POLITECNICO DI TORINO

Dipartimento di Ingegneria Meccanica ed Aerospaziale

Master Thesis in Aerospace Engineering

Academic Year 2017/2018

Inverse heat conduction method and heat flux error estimation in rocket engines

Author
Cosimo Ciracì

Advisor:
Prof. Dario Giuseppe Pastrone

Supervisor TUM:
Nikolaos Perakis, M. Sc.

Marzo 2018

Acknowledgements

I would like to thank my advisor Prof. Dario Giuseppe Pastrone for the opportunity that he has given to me to do my master thesis at TUM and for his availability during the 6 months of the thesis.

I would also like to express my gratitude to my supervisor Nikolaos Perakis, M. Sc. for his support, fruitful discussions about the problems that I encountered, availability and also for all the advice that he has given to me in order to improve my work.

****

Abstract

The propellant combination of methane and oxygen is one of the most promising candidate for future rocket propulsion engines. In order to obtain a more deep understanding of the performance properties of this new propellant, intensive efforts are placed in its experimental characterization.
The ongoing research projects at the Institute for Flight Propulsion (LTF) at the Technische Universität München (TUM) have moved in this direction. The Institute of Flight Propulsion of the Technical University of Munich (Department of Rocket Propulsion) has designed, developed, and it is testing a combustion chambers for rocket engine applications propelled with the aforementioned combination of propellant.
A crucial aspect of these devices is the correct prediction of the temperature and heat flux distribution in order to satisfy structural limits and characterize the injectors.
The LTF at TUM has introduced a new tool (an inverse solver code) for the solution of the inverse heat conduction problem in order to evaluate temperatures and heat fluxes based on transient temperature measurements within the combustion chamber wall.
The background of this thesis is the aforementioned inverse solver. The main interest is the error estimations on heat flux computed using the inverse solver. In fact due to the ill-posedness of inverse problems, it is of primary importance to have an estimation of the commited error on estimated heat fluxes (sources of error will be analyzed and errors will be estimated basically through a sensitivity analysis).
Further studies were carried out for the optimization of the inverse solver (reduction of computational time) and the validation of the code.

Contents

List of Figures

List of Tables

List of symbols

A	[m^2]	Cross-sectional area
$\underline{\mathbf{A}}$	[−]	Coefficients matrix
$\mathbf{b}$	[K]	Boundary conditions vector
Bi	[−]	Biot number
c	[m/s]	Effective exaust velocity
c^*	[m/s]	Characteristic velocity
c_p	[J/kgK]	Specific heat capacity
d	[K^2m^2/W]	Search step size
F	[N]	Force
Fo	[−]	Fourier number
g	[m/s^2]	Standard gravitational acceleration
h	[W/m^2K]	Natural convection coefficient
I_{sp}	[s]	Specific impulse
I_t	[Ns]	Total impulse
$\underline{\mathbf{J}}_S$	[Km^2/W]	Sensitivity matrix
$\underline{\mathbf{J}}_J$	[Km^2/W]	Jacobi matrix
K	[−]	Kurtosis
$\dot{m}$	[kg/s]	Mass flow rate
O/F	[−]	Oxidizer to Fuel Ratio
p	[bar]	Pressure
$\mathbf{P}$	[W/m^2]	Vector of heat flux parameters
Pr	[−]	Prandtl number
$\dot{q}$	[W/m^2]	Heat flux
r	[−]	Recovery factor
R	[K^2s]	Residual function
S	[−]	Skewness
t	[s]	Time

T	[K]	Temperature
T_c	[K]	Computed temperature
T_m	[K]	Measured temperature
T_s	[K]	Smoothed temperature
v	[m/s]	Hot gas velocity
α	[m^2/s]	Thermal diffusivity
β	[W/m^2K^2]	Search step size
γ	[−]	Conjugation coefficient
Δt	[s]	Time step
λ	[W/mK]	Heat conduction
ρ	[kg/m^3]	Density
σ		Standard deviation
BC		Buondary Condition
CGM		Conjugate Gradient Method
LTF		Lehrstuhl für Turbomaschinen und Flugantriebe
TUM		Technische Universität München

Chapter 1

Introduction

1.1 Basics of chemical rocket engines

In the combustion chamber of a chemical rocket engine internal energy of the propellant is released through chemical reactions. The products of these reactions are accelerated in the nozzle. This acceleration leads to a very high velocity flow in the nozzle. These accelerated gases in the nozzle produce thrust to propell the rocket according to the momentum conservation law. The thrust, in a simplified way, it is the reaction experienced by the structure due to the ejection of matter at high velocity. It represents the same phenomenon that makes a gun recoil. In rocket propulsion relatively small masses are involved, which are carried within the vehicle and ejected at high velocities. The total impulse is defined as the force F (which can vary with time) integrated over the burning time t.

$$I_t = \int_0^t F dt \tag{1.1}$$

The specific impulse is defined the total impulse per unit weight of the propellant:

$$I_{sp} = \frac{\int_0^t F dt}{\int_0^t \dot{m} g dt} \tag{1.2}$$

Specific impulse is a measure of the efficiency of a rocket engine since for fixed total impulse, the higer the specific impulse is, the less mass of propellent will be needed. The expression of the thrust generated by a change in momentum is given in the equation below. Here thrust and mass flow are considered to be constant and the exit gas velocity v_2 uniform and axial [20].

$$F = \frac{dm}{dt} v_2 = \dot{m} v_2 \tag{1.3}$$

Eq. (1.3) is valid if the nozzle exit pressure is equal to the ambient pressure. The pressure of the surrounding fluid represents the second contribution that influences the thrust. A schematic view of a rocket engine is presented in Figure 1.1. The external pressure acts uniformly on the outer surface of a rocket chamber and the gas pressure acts on the inside of the chamber. If the exit nozzle pressure is different from atmosphere pressure Eq. 1.3

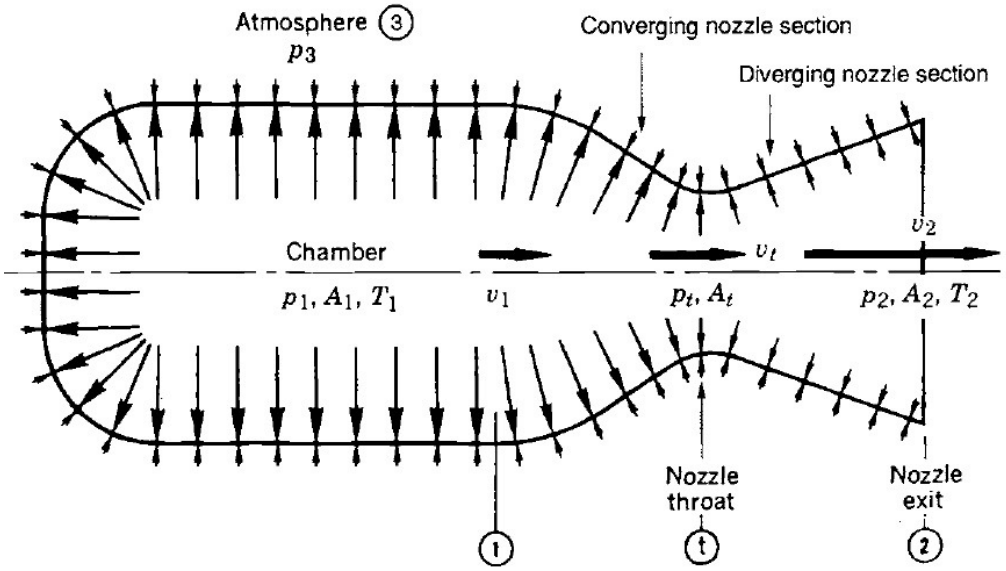


Figure 1.1: Schematic representazion of a rocket engine [20].

has to be modified as follows:

$$F = \dot{m}v_2 + A_2(p_2 - p_3) \tag{1.4}$$

where the subscript 2 is referred to the nozzle and the subscript 3 is referrend to the external atmosphere. Another important parameter is the effective exaust velocity defined as:

$$c = \frac{F}{\dot{m}} \tag{1.5}$$

Both specific impulse and effective exaust velocity can be used to characterize the efficiency of a rocket engine. Their value essentially depend on the choice of propellents and their mass ratio O/F. The O/F is defined as the ratio of oxidizer mass flow to the fuel mass flow:

$$O/F = \frac{\dot{m}_{ox}}{\dot{m}_f} \tag{1.6}$$

The interest of this thesis is direct to **methane/oxygen** combustion, for which the stoichiometric value of the oxidizer to fuel ratio is about 4. But maximum value of specific impulse are usually obtained for lower values of O/F. Since the value of the specific impulse

is also dependent on the nozzle geometry, another parameter is needed to evaluate the efficiency of a combustion independently from nozzle characteristics. It is the characteristic velocity defined as:

$$c^* = \frac{p_1 A_t}{\dot{m}} \tag{1.7}$$

Rocket propulsion systems can be classified according to the type of energy source, the basic function (booster stage, sustainer, attitude control, orbit station keeping, etc.), the type of vehicle (aircraft, missile, assisted take-off, space vehicle, etc.), size, type of propellant, type of construction, or number of rocket propulsion units used in a given vehicle. A chemical rocket engine can work with a solid or liquid propellant. Besides this a combination of solid and liquid propellant can also be used, which is known as hybrid propulsion.

1.2 Methane as rocket fuel

LOX/LH_2 and $LOX/Kerosene$ are the most used propellants in chemical rocket engines. For example Vulcain engine of Ariane 5 uses LOX/LH_2 and $LOX/Kerosene$ is used in for the boosters (RD-107), 1^{st} stage (RD-108) and 2^{nd} stage (RD-0110) of the Soyuz rocket. Storable liquid propellant combinations such as N_2O_4/N_2H_4 (NTO with hydrazine) are also common in launch vehicles and are used in the Aestus engine of Ariane 5 for example. LOX/CH_4 is the most promising propellant for the future if compared to the common aforementioned propellants.
Methane is a pure hydrocarbon extracted from natural gas and futhermore it is not toxic and not corrosive. In comparison to high hidrocarbons, methan has many advantages [19] [12]:

- Higher specific impulse (Figure 1.2).
- Better cooling properties.
- Lower pressure losses in cooling channel.
- Higher availability.
- Higher coking limits.
- Less soot deposition.

Compared to the liquid hydrogen fuel, methane has an important disadvantage which is the lower specific impulse (Figure1.2), but many advantages:

- Higher density (this allows to have more compact tanks).

- Lower vapour pressure (less stringent pressurization requirements).
- The smaller density difference with oxygen with respect to hydrogen allows to have a single shaft instead of two in the turbopumps.
- Higher boiling point (less thermal protection requirements and possibility to use nitrogen as purge gas instead of helium) [9].

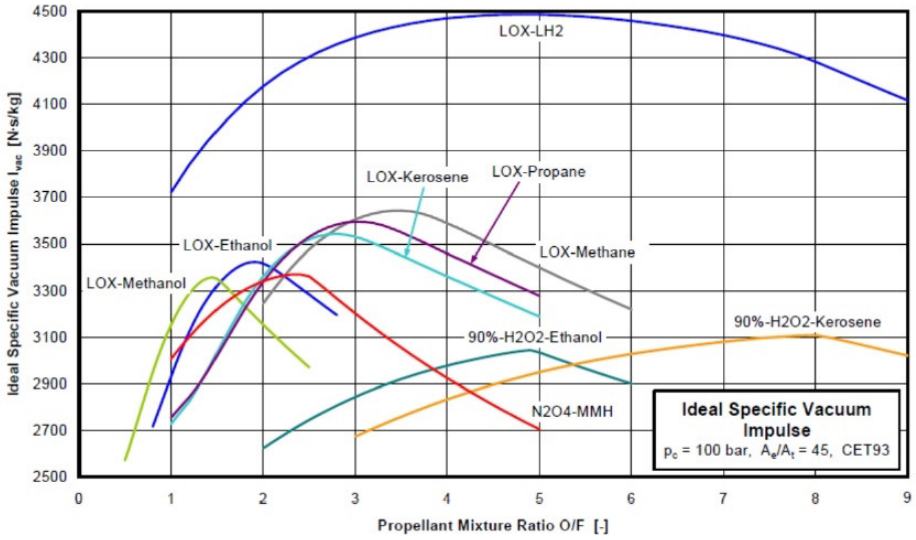


Figure 1.2: Specific impulse of various propellant combinations at ideal conditions [9].

Methane rocket engines have never flown, but extensive research is going on in many companies and resarch centers. In particular at the Institute for Flight Propulsion (LTF) at the Technische Universität München many researchers go toward this direction.

1.3 Heat Transfer in Liquid Rocket Engines

Optimization of heat transfer management is a fundamental factor in designing a rocket combustion chamber. The precise knowledge of the thermal conditions and heat flux is of major importance for the life estimation of structures under high thermal load and for the prediction of the engine performance.
For optimal cooling design the precise knowledge of the heat transfer processes in rocket engines is primary. The importance of these data is confirmed by the fact that life cycle prediction of rocket engine heavily depends on the accuracy of wall temperature predictions, where an error of 40 K may lead to 50 % life reduction in a cryogenic propellant rocket chamber [7].
Estimating correctly temperatures and heat flux is also important not only for the life of

a combustion chamber, but also to work in safe conditions (operating temperatures have to stay far from melting temperature of the material).

Despite long term experience and substantial progress in numerical simulations there is still need for experimental data to evaluate heat transfer for representative conditions, (both at hot gas and at coolant side), but also to validate numerical tools.
When developing an engine and in particular the injection system, the main factors that must be considered are the injector-wall interaction and the resulting heat transfer characteristics, the combustion effciency, and the combustion stability. In particular, from heat flux estimation in a chamber wall, it is possible to characterize the injector, since heat flux is related to the amount of combustion energy. In particular, heat flow from hot gas to chamber wall varies along axial position. Information about heat flux along axial position grants information about the combustion over axial position and this is very helpful to characterize an injector. While combustion efficiency and combustion stability can be easily determined by measuring static and dynamic chamber pressures, it is quite difficult to get detailed information on the temperature and the heat loads at the chamber walls due to the extremely harsh environment. Over the past decades, a large amount of effort has been dedicated to the improvement of the quantitative prediction ability of heat transfer in rocket combustors [14].

1.3.1 Heat flux correlations

Inside the combustion chamber, where the oxidizer and fuel flow takes place (oxygen and methane for the studies of this thesis), the heat transfer mechanism is forced convection. Heat transfer occurring is much more complicated to describe than the one for natural convection. Determination of gas side heat transfer coefficient is a very complex problem. Comparisons of analytical results with experimental heat transfer data often show disagreements. The differences are largely attributed to the assumptions for analytical calculations [11]. Many analytical expressions exist to estimate the heat transfer at the wall. These expressions are in general a function of pressure, geometry factors (injectors, igniter and chamber walls contour), type and speed of chemical reactions, turbulence and radiation. The most used analytical relation are:

- Bartz correlation. [11]
- Cinjarew correlation [18].
- Gnielinski correlation.

For example for Cinjarew formula:

$$h = 0.0198 \frac{\lambda_{gas}^{0.18} (\dot{m} c_p)^{0.82}}{d^{1.82}} \left(\frac{T_{aw}}{T_w} \right)^{0.35} \tag{1.8}$$

with:

- λ_{gas} [W/(mK)] is the thermal conductivity of the gas close to the wall.
- $\dot{m}$ [Kg/s] is the mass flow rate.
- c_p [J/(KgK)] is the specific heat capacity of the gas.
- d [m] is the diameter of the chamber for circular chamber or a quantity related to the trasversal dimensions of the chamber.
- T_{aw} [K] is the adiabatic wall temperature.
- T_w [K] is the wall temperature.

