#shock=np.empty((0,7))

#while imax-j>=1:
while xx<2*pi*Radius:

    for i in range(imax-j):

        p1=save_points[sud_index[-2]+2+i,:]
        p2=save_points[sud_index[-1]+i,:]

        [p3,out1,out2,sh1,sh2]=MOC_3(p1,p2)
        #[p3,out1,out2]=center_MOC(p1,p2)

        save_points=np.append(save_points,[p3],0)

        if out1[0,0]!=0:

            outlet=np.append(outlet,out1,0)

        if out2[0,0]!=0:
```

```
            outlet=np.append(outlet,out2,0)
        if sh1[0,0]!=0:
            shock=np.append(shock,sh1,0)
        if sh2[0,0]!=0:
            shock=np.append(shock,sh2,0)

        print_MOC(p1,p2,p3)

    #wall points sud

    p1=save_points[sud_index[-1]+1,:]
    p3=wallp_sud3(p1)
    save_points=np.append(save_points,[p3],0)
    sud_index.append(save_points.shape[0]-1)
    plot_psud(p1,p3)
    xx=p3[0]

    j=j+1

#plots
x=np.linspace(0,2*Radius*pi+L2,3)
```

```
y=np.zeros(3)

plt.plot(x,y,'-b')

x=np.linspace(0,x_lim,3)
y=L*np.ones(3)

plt.plot(x,y,'b-')

x=np.linspace(Xref_new,2*pi*Radius,10)
y=tan(beta_ref_new)*(x-Xref_new)

plt.plot(x,y)

x=0*np.ones(10)
y=np.linspace(0,det_h,10)

plt.plot(x,y)

x=np.linspace(0,L2,10)
y=det_h-tan(pi/2-beta_ref)*(x)

plt.plot(x,y)

x=(Xref+L1)*np.ones(10)
y=np.linspace(0,det_h,10)

plt.plot(x,y)

x=np.linspace(2*pi*Radius,2*pi*Radius+L2,10)
y=det_h-tan(pi/2-beta_ref)*np.linspace(0,L2,10)

plt.plot(x,y)

x=np.linspace(0,x_lim-2*pi*Radius,10)
y=det_h+tan(beta_ref+teta)*x

plt.plot(x,y)

x=np.linspace(0,(L-det_h)/tan(beta_ref+delta),10)
y=det_h+tan(beta_ref+delta)*x

plt.plot(x,y)
```

```
x=np.linspace(2*pi*Radius,x_lim,10)
y=det_h+tan(beta_ref+teta)*(x-2*pi*Radius)

plt.plot(x,y)
plt.axis('scaled')
plt.grid(True)
#plt.xticks(np.arange(0,x_lim,0.01))
#plt.ylim([0.195,0.205])
#plt.xlim([0.5,0.515])

#plt.text(0.504,0.1975,'1',fontsize=20)
#plt.text(0.511,0.203,'2',fontsize=20)
#plt.text(0.5075,0.2002,'3',fontsize=20)


plt.savefig('RDE_MOC.pdf',bbox_inches='tight')

plt.show()



shock=shock[np.argsort(shock[:, 0])]

M1_new=shock[0,2]

P1_new= shock[0,6]

T1_new=T0_MOC/(1+(gamma1-1)/2*M1_new**2)




M1_new=sum(shock[:, 2])/shock.shape[0]
#T1_new=T0_MOC/(1+(gamma_med-1)/2*M1_new**2)
T1_new=T0_MOC/(1+(gamma_med-1)/2*M1_new**2)
#T1_new=sum(T1_os)/T1_os.shape[0]
P1_new=P0_MOC/((1+(gamma_med-1)/2*M1_new**2)**(gamma_med/(gamma_med-1)
#P1_new=shock[0,6]

print('M1 new = ', M1_new)
print('M1 old = ', M1)
print('P1 new = ', P1_new/1e5)
```

```
        print('P1 old = ', P1/1e5)
        print('T1 new = ', T1_new)
        print('T1 old = ', T1)
        print('')



        if abs(T1-T1_new)<=10 and abs(M1_new-M1)<=0.05 and abs(P1-P1_new)<=10e
            check2=1

        else:

            T1=T1_new
            P1=P1_new
            M1=M1_new


    if abs(Xref-Xref_new)<0.001:

        check=1

    else:
        Vinj=cj_speed*sin(beta_ref_new)
        print('')
        print('Xref difference = ',abs(Xref-Xref_new))
        print('New Vinj = ', Vinj)

        print('')



        #oblique shock equations

#x,y,M,PM,mu,teta,pressure

#extrapolating data pre-oblique shock

shock=shock[np.argsort(shock[:, 0])]

x_os=shock[1:-1,0]-2*pi*Radius
y_os=shock[1:-1,1]
M_sh=shock[1:-1,2]
teta_os=shock[1:-1,5]
```

```
#M_os=np.sqrt(np.absolute(2/((gamma1+1)/M_sh**2-gamma1+1)))
M_os=M_sh

P_os=P0_MOC/(1+(gamma1-1)/2*M_os**2)**(gamma1/(gamma1-1))

T_os=T0_MOC/(1+(gamma1-1)/2*M_os**2)

beta_os=teta+beta_ref-teta_os


#evaluating post-oblique shock data

P2_os=P_os*((2*gamma1*M_os**2*(np.sin(beta_os))**2)-gamma1+1)/(gamma1+1)


x2_os=(L-y_os)/np.tan(beta_ref+delta)+x_os


A=(gamma1-1)
B=M_os**2*(np.sin(beta_os))**2
M2_os=np.sqrt(1/(np.sin(beta_os-(delta+beta_ref-teta_os)))**2*(A*B+2)/(2*gamma



C=2*(gamma1-1)/(gamma1+1)**2

T2_os=T_os*((2*gamma1*B-A)*(A*B+2)/((gamma1+1)**2*B))


a_os=np.ones(T2_os.shape[0])
rho_os=np.ones(T2_os.shape[0])

for i in range(T2_os.shape[0]):

    gas=ct.Solution(mech)
    gas.TPX=T2_os[i],P2_os[i],X
    a_os[i]=soundspeed_fr(gas)
    rho_os[i]=gas.density_mass

W2_os=M2_os*a_os
V2_os_x=abs(W2_os*cos(delta+beta_ref)-U)
```

```
V2_os_y=W2_os*sin(delta+beta_ref)
V2_os=np.sqrt(V2_os_x**2+V2_os_y**2)
M_os_lab=V2_os/a_os


P_tot_os=P2_os*(1+(gamma1-1)/2*M_os_lab**2)**(gamma1/(gamma1-1))
T_tot_os=T2_os*(1+(gamma1-1)/2*M_os_lab**2)

#P_tot_os=P2_os*(1+(gamma1-1)/2*M2_os**2)**(gamma1/(gamma1-1))
#T_tot_os=T2_os*(1+(gamma1-1)/2*M2_os**2)



outlet_os=np.ones((x2_os.shape[0],7))
outlet_os[:,0]=x2_os
outlet_os[:,1]=P2_os
outlet_os[:,2]=y_os
outlet_os[:,3]=x_os
outlet_os[:,4]=P_tot_os
outlet_os[:,5]=T_tot_os
outlet_os[:,6]=T2_os

outlet_os=outlet_os[np.argsort(outlet_os[:, 0])]
x2_os=outlet_os[:,0]
P2_os=outlet_os[:,1]
y_os=outlet_os[:,2]
x_os=outlet_os[:,3]
P_tot_os=outlet_os[:,4]
T_tot_os=outlet_os[:,5]
T2_os=outlet_os[:,6]

#outlet data


out_x=outlet[:,0]
out_M1=outlet[:,2]
out_teta=outlet[:,5]

gamma=np.linspace(gamma2, gamma1 , out_x.shape[0])