The recovery temperature is defined trough the recovery factor r.

$$r = \frac{T_{aw} - T_\infty}{T_c - T_\infty} = \sqrt{Pr} \tag{1.9}$$

Where last equivalence is true for turbulent flow. T_c is the total temperature in combustion chamber and T_∞ is the gas temperature out of the boundary layer. It is evident from Eq. (1.8), that the heat convection coefficient increases after the combustion has developed. Furthermore, if the radius is constant (or in general if the cross-sectional area is constant) the heat transfer coefficient is approximately constant along the length (after the combustion has developed). Beacause of this an almost constant heat flux is expected along z after the combustion has developed (results from inverse method confirm this behaviour).

1.3.2 Calorimetric heat flux measurements

The calorimetric method is the most used measurement method at high thermal loads. If a structure under thermal load is cooled by a cooling fluid and **the system is in thermal equilibrium** the heat flux from hot gas side into the structure is identical to the heat flux from the structure into the cooling fluid. The integral heat flux into the structure can therefore be determined by measuring the enthalpy difference of the cooling fluid between inlet and outlet of a cooling channel. The total enthalpy $h_{cf}(p,T)$ of the cooling fluid is a

function of pressure and temperature and the heat flux $\dot{q}$ is determined as follows [14]:

$$\dot{q} = \frac{\dot{m}_{cf}}{A_w}[h_{cf}(p,T)_{out} - h_{cf}(p,T)_{in}] \tag{1.10}$$

where A_w is the surface of the structure exposed to the termal load and $\dot{m}_{cf}$ is the mass flow rate of the cooling fluid. In order to obtain wall temperatures, a good knowledge of the heat transfer coefficient is required. Using the calorimetric tecnique, integral heat flux is determined with high accuracy.

1.3.3 Heat Flux Measurements with Inverse Methods

In some combustion chambers, no cooling system is present and therefore the calorimetric method cannot be used. Instead, the inverse method can be applied.
A simplified method can be used to estimate heat flux at boundaries. It is the gradient method. This method exploits the temperatures measured at differet distances from the wall and linearize them in order to obtain the value of the wall temperature.
The inverse method is based on the measurement of local temperatures in a structure **at steady thermal conditions** [14].

- Temperatures T_i^m are measured in different positions.
- Temperatures in the domain are computed solving a heat conduction problem, imposing a guessed heat flux boundary.
- With an optimization process, the heat flux boundary conditions are adjusted in order to obtain the maximum matching between computed and measured temperatures.

1.3.4 Heat flux Measurements with Transient Inverse Methods

This is the method on which this thesis is based on. The aim is to reconstruct heat flux distributions on inner chamber walls using **transient hot run temperatures**. The method will be better explained in the following chapters. This method is the only exploitable for the baseline experiments of this thesis. In fact the baseline rocket chamber is capacitive (no cooling channel) and because of this it is not possible to use calorimetric method to determine the boundary heat fluxes. Furthermore the experiments only last 3 second and

no stationary value of temperatures is reached and so it is also not possible to use the inverse method described in Section 1.3.3.

1.4 Inverse problem

"The goal of an inverse problem is to use data, that is collected or measured, to estimate unknown parameters given some assumptions about the forward model."[5] In particular, the case of interest of this thesis are Inverse Heat Transfer Problems (IHTP), and more precisely the estimation of an unknown boundary condition heat flux from the knowledge of transient temperatures in some points of the considered body. In general, inverse problems deal with the estimation of the cause (e.g heat flux boundary conditions) from observed or desired effects (e.g temperature of the body). The best approach to overcome the difficulties of inverse problems is a fruitful combination of numerical and experimental activities [18]. Inverse problems were initially not considered of physical interest because of their ill-posedness (the solutions could become unstable because of measurement errors). But after the 19550's some methods to treat these type of problems were developed and formalized. These methods are based on particular stabilization tecnhiques.

The concept of a well-posed problem, originally introduced by Hadarnard [8], requires that its solution should satisfy the following three conditions:

- The solution must exist.
- The solution must be unique.
- The solution must be stable under small changes to the input (stability condition).

In inverse heat transfer problems the first condition is surely satisfied thanks to physical reasoning (if there is a rate of change in temperature in a body, there will be a heat flux). The uniqueness of a solution can be mathematically proved only for some cases. The last condition, in transient heat transfer problems is the most critical since random errors in measured temperatures largely affect the solution [15]. This leads to the need of constructing special algorithm for its solutions.

In general, algorithms for inverse heat transfer problems are based on the iterative solution of a direct problem. A direct problem consists in the computation of the effects (in this case temperatures) known the causes (heat fluxes). If the heat fluxes boundary condition and the temperature starting condition of a body are known, it is possible to compute the temperature in the whole domain for each time step. The main problem of inverse transient heat flux problems is that the boundary condition is unknown and the temperature

is known only in some positions (the position of thermocouples). So the heat flux has to be "guessed" in order to match the measured temperatures with the one computed in the direct solver.

1.5 Scope of the thesis

As already mentioned, inverse problems are ill-posed and oscillations in temperatures measurements could introduce oscillations in the solution. Other problems related to the inverse transient problem is the low precision of the knowledge of the sensor location [14]. Furthermore thermocouples are characterized by their own accuracy which is around 1^oC. These errors in temperature measurements propagate and affect the heat flux estimation, since ,basically, heat flux is a function of the measured temperatures in the chamber. The final scope of this thesis is to evaluate the errors source and estimate an order of magnitude for each one of them. Furthermore, another issue is to determine the origin of some oscillations of the heat flux over the axial coordinate. These oscillations seem to be not physical since there is no reason for them to be there. The final aim is to understand if they are the results of some amplifications performed by the inverse method or could have other causes.

Chapter 2

Experimental set up

The algorithm is speficically implemented to solve an inverse heat trasfer problem in a transient start up of a combustion rocket chamber with a specific geometry. So it is very useful, to show the experimental set up. The rectangular modular heat sink capacitive combustion chamber for the mobile rocket test bench MoRaP is selected as reference and baseline geometry for this thesis. The entire assembly is capacitively cooled. It includes fuel and oxidizer supply as well as a pilot torch ignition system, a purge system and control and data acquisition systems (DAQ) [16]. Fuel is methane, while oxidizer is Oxygen. The modular combustion chamber with a square cross section is designed for a testing time of up to 4 s. The inner chamber dimensions are shown in Table 2.1

Chamber length	290.0 mm
Inner chamber heigth	12.0 mm
Inner chamber width	12.0 mm
Throat height	4.8 mm
Contraction ratio A_{cc}/A_{th}	2.5

Table 2.1: Inner dimensions of the chamber [4].

The single-element rocket combustion chamber, consists of two chamber segments of 174 mm and 145 mm and a nozzle segment of 20 mm length. The nozzle differs from a usual configuration due to its throat with a rectangular cross section of 4.8 mm x 12 mm, which results in a contraction ratio of 2.5 and Mach number in the chamber of 0.24. The material used for the chamber segments and the nozzle segment is oxygen-free copper (Cu-HCP) [4]. Figures 2.1, 2.2 show a global view of the combustion chamber. One can note that:

- The two segments and the nozzle (made of oxygen-free copper).

- The pressure transducers and the thermocouples.
- The propellant feed system with the injector and its support.

The reference system shown in Figure 2.1 is the one used for all the results shown in this thesis.

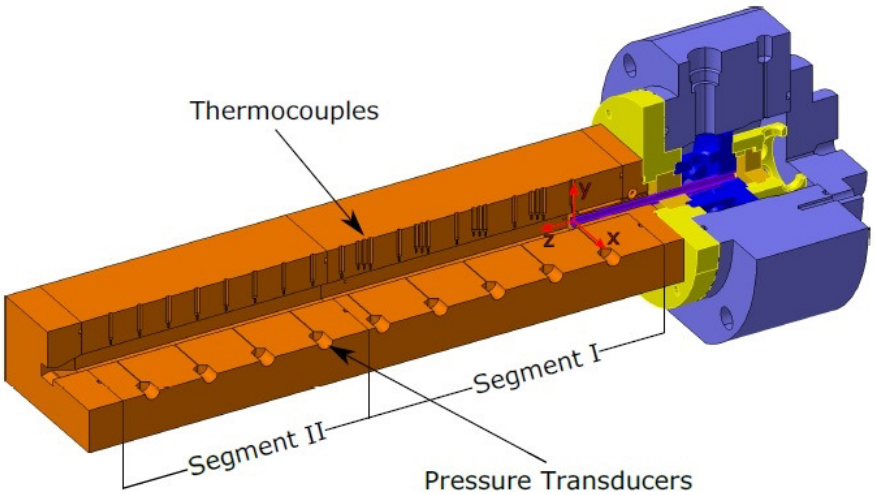


Figure 2.1: Axonometric view cad of the rectangular combustion chamber [16].

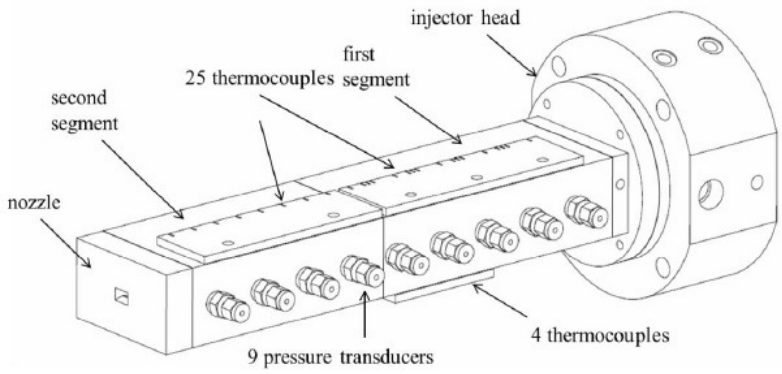


Figure 2.2: Axonometric view of the rectangular combustion chamber [4].

The chamber is equiped with equally spaced pressure transducers on the side wall for measurement of the wall pressure distribution p(x) along the chamber axis. 10 pressure transducers are used to record the axial evolution of the static chamber wall pressure. The pressure sensors are individually calibrated and operated at a data acquisition rate of 100 Hz. To determine the temperature field in the chamber, thermocouples are applied in the chamber walls in three different ways: type T thermocouples of 0.5 mm diameter located within the chamber wall with 1 mm, 2 mm and 3 mm distance to the hot wall.

Two coaxial Type T thermocouples (Medtherm), are flush mounted with the hot wall, and five Type K surface thermocouples attached to the external surface. The Type T thermocouples and the surface thermocouples are mounted with a regular path of 17 mm in the upper surface of the first and second segment, along the center plane of the combustion chamber. A scheme explaining the thermocouples pattern is present in Figure 2.3.

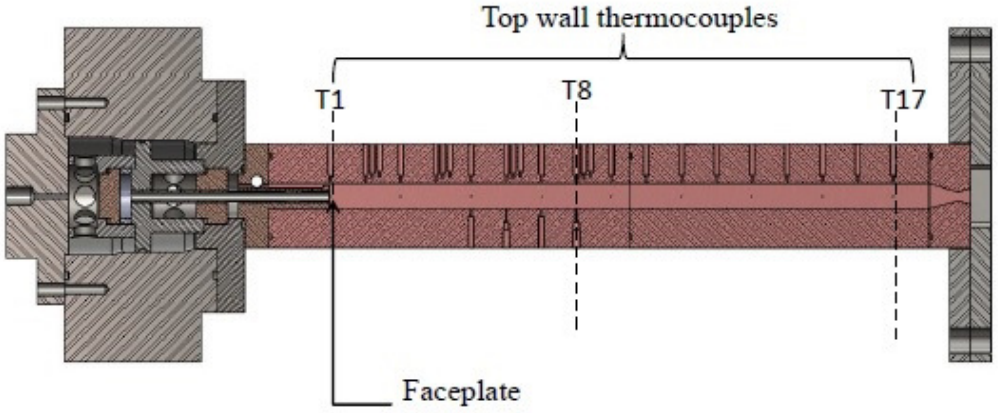


Figure 2.3: Cross section of the combustion chamber [3].

The thermocouples are positioned in precisely manufactured cylindrical holes and are kept in positions by a spring system. The spring ensures a continuous contact between the thermocouples tip and the base of the hole. This setup aims to minimize the chance of potential loss of contact during the test due to expansion or contraction of the material and vibrations [4].
The algorithm mainly works on temperatures near the hot wall, so the thermocouples type T on the top wall deserve a particular attention. Previous studies [18] show that thermocouples type at 2 mm and 3 mm do not add any information about the boundary heat flux respect to thermocouples at 1 mm. Beacause of this, **only the thermocouples type T at 1 mm from the hot gas wall are taken into account in all the studies of this thesis.**

Chapter 3

RoqFITT

In Chapter 1 inverse problem has been introduced. In particular, the attention of this thesis is directed towards the solution of an inverse transfer problem in the transient startup of a rocket combustion chamber. The temperatures in some positions of the chamber are known (measured by thermocouples during the experiments) and the aim is to estimate the boundary heat flux which comes from the hot gases in contact with the combustion chamber internal wall.
RoqFITT (Rocket $\dot{q}$ Flux Inverse Thermal Tool) is the code used in this study. It has been developed by Perakis [18], [3] at the TUM and is able to solve the Inverse heat conduction problem for rectangular and cylindrical rocket combustion chambers. The background algorithm on which the studies of this thesis ais based has been developed at TUM and it will be described in this chapter. Figure 3.1 shows how the inverse method is implemented in the aforementioned algorithm.

1. START

2. **Initial temperature** in the all domain is known, since the chamber is almost at ambient temperature before the test start.

3. The **heat flux boundary condition** is the parameter of the analysis that should be estimated. Since it is unkwown (or known approximately) one starts with a **guess**.

4. Since the initial temperature condition in the all body is given and the boundary condition heat flux is guessed, it is possible to solve a direct problem and compute the temperature after a time step.

5. Now, the **measured temperatures** at the given time step (information coming from thermocouples and not from the direct solver) and the **calculated temperatures** (information coming from direct solver and so from the boundary heat flux)

have to be **compared**. If the difference between these themperature is "small" (the concept of small is better explained by the **stopping criteria** which will be discussed in Section 3.3) the guessed boundary heat flux is the solution of the inverse problem and the algorithm switches to the next time step. If the stopping criteria is not satisfied, the boundary heat flux need to be updated in order to solve again the direct problem and reduce difference between calculated and measured temperature and finally satisfy the stopping criteria.

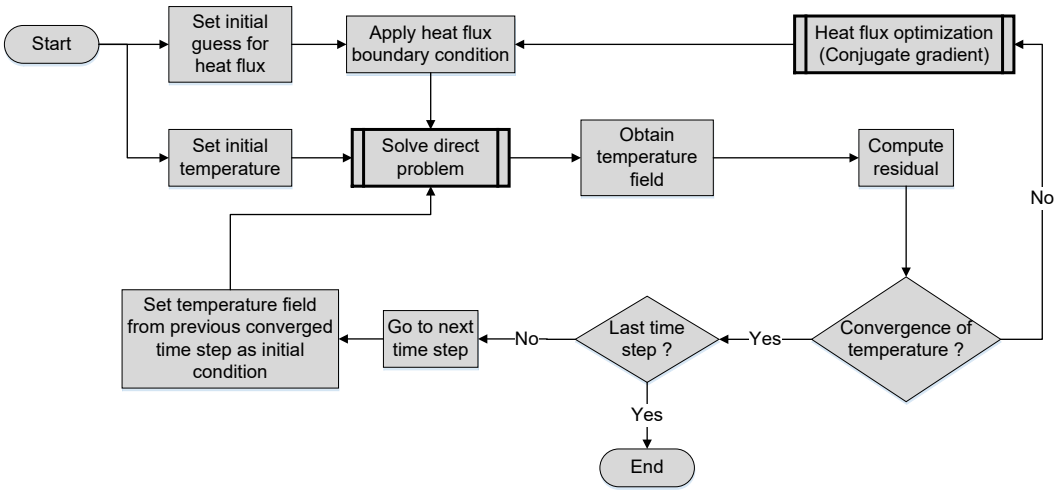


Figure 3.1: Iterative process of the inverse method [3].

Note that the inverse method is solved time step by time step and the converged computational temperatures at time t_i (obtained solving the inverse problem in time step which goes from t_{i-1} to t_i) is the starting condition for the next time step.
The flow chart in Figure 3.1 shows that the solution of the inverse problem can be basically divided in three sub-problems.

- The experimental measurements of temperatures along time.
- The solution of the direct problem (described in Section 3.2).
- The updating (or optimization) of the boundary heat fluxes in order to match computational and measured temperatures (described in Section 3.3).

The updating (or **optimization**) of the boundary condition heat fluxes and the **stopping criteria** are the core of an inverse method algorithm and constitute the so called **Optimization method**. This Chapter will describe what was the status of the inverse

method code used in this thesis at the beginning, before the changes that have been made to it as a results of studies and considerations that will be done in the following chapters. Since the code is essentially based on the algorithm of Figure 3.1 one can note that its fundamental parts are the **optimization process** and the **direct solver**. Because of this, it is useful to fully understand :

- How the combustion chamber is modeled.
- What boundary condition are used.
- How the equation are discretized.
- How the optimization process works.

Then a more detailed description of the code from an implementation point of view follows.

3.1 Modeling of the chamber and boundary conditions

Figure 3.2 shows a cross section view (z,y plane) of the combustion chamber. All the grey parts do not belong to the computational domain. The black parts consitute the computational domain. Note that only the solid material (made of copper) constitutes the domain. Furthermore, note that the nozzle has been excluded. The origin of z axis[1] is the face of the injector. The dotted part at $z<0$ is the so called heat sink. It is part of the computational domain (its length is one of the study parameters of the thesis). Internal wall of the domain for $z>0$ are in contact with the hot gas. The wall located at $z<0$ are in contact with the injector (and so there is no heat flux coming). Another fundamental difference is that the heat sink has no experimental informations about temperature.

In Figure 3.2 the surface of the computational domain are marked with 3 different colors. These colors refer to the **boundary condition** applied to the surface.

- Blue $\rightarrow$ natural convection boundary condition (it has been set for all the external side of the chamber).

[1]This refence system is the one used to show results. In the Matlab code reference system has been inverted and the origin of z axis is located at the nozzle. But all the considerations will be referred to reference system in Figure 3.2.

- Green → adiabatic boundary condition (it has been set for the contact surface between the nozzle and the domain and for the contact surface between heat sink and the left part of the chamber).
- Red → unknown heat flux boundary condition (it is basically the optimization parameter of the inverse analysis).

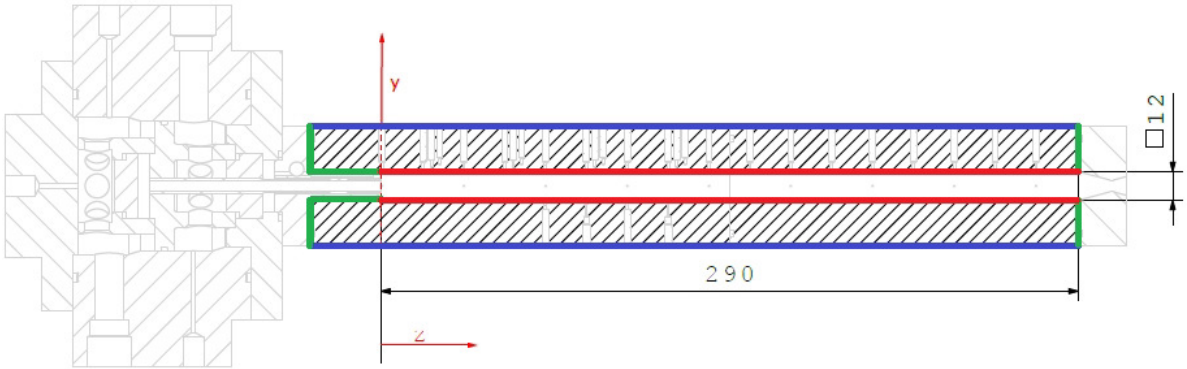


Figure 3.2: Cross-sectional view of the modeled combustion chamber (y-z plane) [16].

Figure 3.3 shows a frontal view of the modeled combustion chamber. The green dot is the thermocouple position. There are 17 thermocouples at 1 mm from the hot gas wall. Tables 3.1 show the coordinates of termocouples positions.

x [mm]	y [mm]	z [mm]
0,0425	0,032	0
0,0425	0,032	0,017
0,0425	0,032	0,034
0,0425	0,032	0,051
0,0425	0,032	0,0680
0,0425	0,032	0,0850
0,0425	0,032	0,1020
0,0425	0,032	0,1190
0,0425	0,032	0,1360
0,0425	0,032	0,1530
0,0425	0,032	0,1700
0,0425	0,032	0,1870
0,0425	0,032	0,2040
0,0425	0,032	0,2210
0,0425	0,032	0,2380
0,0425	0,032	0,2550
0,0425	0,032	0,2720

Table 3.1: Coordinates of thermocouples locations (reference system is defined in Figures 3.3 and 3.2).

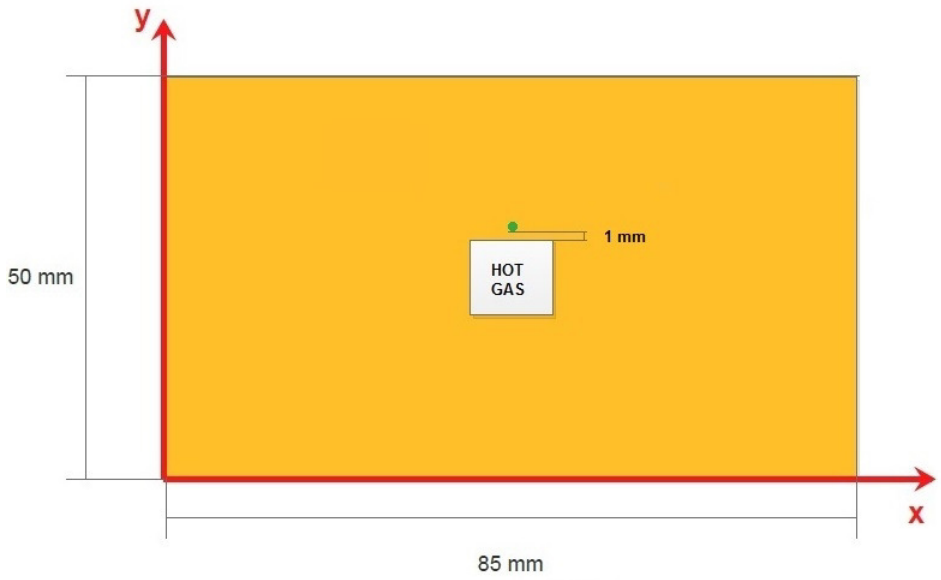


Figure 3.3: Cross-sectional view of the modeled combustion chamber (y-x plane).

3.1.1 Natural convection coefficient

Since the external surfaces of the combustion chamber are in contact with the atmosphere, thery are subject to a natural convection heat transfer.
In general convection (forced or natural) heat transfer is energy transfer by both the bulk fluid motion (advection) and the random motion of fluid molecules (conduction or diffusion) [10]. Forced convection exists when the fluid is characterized by a forced velocity. Natural convection exists when there is no forced velocity and yet convection currents exist within the fluid.
Natural convection basically originates when a body force acts on a fluid in which a density gradient exist. The net effect of a body force acting on a fluid and a density gradient is a **buoyancy force** which causes **convective motion**. Free convection flow velocities are generally much smaller than those associated with forced convection, the corresponding heat transfer rates are also smaller [10]. Because of this, for the transient start up of the considered combustion chamber (which last 3 s) the heat flux coming from the hot gas wall (caused mainly by forced convection and radiation of the flame) is much bigger than the heat flux emanated by the chamber towards the external environment. Writing the energy balance equation in integral form for a solid (no mass transfer) control volume (Eq. (3.1)) one can see that the imbalance between coming in (from hot gases) and coming out heat flux (towards the external environment due to natural convection) is responsible for the temperature rate of change of the combustion chamber.

$$\int_V \rho \frac{DT}{Dt}\, dV = \int_{A_{in}} (\dot{q}_{in})\, dA - \int_{A_{out}} (\dot{q}_{out})\, dA \tag{3.1}$$

Note that the use of Lagrangian or Eulerian derivative is indifferent since the control volume has no velocity.
Newton's law of cooling [10] established that the natural convection heat flux on a unit surface is:

$$\dot{q}_{conv} = h(T_s - T_\infty) \tag{3.2}$$

Here:

- The subscript s indicates the external surface of the chamber.
- The subcript ∞ indicates the external atmosphere.

Altough the natural convection coefficient h is different for the four external sides (the two vertical sides, top side and bottom side) of the chamber for different reasons (see [10]), previous studies have demonstrated, through sensitivity analysis, that the influence

of natural convection coefficient on the solution is not of great significance [18].
Because of this a constant natural convection coefficient of 10 $\frac{W}{m^2K}$ has been set.

3.1.2 Hot gas side condition

As already said in the previous section, the **heat flux from hot gas side is the optimization parameter** of the analysis.The heat flux will be calculated only in the parameters points[2]. Once heat flux in parameter have been calculated, they need to be interpolated in the x-y plane and along z. In particular, in xy plane two possibilities exist:

- Heat flux constant for each plane (i.e. the four hot gas walls in plane have the same heat flux).
- Parabolic heat flux for each plane. Heat flux is parabolic for all the 4 hot gas walls in the plane, with a maximum value of the parabola at the middle point of each hot gas wall and a minimun value at the corners.

Since the heat flux at boundaries are the unkown of the considered problem, no further consideration about them would be needed to solve the problem. But it is always important to have an order of degree of the unkown greatness. Because of this a small analitical analysis about hot gas condition is presented below.

3.2 Energy equation

In Section 3.1 the domain of the computational analysis has been defined. Since the interest is on the evaluation of the heat flux, the energy equation has to be solved. In a solid volume, without volumetric heating inside the , the energy equation written in differential form is [2]:

$$\rho c_p \frac{\partial T}{\partial t} = -\nabla \cdot \dot{\mathbf{q}} \tag{3.3}$$

The Fourier law says that:

$$\dot{\mathbf{q}} = -\lambda \nabla T \tag{3.4}$$

Substituting Eq. (3.4) into Eq. (3.3) one obtains parabolic heat diffusion differential equation:

$$\frac{1}{\alpha}\frac{\partial T}{\partial t} = \nabla^2 T \tag{3.5}$$

[2]The number of parameters points for the all results discussed in this thesis will be equal to number of thermocouples. Parameter points positions will be thermocouples positions projected on the hot gas wall.

In cartesian coordinates Eq. (3.5) becomes:

$$\frac{1}{\alpha}\frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial^2 x} + \frac{\partial^2 T}{\partial^2 y} + \frac{\partial^2 T}{\partial^2 z} \tag{3.6}$$

$\alpha = \frac{\lambda}{\rho c_p}$ is the thermal diffusivity. "It measures the ability of a material to conduct thermal energy relative to its ability to store thermal energy. Materials of large α will respond quickly to changes in their thermal environment, while materials of small α will respond more sluggishly, taking longer to reach a new equilibrium condition."[10]
The material properties are supposed to be constant in the whole domain and in time. Eq. (3.6) is valid in the whole domain. Plus, on the boundary surface of the control volume there is further information about heat flux. On the boundary surfaces it is valid:

$$-\lambda\frac{\partial T}{\partial \mathbf{n}}|_S = \dot{q}_{BC} \tag{3.7}$$

where $\mathbf{n}$ is the vector normal to the boundary surface and $\dot{q}_{BC}$ is the boundary heat flux that has been assigned (it can be the parameter heat flux on the hot gas wall, natural convection heat flux for external surface or adiabatic condition and so $\dot{q}_{BC} = 0$)

3.2.1 Internal nodes discretization of the equations

The approach to discretize the equation is a finite difference method in the all control volume, except for the corners (internal and external in which it has been used a finite volume approach). The discretization method used is an implicit method, forward in time, central in space. Figure 3.4 shows the situation for an internal node. ($\Delta x = \Delta y$)
Eq. 3.6 in an internal node is discretized as follows:

$$\frac{1}{\alpha}\frac{T^{k+1}_{p,m,n} - T^{k}_{p,m,n}}{\Delta t} = \frac{T^{k+1}_{m-1,n,p} - 2T^{k+1}_{m,n,p} + T^{k+1}_{m+1,n,p}}{\Delta x^2} + \frac{T^{k+1}_{m,n-1,p} - 2T^{k+1}_{m,n,p} + T^{k+1}_{m,n+1,p}}{\Delta y^2} + \frac{T^{k+1}_{m,n,p-1} - 2T^{k+1}_{m,n,p} + T^{k+1}_{m,n,p+1}}{\Delta z^2} \tag{3.8}$$

Here central-differences approximation has been used for the spatial derivatives. Index scrolls as follows:

- m along x.
- n along y.
- p along z.

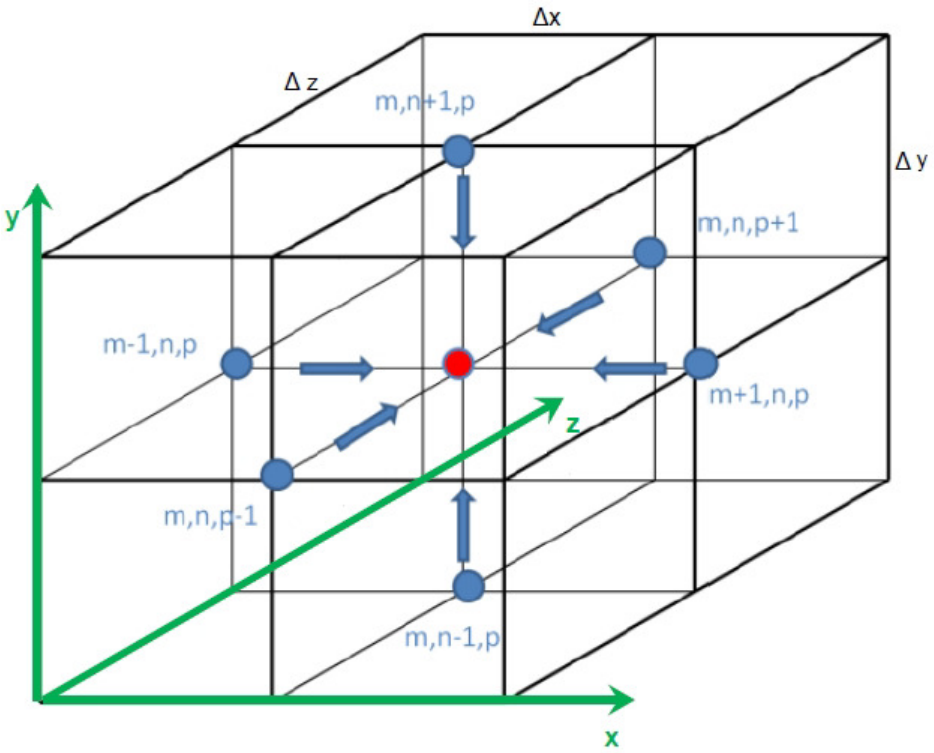


Figure 3.4: Discretization of an internal node [18].

- k along time.