#out_M=np.sqrt(np.absolute(2/((gamma+1)/out_M1**2-gamma+1)))
out_M=out_M1
```

```
out_P=P0_MOC/(1+(gamma_med-1)/2*out_M**2)**(gamma_med/(gamma_med-1))

out_T=T0_MOC/(1+(gamma_med-1)/2*out_M**2)


out_a=np.ones(out_T.shape[0])
rho_out2=np.ones(out_T.shape[0])

for i in range(out_T.shape[0]):

    gas=ct.Solution(mech)
    gas.TPX=out_T[i],out_P[i],X
    out_a[i]=soundspeed_fr(gas)
    rho_out2[i]=gas.density_mass

out_W2=out_M1*out_a



out_Vx=abs(out_W2*np.cos(out_teta)-U)
out_Vy=out_W2*np.sin(out_teta)
out_V=np.sqrt(out_Vx**2+out_Vy**2)

out_M_lab=out_V/out_a


out_P_tot=out_P*(1+(gamma_med-1)/2*out_M_lab**2)**(gamma_med/(gamma_med-1))
out_T_tot=out_T*(1+(gamma_med-1)/2*out_M_lab**2)


x_out=np.concatenate((x2_os,out_x))
P_out=np.concatenate((P2_os,out_P))
T_out=np.concatenate((T2_os,out_T))
P_tot_out=np.concatenate((P_tot_os,out_P_tot))
T_tot_out=np.concatenate((T_tot_os,out_T_tot))

rho_out_final=np.concatenate((rho_os,rho_out2))

v_out=np.concatenate((V2_os,out_V))
vy_out=np.concatenate((V2_os_y, out_Vy))
rho_out=P_out/(R_med*T_out)

mass_avg_T=sum(rho_out*v_out*T_out)/sum(rho_out*v_out)
```

```
print('Mass averaged Temperature ', mass_avg_T)

mass_avg_P=sum(rho_out*v_out*P_out)/sum(rho_out*v_out)
print('Mass averaged pressure ', mass_avg_P/1e5)

print('')
mass_avg_T=sum(rho_out_final*v_out*T_out)/sum(rho_out_final*v_out)
print('Mass averaged Temperature ', mass_avg_T)

mass_avg_P=sum(rho_out_final*v_out*P_out)/sum(rho_out_final*v_out)
print('Mass averaged pressure ', mass_avg_P/1e5)

#print(v_out/(np.sqrt(gamma_med*R_med*T_out)))


#plots

plt.figure(2)
x=np.linspace(0,x_lim-2*pi*Radius,10)
y=det_h+tan(beta_ref+teta)*x

plt.plot(x,y)

x=np.linspace(0,(L-det_h)/tan(beta_ref+delta),10)
y=det_h+tan(beta_ref+delta)*x

plt.plot(x,y)

for i in range(x2_os.shape[0]):
    x=np.linspace(x_os[i],x2_os[i],4)
    y=y_os[i]+tan(beta_ref+delta)*(x-x_os[i])
    plt.plot(x,y)

plt.figure(3)
plt.plot(x_out,P_out/1e5)
plt.xlabel('x [m]')
plt.ylabel('Outlet pressure [bar]')
plt.show



plt.figure(4)
plt.plot(x_out,T_out)
```

```
plt.xlabel('x [m]')
plt.ylabel('Outlet Temperature [K]')
plt.show


plt.figure(5)
plt.plot(x_out,P_tot_out/1e5)


y=p0/1e5*np.ones(3)
x=np.linspace(x_out[0],x_out[-1],3)
plt.plot(x,y,'r--')

plt.xlabel('x [m]')
plt.ylabel('Total Outlet pressure [bar]')
plt.show

plt.figure(6)
plt.plot(x_out,T_tot_out)
plt.xlabel('x [m]')
plt.ylabel('Total Outlet Temperature [K]')
plt.show


p_in=save_points[sud_index, 6]
x_in=save_points[sud_index, 0]

plt.figure(7)
plt.plot(x_in,p_in/1e5)
plt.plot(x_in, p0/1e5*np.ones(x_in.shape[0]))
plt.xlabel('x [m]')
plt.ylabel('Inlet Pressure [bar]')

plt.show
```

Affidavit

Ich versichere wahrheitsgemäß, die Arbeit selbstständig verfasst, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde sowie die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet zu haben.

Karlsruhe, 31/01/2025 _____

Riccardo Faggion