Rearranging Eq. (3.8) and remembering that $\Delta y = \Delta x$ one obtains:

$$\begin{aligned} &T^{k+1}_{m,n,p}(1 + 2Fo_z + 4Fo_x) - Fo_x T^{k+1}_{m-1,n,p} - Fo_x T^{k+1}_{m,n+1,p} - Fo_x T^{k+1}_{m+1,n,p} \\ &- Fo_x T^{k+1}_{m,n-1,p} - Fo_z T^{k+1}_{m,n,p+1} - Fo_z T^{k+1}_{m,n,p-1} = T^{k}_{m,n,p} \end{aligned} \tag{3.9}$$

Where $Fo_x = \frac{\alpha \Delta t}{\Delta x^2}$ and $Fo_z = \frac{\alpha \Delta t}{\Delta z^2}$ are finite difference forms of the Fourier number [10]

3.2.2 Boundary nodes discretization

In Section 3.1 boundary conditions for the computational domain have been shown. All the boundary conditions are given in therm of heat fluxes and so Neumann boundary condition has been established, recalling that on the boundaries Eq. (3.7) is valid. This means that, knowing information about the heat fluxes at the boundaries, the first derivative of temperature at the boundaries is known.
According to this, the discretization of the energy equation has to be slightly modified. As an example, the heat flux equation at a generic node on the hot gas side will be discretized (not in a corner), as shown in Figure 3.5. Here, the parabolic heat diffusion differential

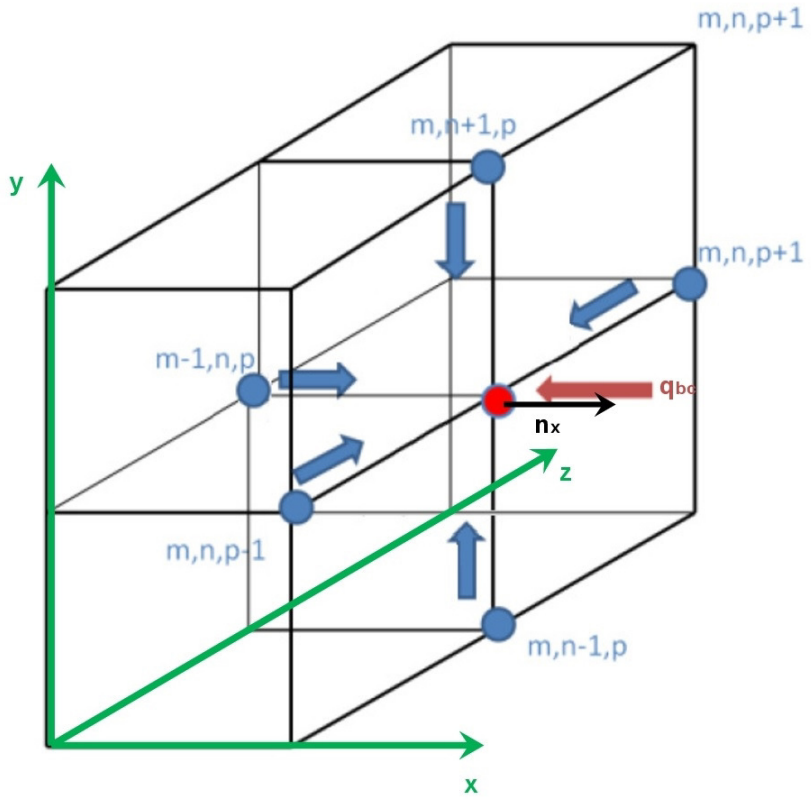


Figure 3.5: Discretization for a generic node on the hot gas side boundary [18].

equation (3.5) has to be discretized which is valid in the whole domain and so also on the boundaries. In this particular example the considered node is on the left hot gas wall side of the combustion chamber. So information on the first derivative of temperature along x is given. In fact, knowing that the normal to the surface in this case is parallel to x direction, Eq. (3.7) becomes[3]:

$$\lambda \frac{\partial T}{\partial \mathbf{x}}|_{m,n,p} = \dot{q}_{BCm,n,p} \Rightarrow \frac{\partial T}{\partial \mathbf{x}}|_{m,n,p} = \frac{\dot{q}_{BCm,n,p}}{\lambda} \tag{3.10}$$

Here the subscripts m, n, p indicate that the considered quantity is evaluated in the position of the node m, n, p.

To descritize the second derivative of temperature along y and z, it is still possible to use the Central Difference Scheme. The second derivative along x needs a special treatment beacause of the presence of the boundary condition. Since the situation along y and z

[3]Note the the sign minus has disappeared since a heat flux with direction adverse respect to x is considered as positive here.

doesn't differ from the case of an internal node let's focus on the nodes in x direction (Figure 3.6).

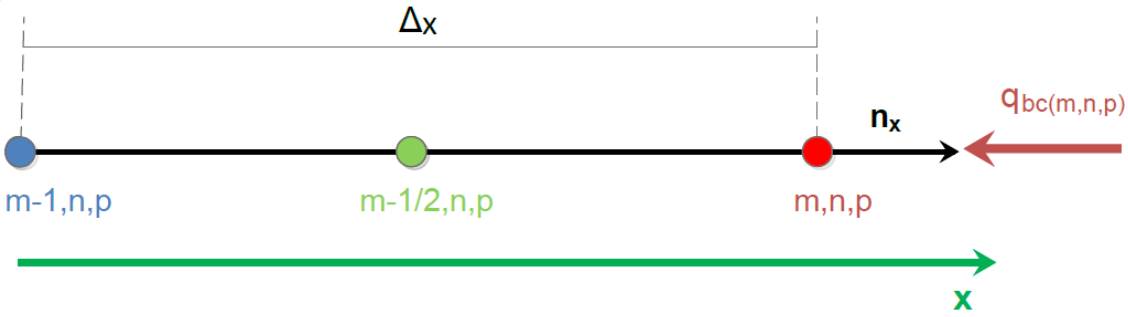


Figure 3.6: Discretization along x direction on the hot gas side boundary.

The final aim is to estimate the second derivative of temperature respect to x in the hot gas side node (m,n,p) $\frac{\partial^2 T}{\partial^2 x}|_{m,n,p}$.
In order to achieve this aim it is useful to define a fictitious node $(m-1/2,n,p)$. One can estimate the first derivative in this node using a centred in space scheme which is second order accurate.

$$\frac{\partial T}{\partial x}|_{m-1/2,n,p} \approx \frac{T_{m,n,p} - T_{m-1,n,p}}{\Delta x} \tag{3.11}$$

Now, a backward in space scheme can be used in order to estimate the second derivative in m,n,p.

$$\frac{\partial^2 T}{\partial^2 x}|_{m,n,p} \approx \frac{\frac{\partial T}{\partial x}|_{m,n,p} - \frac{\partial T}{\partial x}|_{m-1/2,n,p}}{\Delta x/2} \tag{3.12}$$

Substituting Eq. (3.10) and (3.11) into Eq. (3.12) one obtains:

$$\frac{\partial^2 T}{\partial^2 x}|_{m,n,p} \approx \frac{\frac{\dot{q}_{BC}}{\lambda} - \frac{T_{m,n,p} - T_{m-1,n,p}}{\Delta x}}{\Delta x/2} \tag{3.13}$$

This is the approximation of the second derivative on a boundary nodes and now it can be used in order to discretize **the parabolic heat diffusion differential equation on the boundaries**. Rearranging one obtains:

$$\begin{aligned} &T^{k+1}_{m,n,p}(1 + 2Fo_z + 4Fo_x) - 2Fo_x T^{k+1}_{m-1,n,p} - Fo_x T^{k+1}_{m,n+1,p} - Fo_x T^{k+1}_{m,n-1,p} - \\ &Fo_z T^{k+1}_{m,n,p+1} - Fo_z T^{k+1}_{m,n,p-1} = T^{k}_{m,n,p} + \frac{2\dot{q}_{BC(m,n,p)}\Delta t}{\Delta x \rho c} \end{aligned} \tag{3.14}$$

Note that the vector of known terms at boundaries has been modified with the additional term $\frac{2\dot{q}_{BC(m,n,p)}\Delta t}{\Delta x \rho c}$. **Equation** (3.15) **is valid as is, only for left hot gas side boundaries**.
For top, bottom and right hot gas side of the chamber the same procedure has been used

to discretize the equation.

At **boundary surfaces** for which an **adiabatic boundary condition** has been set (Figure 3.1), the additional term disappears since $\dot{q}_{BC} = 0$.
On the external surfaces of the chamber, where free convection boundary condition has been set, Eq. (3.2) is valid. Suppose to be at the external rigth side[4] of the chamber. One have to replace $-h(T_{m,n,p} - T_\infty)$ in place of $\dot{q}_{BC}$. Rearranging one obtains:

$$\begin{aligned} &T^{k+1}_{m,n,p}(1 + 2Fo_xBi_x + 2Fo_z + 4Fo_x) - 2Fo_xT^{k+1}_{m-1,n,p} - Fo_xT^{k+1}_{m,n+1,p} - Fo_xT^{k+1}_{m,n-1,p} - \\ &Fo_zT^{k+1}_{m,n,p+1} - Fo_zT^{k+1}_{m,n,p-1} = T^k_{m,n,p} + 2Fo_xBi_xT_\infty \end{aligned} \tag{3.15}$$

where $Bi_x = \frac{h}{\Delta x \lambda}$ is the Biot number. Note that $Bi_x = Bi_y$ since $\Delta x = \Delta y$.

Corner nodes need a special treatment since in corners normal to the surface is not defined. Because of this it is not possible to exploit the fact that the first derivative along the normal direction is equal to the boundary heat flux.
Here a finite volume approach has been used in order to discretize the integral form of the energy conservation. The energy equation in integral form for a solid control volume is:

$$\int_V \rho c_p \frac{dT}{dt}\, dV = \int_S q\, dS \tag{3.16}$$

For the control volume defined in Figure 3.7 it can be discretized as follows [18]:

$$\rho c_p \frac{T^{k+1}_{m,n,p} - T^k_{m,n,p}}{\Delta t}\Delta x^2 \Delta z = q_{convx}\frac{\Delta x \Delta z}{4} + q_{convy}\frac{\Delta x \Delta z}{4} + q_{condx}\frac{\Delta x \Delta z}{4} + q_{condy}\frac{\Delta x \Delta z}{4} + q_{condz}\frac{\Delta x^2}{4} \tag{3.17}$$

Substituting the respective values fot conductive and convective heat fluxes and using Fourier law (Eq. (3.4)) and Newton cooling law (Eq. (3.2)), discretizing and rearranging one obtains:

$$\begin{aligned} &T^{k+1}_{m,n,p}(1 + 4Fo_xBi_x + 2Fo_z + 4Fo_x) - 2Fo_xT^{k+1}_{m+1,n,p} - 2Fo_xT^{k+1}_{m,n-1,p} \\ &- 2Fo_zT^{k+1}_{m,n,p+1} = T^k_{m,n,p} + 4Fo_xBi_xT_\infty \end{aligned} \tag{3.18}$$

Since the pattern of the derivation of all these equations has been shown, a list ofthe results for the other cases it is presented in the following section.

[4]The rigth side has been chosen because in this way the discretization of the equation is exactly the same of left hot gas wall side case. In fact the scheme of Figure 3.5 is still valid and the only difference is that heat flux comes from convection and not from hot combusted gases

3.2.3 Discretized equations summary

Node:Outside Corner, First plane

$$T^{k+1}_{m,n,p}(1 + 4Fo_xBi_x + 2Fo_z + 4Fo_x) - 2Fo_xT^{k+1}_{m+1,n,p} - 2Fo_xT^{k+1}_{m,n-1,p} - 2Fo_zT^{k+1}_{m,n,p+1} = T^{k}_{m,n,p} + 4Fo_xBi_xT_\infty \tag{3.19}$$

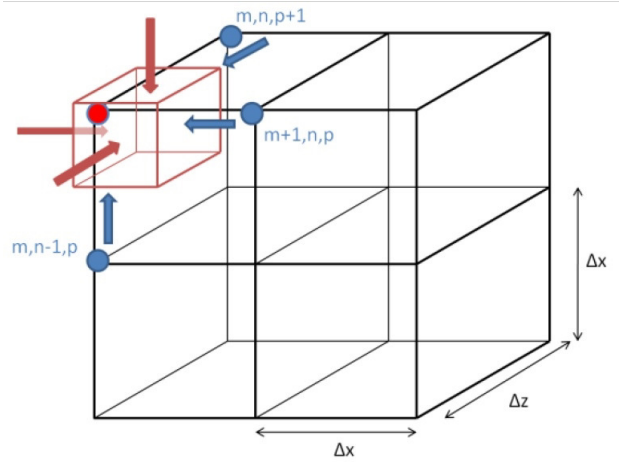


Figure 3.7: Outer corner node in the first plane ($z = 0$ m) [18].

Node:Outside Corner, Middle plane

$$T^{k+1}_{m,n,p}(1 + 4Fo_xBi_x + 2Fo_z + 4Fo_x) - 2Fo_xT^{k+1}_{m-1,n,p} - 2Fo_xT^{k+1}_{m,n-1,p} - Fo_zT^{k+1}_{m,n,p+1} - Fo_zT^{k+1}_{m,n,p-1} = T^{k}_{m,n,p} + 4Fo_xBi_xT_\infty \tag{3.20}$$

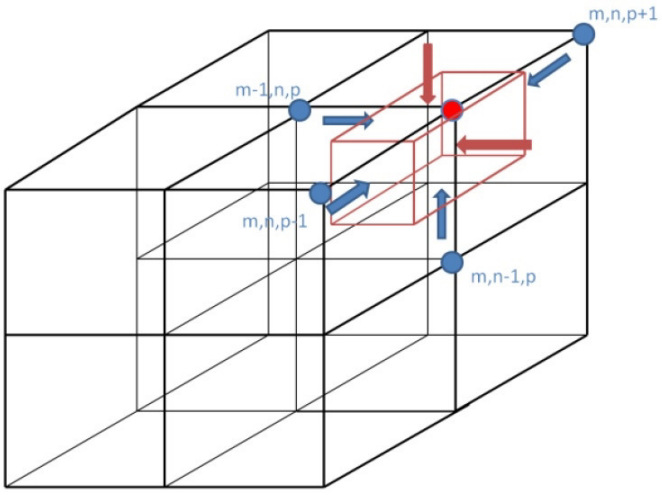


Figure 3.8: Outside corner in a generic middle plane [18].

Node:Outside/Inside middle plane

Inside:

$$T^{k+1}_{m,n,p}(1+2Fo_z+4Fo_x)-2Fo_xT^{k+1}_{m-1,n,p}-Fo_xT^{k+1}_{m,n+1,p}-Fo_xT^{k+1}_{m,n-1,p}- \\ Fo_zT^{k+1}_{m,n,p+1}-Fo_zT^{k+1}_{m,n,p-1}=T^{k}_{m,n,p}+\frac{2q_{BC(m,n,p)}\Delta t}{\Delta x\rho c_p} \tag{3.21}$$

Outside:

$$T^{k+1}_{m,n,p}(1+2Fo_xBi_x+2Fo_z+4Fo_x)-2Fo_xT^{k+1}_{m-1,n,p}-Fo_xT^{k+1}_{m,n+1,p}-Fo_xT^{k+1}_{m,n-1,p}- \\ Fo_zT^{k+1}_{m,n,p+1}-Fo_zT^{k+1}_{m,n,p-1}=T^{k}_{m,n,p}+2Fo_xBi_xT_\infty \tag{3.22}$$

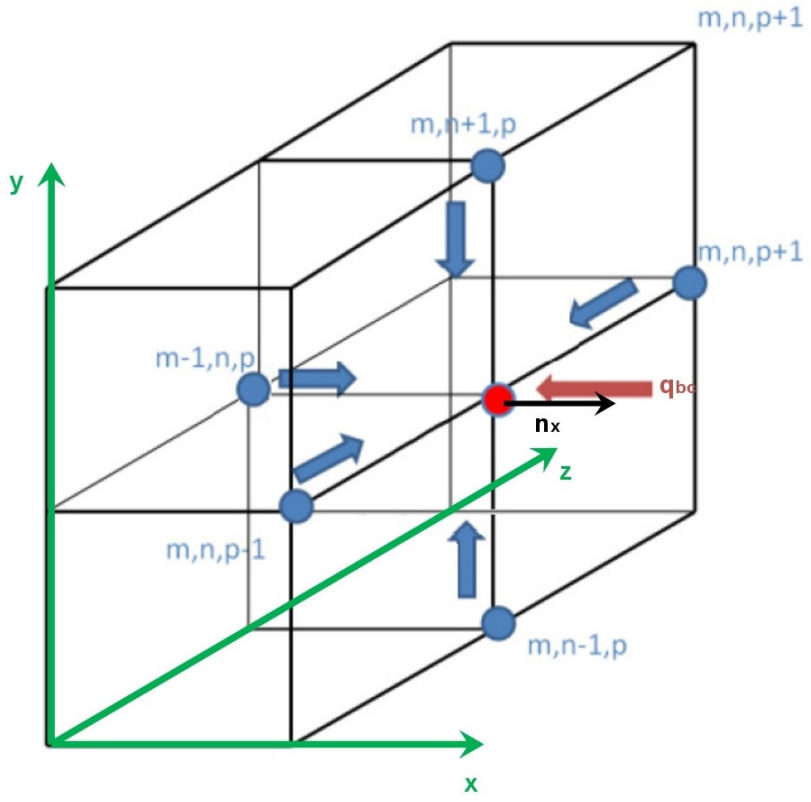


Figure 3.9: Outside/Inside node in a generic Middle plane [18].

Node:Inside Corner, First plane

$$T^{k+1}_{m,n,p}(1+2Fo_z+4Fo_x)-4/3Fo_xT^{k+1}_{m-1,n,p}-4/3Fo_xT^{k+1}_{m,n+1,p}-2/3Fo_xT^{k+1}_{m,n-1,p}-2/3Fo_xT^{k+1}_{m+1,n,p}$$
$$2Fo_zT^{k+1}_{m,n,p+1}=T^{k}_{m,n,p}+\frac{4/3q_{BC(m,n,p)}\Delta t}{\Delta x\rho c} \tag{3.23}$$

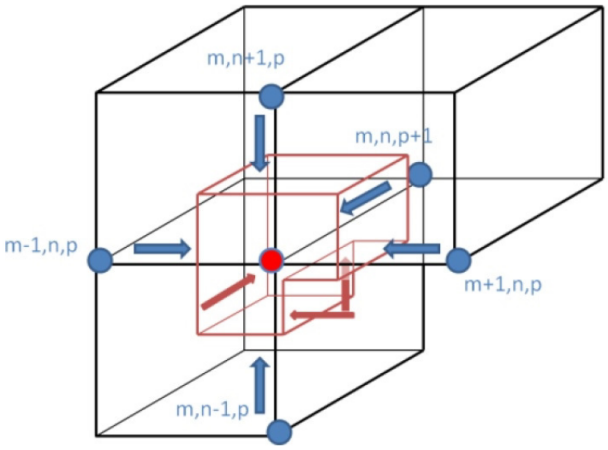


Figure 3.10: Inside corner node in the first plane ($z = 0$ m) [18].

Node:Inside Corner, Middle plane

$$T^{k+1}_{m,n,p}(1+2Fo_z+4Fo_x)-4/3Fo_xT^{k+1}_{m-1,n,p}-4/3Fo_xT^{k+1}_{m,n+1,p}-2/3Fo_xT^{k+1}_{m,n-1,p}-2/3Fo_xT^{k+1}_{m+1,n,p}$$
$$-Fo_zT^{k+1}_{m,n,p+1}-Fo_zT^{k+1}_{m,n,p-1}=T^{k}_{m,n,p}+\frac{4/3q_{BC(m,n,p)}\Delta t}{\Delta x\rho c} \tag{3.24}$$

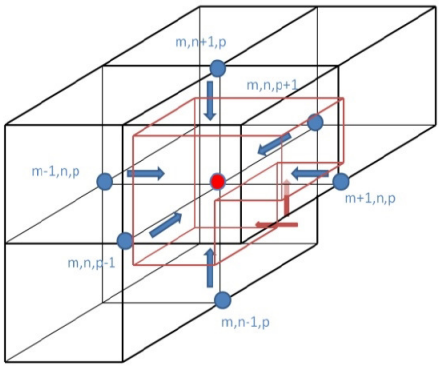


Figure 3.11: Inside corner node a generic plane [18].

Internal node

$$T^{k+1}_{m,n,p}(1 + 2Fo_z + 4Fo_x) - Fo_x T^{k+1}_{m-1,n,p} - Fo_x T^{k+1}_{m,n+1,p} - Fo_x T^{k+1}_{m+1,n,p} - Fo_x T^{k+1}_{m,n-1,p} - Fo_z T^{k+1}_{m,n,p+1} - Fo_z T^{k+1}_{m,n,p-1} = T^{k}_{m,n,p} \tag{3.25}$$

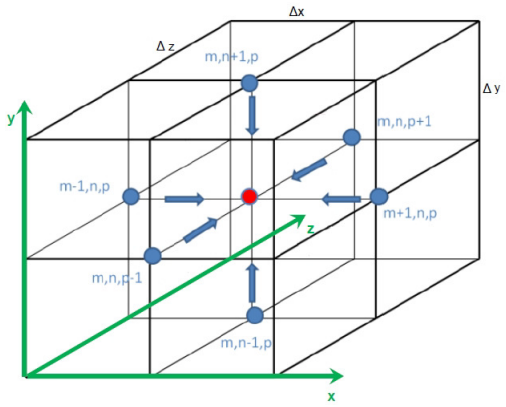


Figure 3.12: Generic internal node [18].

It is important to note that all the coefficient at left hand side of the discretized equation don't change over iteration and time step. So it is possible to calculate them out of the cycle of optimization. On the other side, the vector of the boundary condition in general changes because of the change of the heat fluxes. Because of this, the right hand side term has to be updated every time that the linear system is solved. (That is every time that the boundary hot gas wall heat fluxes are updated)

3.2.4 Building and solution of the linear system

It is easy to understand the form of the linear system being created after the differential equation has been modified numerically. It looks like this:

$$\underline{\mathbf{A}}\mathbf{T}^{k+1} = \mathbf{T}^{k} + \mathbf{b} \tag{3.26}$$

where:

- $\mathbf{T}^{k+1}$ is the vector of nodal temperatures a time t^{k+1}. It is the unkown of the linear system.
- $\mathbf{T}^{k}$ is the vector of nodal temperatures at time t^{k}. It is the known term of the system since it comes from the solution at previous time step.

- $\underline{\mathbf{A}}$ is the coefficients matrix and it depends on the discretized equations but also on the boundary conditions. Different boundary conditions lead to a different matrix of coefficients.
- $\mathbf{b}$ is a vector which depends on the boundary conditions.

Let's take as an example a line of the linear system referring to an outside node (Eq. 3.22). It is possible to note that the considered line of the coeffients matrix $\underline{\mathbf{A}}$ is filled by two different contributions:

- Geometry coefficients (which are independent from boundary conditions).
- Boundary conditions coefficients (which are dependent on the boundary conditions).

In the considered case geometry coefficients are:

- $(1 + 2Fo_z + 4Fo_x)$ which multiply $T^{k+1}_{m,n,p}$.
- $-2Fo_x$ which multiplies $T^{k+1}_{m-1,n,p}$.
- $-Fo_x$ which multiplies $T^{k+1}_{m,n+1,p}$.
- $-Fo_x$ which multiplies $T^{k+1}_{m,n-1,p}$.
- $-Fo_z$ which multiplies $T^{k+1}_{m,n,p+1}$.
- $-Fo_z$ which multiplies $T^{k+1}_{m,n,p-1}$.

Boundary condition coefficient is:

- $2Fo_xBi_x$ which multiplies $T^{k+1}_{m,n,p}$.

Note that for the considered case (but it is valid in general), boundary conditions coefficients are only present in the main diagonal.
The relative term of the vector $\mathbf{b}$ is here:

- $2Fo_xBi_xT_\infty$.

Note, furthermore, that a boundary condition coefficient only exists when the boundary condition for the considered node is a convective heat flux (outside nodes). In cases in which the boundary condition is the unkown heat flux $\dot{q}_{BC}$, no boundary condition coefficient affects the coefficients matrix and the term of vector $\mathbf{b}$ is $\frac{2\dot{q}_{BC(m,n,p)}\Delta t}{\Delta x \rho c_p}$.
In the following lines the method used to solve the linear system is described [18].
The coefficients matrix has the dimensions NxN where N represents the number of nodes

in the system. Is is mainly possessed by zeroes and has non-zero elements only along 7 of its diagonals (main diagonal and diagonals relatives to other six nodes which affect the main diagonal node). Since the matrix is sparse, the memory requirements are reduced radically, when the definition of its elements is done the right way. Because of this, the Matlab built in function "spdiags" was used for the definition of the coefficients matrix [18].

Many methods exist to solve lineare systems (bouth direct and iterative). The computational implementation of direct methods has a high computational cost for sparse matrices and for that reason an iterative approach was chosen for the solution of the temperature profile. In particular the selected iterative method to solve the linear system is the **conjugate gradient method** (**CGM**). A simplified description of the CGM is given below. Imagine to want to solve the following linear system: $\underline{\mathbf{A}}\mathbf{x} = \mathbf{b}$.

CGM works as follow:

1. Given the first approximation $\mathbf{x_0}$ calculate the residual $\mathbf{r_0} = \mathbf{b} - \underline{\mathbf{A}}\mathbf{x_0}$; $\mathbf{p_0} = \mathbf{r_0}$
2. For j=0,1,..., until convergence Do:
3. $\alpha_j = \frac{\mathbf{r_j} \cdot \mathbf{r_j}}{\underline{\mathbf{A}}\mathbf{p_j} \cdot \mathbf{p_j}}$
4. $\mathbf{x_{j+1}} = \mathbf{x_j} + \alpha_j \mathbf{p_j}$
5. $\mathbf{r_{j+1}} = \mathbf{r_j} - \alpha_j \underline{\mathbf{A}}\mathbf{p_j}$
6. $\beta_j = \frac{\mathbf{r_{j+1}} \cdot \mathbf{r_{j+1}}}{\mathbf{r_j} \cdot \mathbf{r_j}}$
7. $\mathbf{p_{j+1}} = \mathbf{r_{j+1}} + \beta_j \mathbf{p_j}$
8. EndDo

Matlab already has an implementation of this algorithm in its libraries and also offers different modifications of the CGM. In particular, previous studies [18] have shown that Conjugate Gradient Squared Method is the best method for the considered problem both for results and computational power.

The efficiency and robustness of iterative methods (and so also of CGM) can be improved using **preconditioning**. Preconditioning is the transformation of the original linear system into an other system which has the same solution but it can be more easily solved using an iterative method.

The first step in preconditioning is to find a preconditioning matrix $\underline{\mathbf{M}}$. This matrix can be defined in many different ways but is has to satisfy some fundamental requirements:

- It should be easy to solve linear system in the form $\underline{\mathbf{M}}\mathbf{x} = \mathbf{b}$, since the preconditioneD algorithm of CGM requires the solution of the aforementioned linear system at each iteration.
- $\underline{\mathbf{M}}$ and $\underline{\mathbf{A}}$ should be close in some sense.
- Clearly $\underline{\mathbf{M}}$ should be nonsingular.

The chosen preconditioner was a preconditioner being present in factored form:
$\underline{\mathbf{M}}_{\mathbf{L}} = \underline{\mathbf{M}}_{\mathbf{R}}$
In this case the preconditioner is being split:

$$\underline{\mathbf{M}}_{\mathbf{L}}^{-1}\underline{\mathbf{A}}\ \underline{\mathbf{M}}_{\mathbf{R}}^{-1}\mathbf{u} = \underline{\mathbf{M}}_{\mathbf{L}}^{-1}\mathbf{u};\ \textit{where}\ \mathbf{x} = \underline{\mathbf{M}}_{\mathbf{R}}^{-1}\mathbf{b} \tag{3.27}$$

The factorization of the preconditioner was obtained in the code in the form of an incomplete Cholesky factorization [13]. This implies:
$\underline{\mathbf{M}} = \underline{\mathbf{L}}\ \underline{\mathbf{L}}^{T}$
Cholesky preconditioning is performed using the built in Matlab function named "ichol". The modified conjugate gradient method is presented below:

1. Given the first approximation $\mathbf{x_0}$ calculate the residual $\mathbf{r_0} = \mathbf{b} - \underline{\mathbf{A}}\mathbf{x_0}$;
   $\mathbf{q_0} = \underline{\mathbf{L}}^{-1}\mathbf{r_0}$; $\mathbf{p_0} = \underline{\mathbf{L}}^{-T}\mathbf{q_0}$;
2. For j=0,1,..., until convergence Do:
3. $\alpha_j = \frac{\mathbf{q_j}\cdot\mathbf{q_j}}{\underline{\mathbf{A}}\mathbf{p_j}\cdot\mathbf{p_j}}$
4. $\mathbf{x_{j+1}} = \mathbf{x_j} + \alpha_j\mathbf{p_j}$
5. $\mathbf{q_{j+1}} = \mathbf{q_j} - \alpha_j\underline{\mathbf{L}}^{-1}\underline{\mathbf{A}}\mathbf{p_j}$
6. $\beta_j = \frac{\mathbf{q_{j+1}}\cdot\mathbf{q_{j+1}}}{\mathbf{q_j}\cdot\mathbf{q_j}}$
7. $\mathbf{p_{j+1}} = \underline{\mathbf{L}}^{-T}\mathbf{q_{j+1}} + \beta_j\mathbf{p_j}$
8. EndDo

3.3 Optimization method

Many techniques exist to solve inverse problems. The one used in the code developed at TUM, and on which this thesis is based, is the ”Conjugate gradient method for parameter

estimation". This section will describe how this method works [15]. The Conjugate Gradient Method, is a powerful iterative technique used to solve linear and nonlinear inverse problems of parameter estimation. Let's consider the combustion chamber in which the transient temperatures at m locations x_{meas} are measured, m = 1,2,. . . ,M (as mentioned in Chapter 2, in this thesis the measured temperatures will come from the 17 thermocouples of type T placed at 1 mm from the hot gas wall surface) at time t_i, i=1,2, ... , I. One defines:

- $\mathbf{T_m}$ as the vector of measured temperatures at a given time t_i. It has dimension Mx1.
- $\mathbf{T_c}$ as the vector of calculated temperatures (from direct problem) in the same position in which temperatures have been measured. It has dimension Mx1.
- $\mathbf{P}$ as the vector of parameters given as input to the direct solver to calculate the temperatures. Note that these parameters could be in general the boundary heat fluxes, the convective coefficient h, etc. In the considered case they are the boundary heat fluxes. It is very important to understand that the calculated temperatures are function of that parameters so that $\mathbf{T_c} = \mathbf{T_c}(\mathbf{P})$. This vector has dimension Nx1.

Note that it has to be:

$$N \leq M \tag{3.28}$$

Such parameter estimation problem is solved by the minimization of the ordinary least squares norm:

$$S(\mathbf{P}) = [\mathbf{T_m} - \mathbf{T_c}(\mathbf{P})]^T [\mathbf{T_m} - \mathbf{T_c}(\mathbf{P})] \tag{3.29}$$

This minimization problem has to be solved for each time t_i, that is for each time step. The iterative procedure of the Conjugate Gradient Method for the minimization of the above square norm S($\mathbf{P}$) is:

$$\mathbf{P}^{k+1} = \mathbf{P}^k - \beta^k d^k \tag{3.30}$$

where β^{k} is the *search step size*, d^k is the *direction of descent* and the supercript k indicates the iteration. The direction of descent is a conjugation of the gradient direction, ∇S($\mathbf{P}^{\mathrm{k}}$) and the direction of descent of the previous iteration $\mathrm{d}^{\mathrm{k\text{-}1}}$. It is:

$$d^k = \nabla S(\mathbf{P}^k) + \gamma^k d^{k-1} \tag{3.31}$$

Different expressions exist for the conjugation coefficient. One of these is the *Fletcher-Reeves*:

$$\gamma^k = \frac{\sum_{j=1}^{N} [\nabla S(\mathbf{P}^k)]_j^2}{\sum_{j=1}^{N} [\nabla S(\mathbf{P}^{k-1})]_j^2} \quad \text{for } k = 1,2,\ldots \tag{3.32}$$

$$\gamma^k = 0 \quad \text{for } k = 0 \tag{3.33}$$

Here, $[\nabla \mathrm{S}(\mathbf{P})^{\mathrm{k}}]_{\mathrm{j}}$ is the j^{th} component of the gradient direction evaluated at iteration k. The expression for the gradient direction is obtained by differentiating equation (3.29) with respect to the unknown parameters $\mathbf{P}$.

$$\nabla S(\mathbf{P}^k) = -2(\underline{\mathbf{J}}_S^k)^T[\mathbf{T_m} - \mathbf{T_c}(\mathbf{P}^k)] \tag{3.34}$$

where $\underline{\mathbf{J}}_S$ is the **sensitivity matrix** defined in Section 3.3.1. The search step size is:

$$\beta^k = \frac{[\underline{\mathbf{J}}_S^k d^k]^T[\mathbf{T_c}(\mathbf{P}^k) - \mathbf{T_m}]}{[\underline{\mathbf{J}}_S^k d^k]^T[\underline{\mathbf{J}}_S^k d^k]} \tag{3.35}$$

3.3.1 The sensitivity matrix

The sensitivity matrix is defined as follow:

$$\underline{\mathbf{J}}_S(\mathbf{P}) = \begin{bmatrix} \frac{\partial T_1}{\partial P_1} & \frac{\partial T_1}{\partial P_2} & \frac{\partial T_1}{\partial P_3} & \cdots & \frac{\partial T_1}{\partial P_N} \\ \frac{\partial T_2}{\partial P_1} & \frac{\partial T_2}{\partial P_2} & \frac{\partial T_2}{\partial P_3} & \cdots & \frac{\partial T_2}{\partial P_N} \\ \frac{\partial T_3}{\partial P_1} & \frac{\partial T_3}{\partial P_2} & \frac{\partial T_3}{\partial P_3} & \cdots & \frac{\partial T_3}{\partial P_N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial T_M}{\partial P_1} & \frac{\partial T_M}{\partial P_2} & \frac{\partial T_M}{\partial P_3} & \cdots & \frac{\partial T_M}{\partial P_N} \end{bmatrix}$$

The elements of the sensitivity matrix are the sensitivity coefficients.
The sensitivity coefficient $J_{S\,ij}$, is defined as the first derivative of the estimated temperature T_i with respect to the unknown parameter P_j. Generally, the sensitivity matrix $\underline{\mathbf{J}}_S$ is not invariant but varies. In RoqFITT, the sensitivity matrix is considered to be constant and so it can be calculated out of the loop of iterations.
The sensitivity matrix plays an important role in parameter estimation problems. The sensitivity coefficient $J_{S\,ij}$, is a measure of the sensitivity of the estimated temperature T_i with respect to changes in the paramter P_j. It gives a measure of how much the temperature T_i changes respect to a variaton of the P_j. If is has a small value, one can easily note that the estimation of the parameter is extremely difficult in such a case, because small changes in temperatures would lead to big variations of $\mathbf{P}$. In fact, when the sensitivity coefficients are small an inverse problem is ill-conditioned. The literature offers many different possibilities to determine the values of $\underline{\mathbf{J}}_S$.
In the RoqFITT, a finite difference approach has been chosen. More specifically, a forward finite difference approach [2] of first order has been chosen. The implemented solver is described by the following equation for the computation of the sensitivity of all temperature

sensors to the variation of the heat flux at a single parameter:

$$J_{S\,ij} = \frac{T_i(P_1,P_2,P_3,\ldots,P_j+\epsilon P_j+\ldots+P_N) - T_i(P_1,P_2,P_3,\ldots,P_j+\ldots+P_N)}{\epsilon P_j} \tag{3.36}$$

It is has to be noted that the determination of the sensitivity matrix in such a way requires the solution of the direct problem $N+1$ times and could be very time-consuming. This is the reason why it has been chosed to calculate it out of the loop only one time and to consider it constant (since it is only used for the convergence of the optimization method the only important thing is that the convergence is reached and not to have an accurate value of the sensitivity matrix). More precisely, the sensitivity matrix used in the computation of the Optimization method is computed in the following way:

1. Set a starting condition of temperatures in the all copper block.
2. Set the heat flux value for N given parameters. It should be a reasonable value (it has be chosen 5 MW).
3. Calculate the solution of the direct problem at an "appropriate time" (e.g. 1 s) using a small time step. An appropriate time t is reached when the sensitivity coefficients are stationary.
4. Calculate the solution of the direct problem at the same appropriate time for each perturbed parameter (during the whole time from 0 to 1 s the parameter remain perturbed). In this works $\epsilon = 10^{-5}$ wash chosen.
5. Knowing the solution of temperature in the thermocouples position at a certain time (with perturbed and unperturbed parameters), it is possible to compute the first derivate at time t as defined in equation (4.16)

3.3.2 The stopping criteria

Even if, in the conjugate gradient method the function S is minimized, as a stopping criteria it has been chosen to compare a residual function R(defined in the equation below) with an appropriate value defined in the following line. In the case of the considered system, the fact that there are M measurements due to the M thermocouples positioned in the chamber, combined with the fact that the analyzed phenomenon is time-dependent produces following expression for the residual function [18]:

$$R(\mathbf{P}) = \int_{t_{i-1}}^{t_i} \sum_{m=1}^{M} ||\mathbf{T_c}(\mathbf{P}) - \mathbf{T_m}||^2 \, dt \tag{3.37}$$

The stopping criteria is thus defined:

$$R(S) < \delta \tag{3.38}$$

The value of ϵ should be [18]:

$$\delta = 2M\sigma^2\Delta t \tag{3.39}$$

Despite these definitions, the stopping criteria has been decreased since it was one of the parameters study of this thesis and also the standard deviation σ which was origially set equal to 0.3 K in the original code has been modified due some statistical considerations explained in the following chapters.

3.3.3 The implementation of the algorithm

Now that the optimization method has been described it is useful to see how it can be implemented. Imagine to be at the iteration k of the time step i.

1. Obtain the temperatures $\mathbf{T_c}(\mathbf{P})$ by solving the direct heat transfer problem based on the estimated P^k.
2. Calculate the temperature residual function **R(S)** from Eq. (3.37) and check if the solution has converged (that is Eq. (3.38) is satisfied). If the solution is converged (the parameters heat flux are solution of the inverse problems) switch to the next time step and start from iteration 1 and step 1. If the solution is not converged go to step 3
3. Compute $\nabla \mathrm{S}(\mathbf{P}^{\mathrm{k}})$ from Eq. (3.37)
4. Compute $\mathbf{d^k}$ from Eq. (3.31)
5. Compute $\gamma^{\mathbf{k}}$ from Eq. (3.32)
6. Compute $\beta^{\mathbf{k}}$ from Eq. (3.35)
7. Compute $\mathbf{P^k}$ from Eq. (3.30) and go to step 1 with this new set of parameters.

Note that it is necessary to interpolate the heat fluxes (in the plane and along z) in order to be able to solve the direct problem. Infact the number of parameter is finite, but the heat flux boundary condition has to be evaluated along the all hot gas wall.

3.4 Code implementation

Now that is more clear how the code works in its fundamental parts (optimization method and direct solution) it is easier to have an in-depth view into it. The code can be summarized as follows:

- Loading experimental data.
- User input.
- Domain discretization.
- Thermocouples measurements managing.
- Time transformation.
- Parameter points definition.
- Heat flux first estimation.
- Building matrix and boundary condition coefficients.
- Inverse cycle.
- Post processing and results plotting.

3.4.1 Loading experimental data

All the experimental data are loaded in a structural variable. Here there are informations about the test, thermocouples measurements, thermocouples position, pressure information etc. For the purpose of this thesis the interest is mainly in the thermocouple measurements and relative positions.

3.4.2 User input

In this section of the code the user:

- Select how to interpolate the heat flux in a plane (x-y). Constant or parabolic options are available.
- Remove selected thermocouples from calculation.

- Calculate or not a new sensitivity matrix for the convergence of the optimization process.
- Define the maximum number of iteration in a single time step.
- Free convection coefficient (default value is 10 $W/(m^2K)$).
- Choose to smooth (in time) or not the thermocouples measurements.
- Set the number of nodes along x,y and z.
- Define the initial temperature.
- Choose what thermocouples to use in solving the inverse method. As already said only 1 mm Thermocouples will be used.
- Choose the time step Δt.
- Input material properties ρ, c_p, λ (which are considered to be constant).
- Input geometry chamber information.
- Choose starting and ending time of the simulation.
- Set the length of the copper block before the faceplate.

3.4.3 Domain discretization

The computational domain is discretized making sure that nodes on the boundaries exist (internal and external). As mentioned in Section 3.2 $\Delta y = \Delta x$.

3.4.4 Thermocouples measurements managing

This section of the code makes the thermocouples measurements (of chosen thermocouples) usable for the computational analysis. Here code read read measurements and positions from experimental data. Then temperatures are smoothed (if user has chosen to smooth) in time using a moving average on 3 points. Smoothing is very important since oscillations due to precision errors are reduced in the measurements and errors on heat flux estimation are reduced (see Section 4.2).

3.4.5 Time transformation

This section of the code transforms the time domain in time segment. Each time segment is composed of two time steps. In each time segments the heat flux will be constant and the direct problem will be solved two times with the same heat flux.

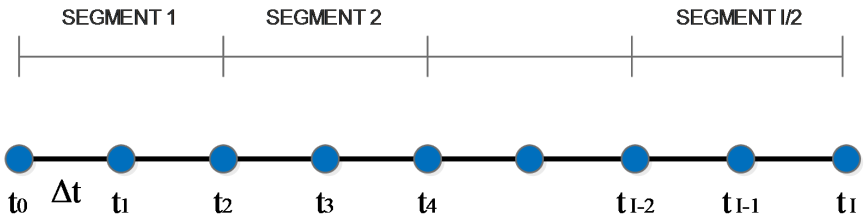


Figure 3.13: Time transformation.

3.4.6 Parameters points definition

Here parameter points are defined (positions in which heat flux will be optimized in order to reach the convergence through inverse method). The code, automatically, sets these parameters at the same locations of the thermoucoples projected on the hot gas wall. The user can also set other points manually. The number of parameters (N) is set set equal to the number of thermocouples (M).

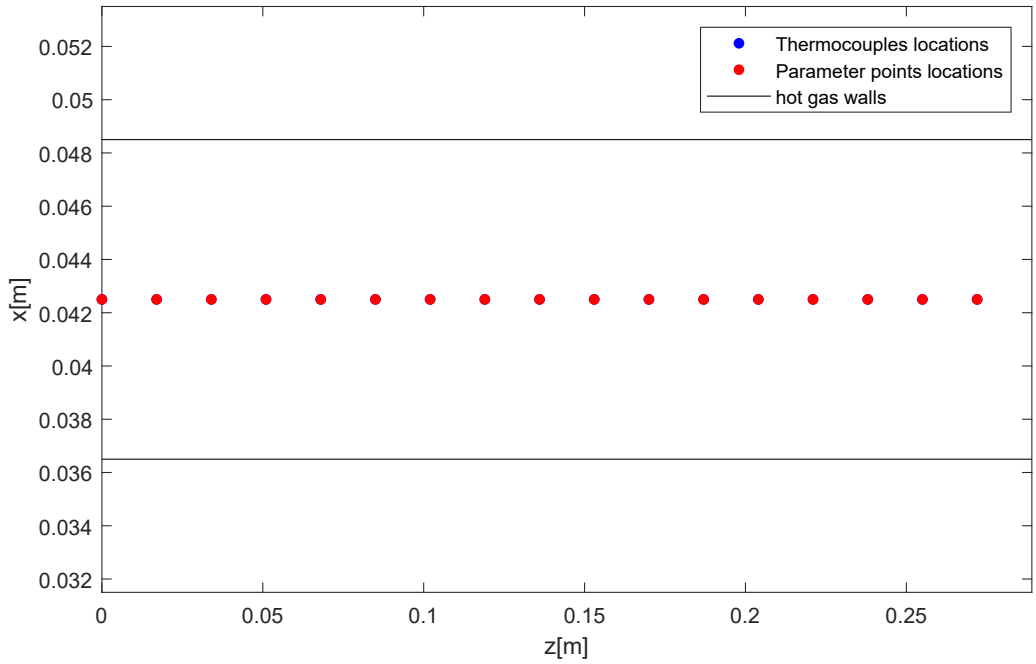


Figure 3.14: Parameter points and thermocouple locations (x-z view).

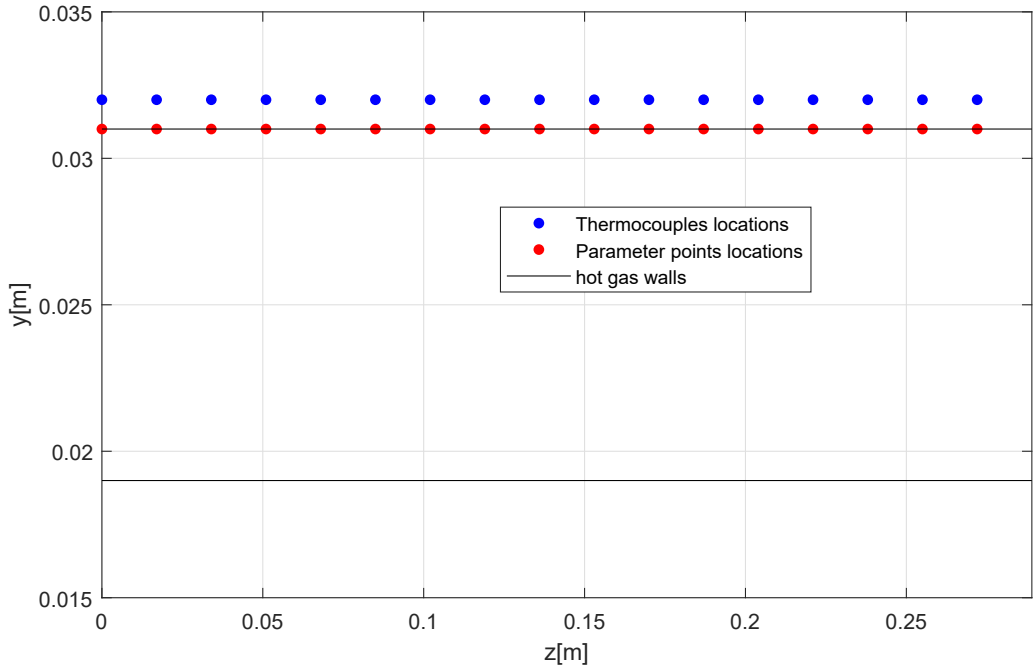


Figure 3.15: Parameter points and thermocouples locations (y-z view).

Figures 3.14 and 3.15 show a typical pattern for thermocouples and parameter points locations.

3.4.7 Heat flux first estimation

As already mentioned, the boundary heat fluxes are not known, so they have to be guessed. It has been explained in Section 3.3 how to update heat flux in parameters points for the next iteration (Eq. (3.30)). But at the first iteration for each time step a manual first guess has to be set. For the first estimation, a linear heat flux with respect to z axis was implemented. Although first guess heat flux has been initialized in this way, after the combustion is started, the first heat flux guess for the following time step is the converged solution from the previous one, which accelerates the convergence. Figure 3.16 shows an example of inizialization over z.

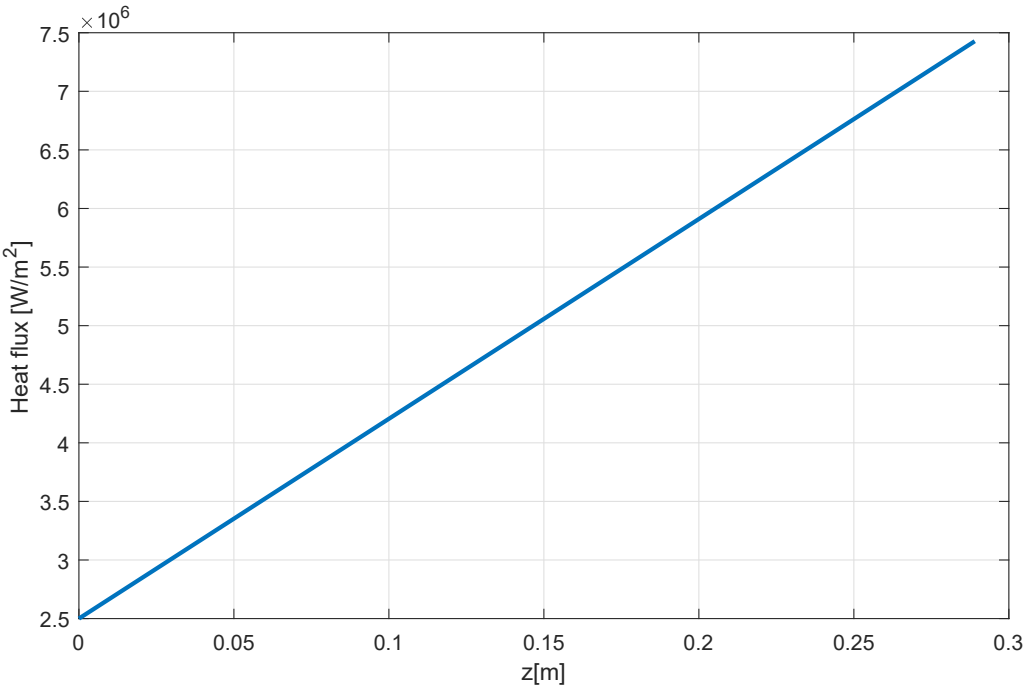


Figure 3.16: Example of heat flux first guess.

3.4.8 Building matrix and boundary condition coefficients

This section prepares the matrix to subsequently solve the direct problem according to equations in Section 3.2. As previously noted, the matrix coefficients will be constant and will depend only on the geometry of the chamber and on the chosen discretization. Also the boundary condition coefficients will be constant. Because of this both matrix coefficient and known terms vector coefficient can be calculated outside the optimization process. According to equations summarized in 3.2.3 the boundary heat flux affect the known terms vector of the linear system. Because of this, the heat flux has to be updated

in this vector for each iteration. Figure 3.17 shows how the matrix coefficients looks like. The dots are the non 0 therm of the matrix.

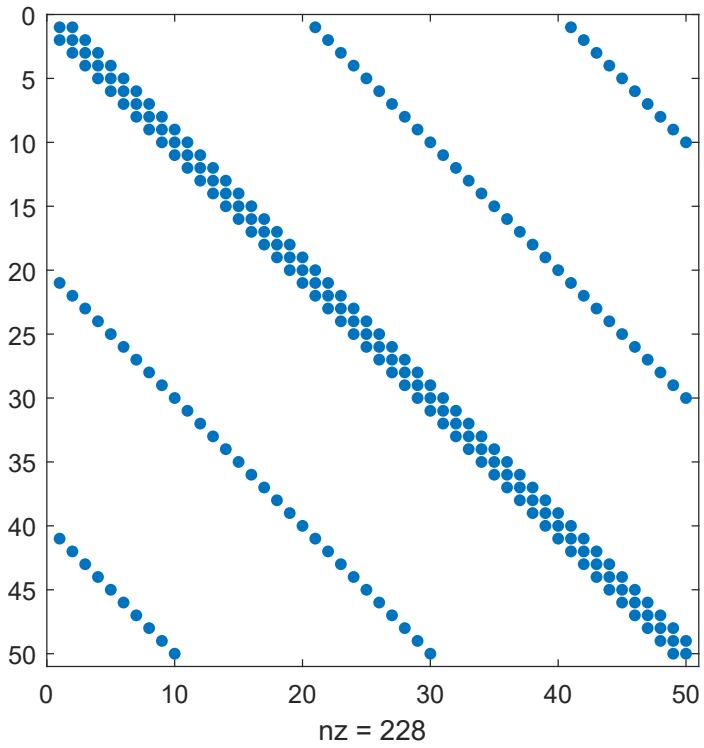


Figure 3.17: Example of the coefficients matrix filling.

3.4.9 Inverse cycle

After all the variables have been initilized, the inverse cycle can start. It basically works as it has been described in Section 3.3, solving at each iteration a direct problem using the discretized equations of Section 3.2. But as it has been disclosed in Section 3.4.5 the optimization process is done on a time segment and not on a single time step. Figure 3.18 better explains how the optimization process on segments works.

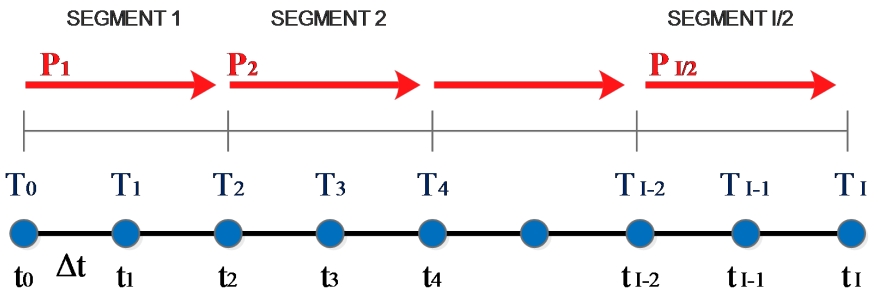


Figure 3.18: Optimization process on segments.

As an example the optimization process on time segment 2 is explained:

- The temperatures of all computational nodes vector at time t_2 are the initial condition of the optimization process in time segment 2. Note that they are the results of the optimization process in time segment 1.
- The code starts solving the direct problem using the first guess of heat flux from Section 3.4.7. In particular it solves the direct problem for time step from t_2 to t_3 and from t_3 to t_4, using the same heat flux.
- Now, after solving the direct problem two times, the optimization cycle, as seen in Section 3.3, starts. The residual function is calculated according to Eq. (3.37) from t_2 to t_4 using the trapezoid rule. Sensitivity matrix is not computed for each iteration since it would have an high computational cost and so it is calculated outside of the cycle and supposed to be constant. But the fundamental difference is now that the entire optimization process (calculation of step size, direction of descent) is only based on temperatures T_4. This lead to a very accurate matching of temperatures in this point, but temperatures at time t_3 are not optimized.

3.4.10 Post processing and results plotting

Post processing basically extracts information from converged temperatures and heat flux. Essentially the results are heat flux over space and time, and temperatures over space and time. As an example some results from the original code are presented here. The experimantal test on which the presented results are based is characterized by:

- Fuel: CH_4 and Oxidizer: O_2.
- Nominal chamber pressure=20 bar.
- O/F=3.

Note that also for all next sections this will be the reference test on which all the considerations will be made.

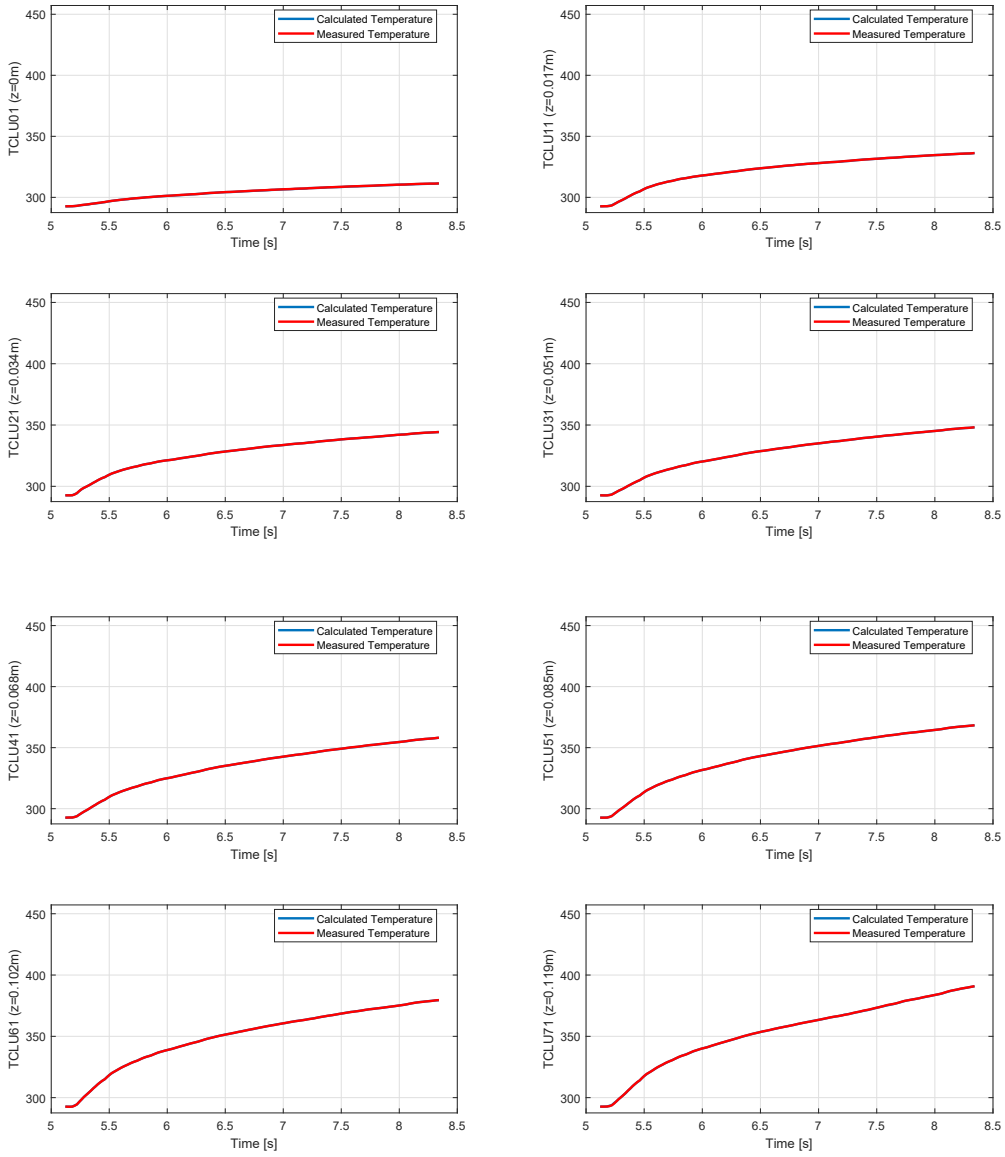


Figure 3.19: Calculated and measured temperatures in thermocouples positions over time (first eight thermocouples).

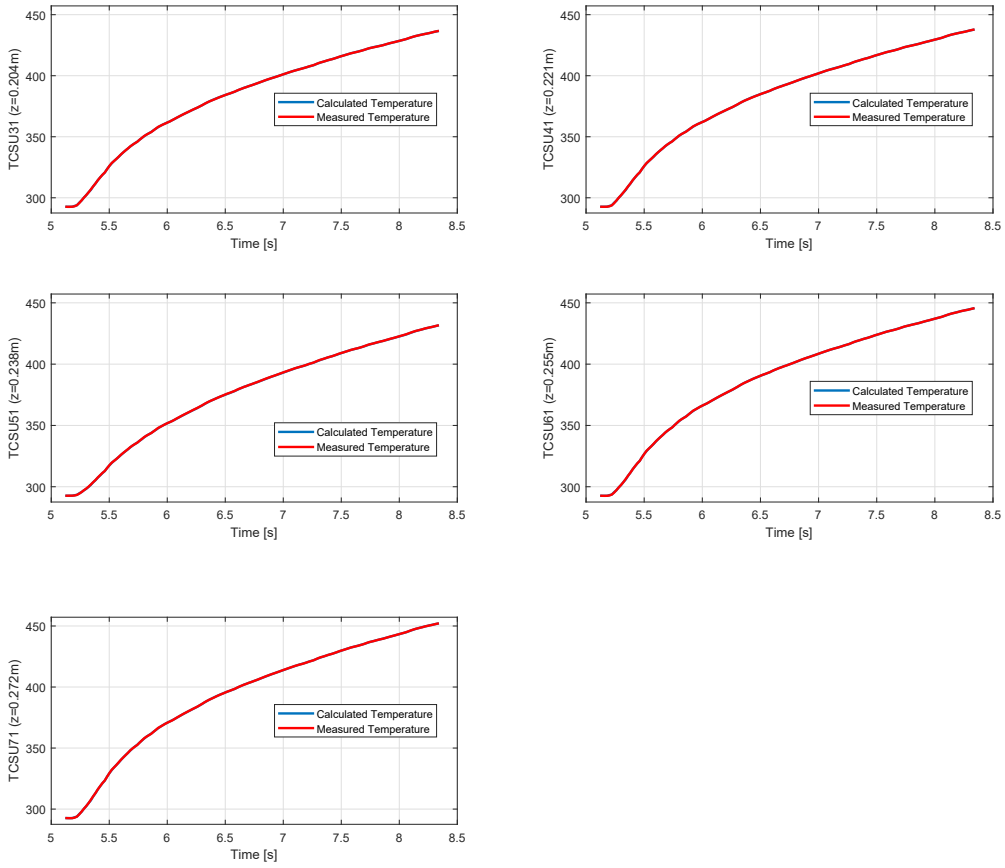


Figure 3.20: Calculated and measured temperatures in thermocouples positions over time (last five thermocouples).

Figure 3.20 shows that computational temperatures match in a good way with measured one at the end of the optimization.

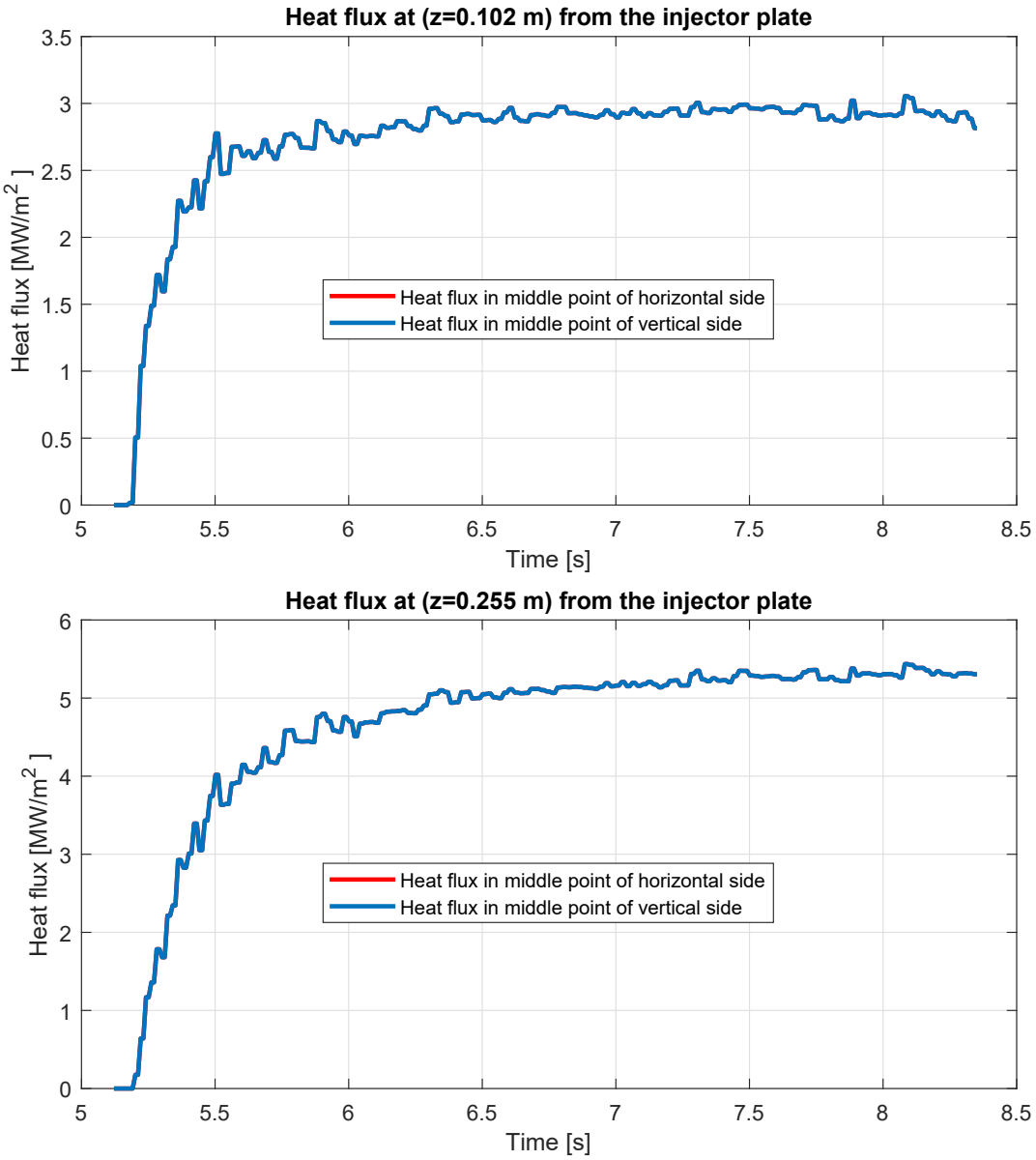
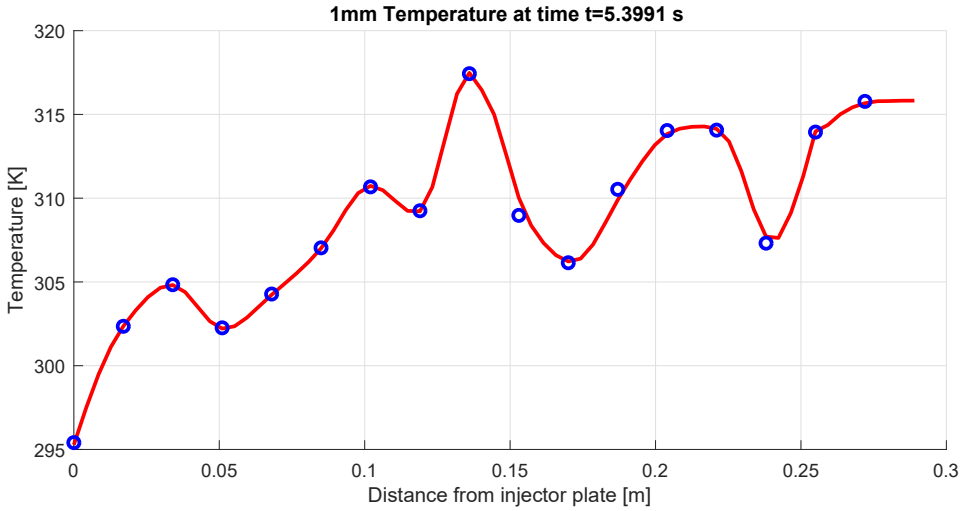
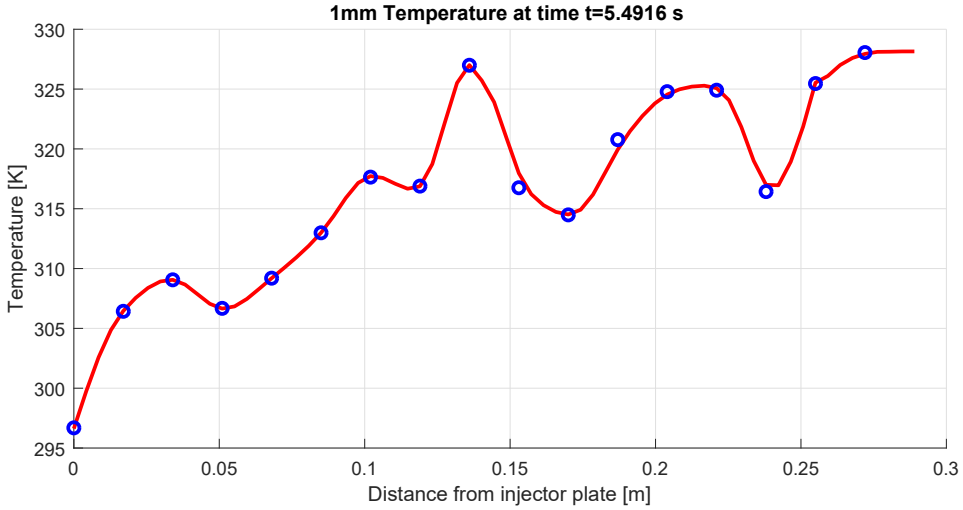
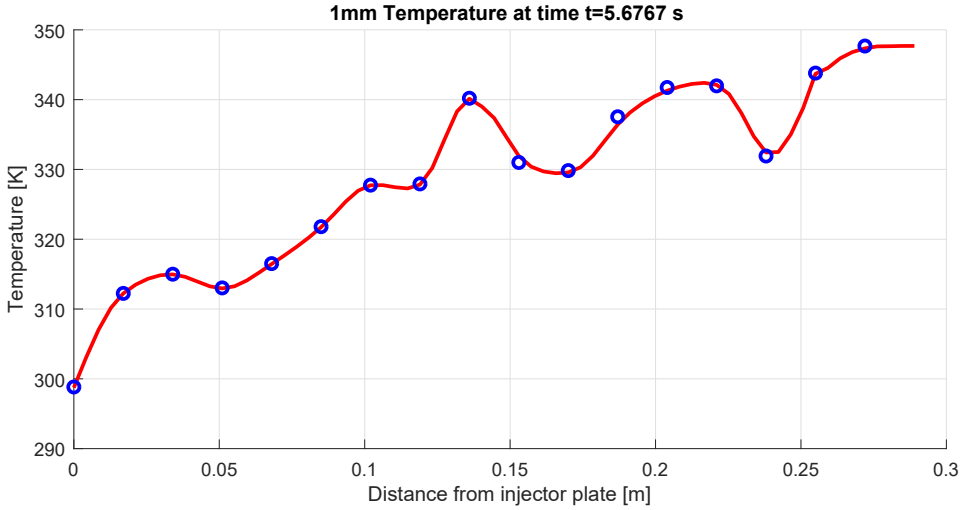


Figure 3.21: Heat flux at $z = 0.102$ m and $z = 0.255$ m over time.

Figure 3.21 shows the evolution of heat flux over time at some z positions. One can note that the heat flux reach an almost stationary values after about 1 s.


1mm Temperature at time t=5.3991 s
Temperature [K]
320
315
310
305
300
295
0
0.05
0.1
0.15
0.2
0.25
0.3
Distance from injector plate [m]


1mm Temperature at time t=5.4916 s
Temperature [K]
330
325
320
315
310
305
300
295
0
0.05
0.1
0.15
0.2
0.25
0.3
Distance from injector plate [m]


1mm Temperature at time t=5.6767 s
Temperature [K]
350
340
330
320
310
300
290
0
0.05
0.1
0.15
0.2
0.25
0.3
Distance from injector plate [m]

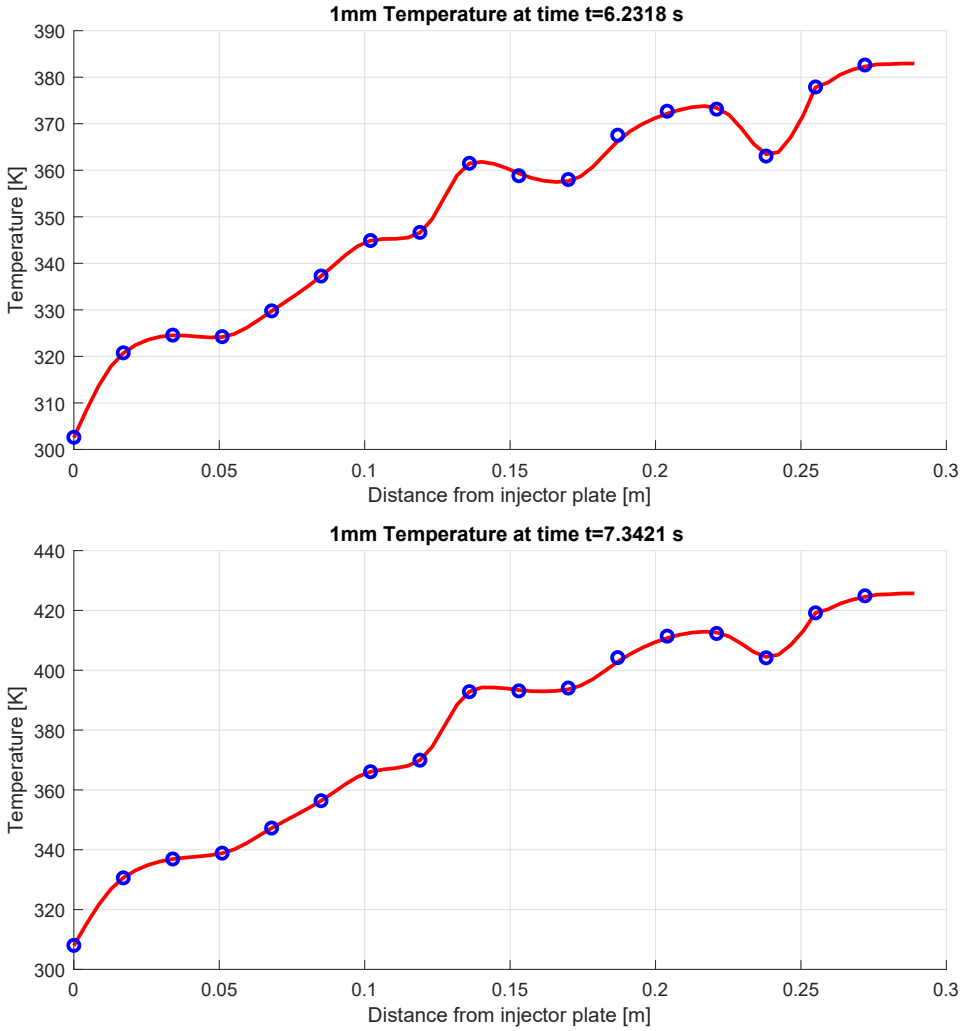


Figure 3.22: Thermocouples temperatures over z at different times.

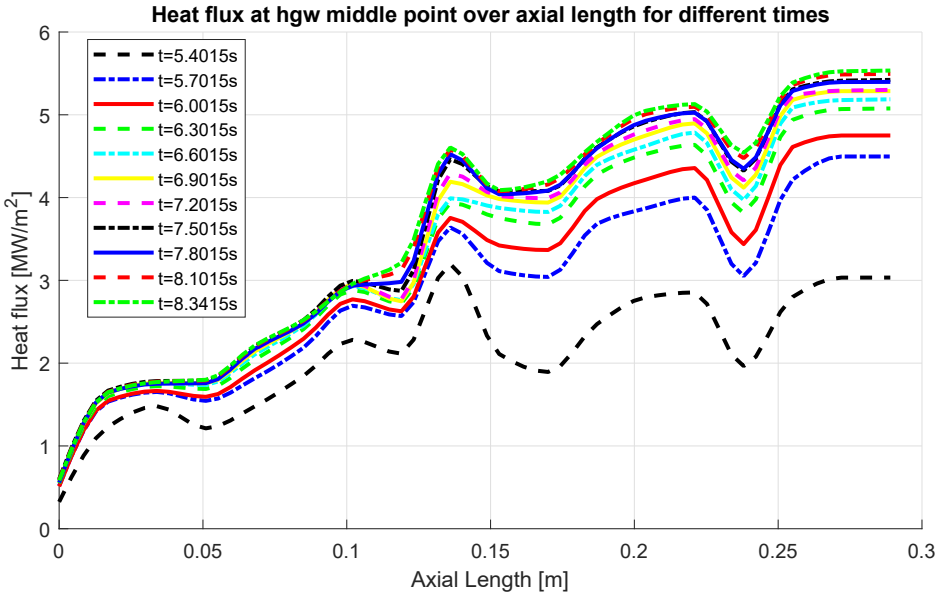


Figure 3.23: Heat flux over z at different times.

3.4.11 Oscillation over z

In Figure 3.23 heat flux profiles over z for different times are plotted. It is possible to note that some oscillations are present along z for all the evalauted times. Theses oscillations could be explained by the presence of the igniter which is mounted to the side wall of the combustion chamber in the region near the faceplate [4]. But the igniter is operative only for the first 0.5 s of the experiment. So these oscillations are expected to disappear after this time, however they are still present in the inverse method results. These oscillations seems to be non physical since the combustion processe should develop along z axis increasing the heat flux until it is completely developped and the heat flux shows a flat. The main goal of this thesis is to try to understand the source of these oscillations. It is also interesting to look at the temperatures over z for different times (Figure 3.22). The measured temperatures show oscillations over z at time t $>$ 0.5 s after burning, but it is important to note that altough temperatures and heat flux are related, oscillations in temperatures don't necesarily lead to oscillations in heat flux. This is true because as seen in Section 3.2 heat flux relates also to the derivative of temperature (in time and space) and not only its absolute value.

Chapter 4

Code modifications

This chapter describes the preliminary modifications made on the code before implementing the errors estimation. The goal of these modifications was to understand the nature of the oscillations along z and try to get rid of them.

4.1 Modification of time segments

As explained Sections 3.4.5 and 3.4.9 the optimization process was not operated on the single time steps, but on time segments (a segment was composed of two time steps). This means that each iteration costs two solution of the direct problem. To reduce the computational cost, the time step for the solution of the direct problem was made coincident to the optimization segment, as shown in Figure 4.1.

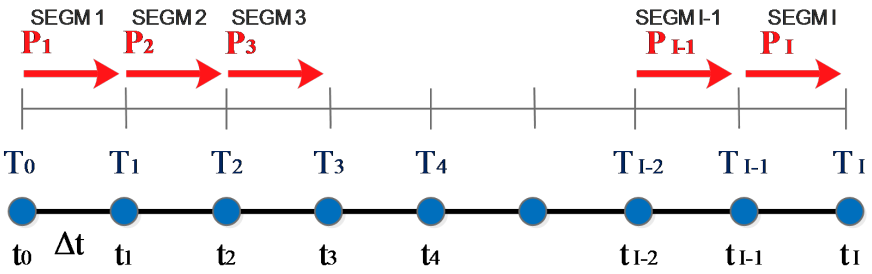


Figure 4.1: New optimization process on time segments.

Now that optimization segments and time step coincide, each iteration costs only one solution of the direct problem. The advantage in computational time is very high when the optimization is performed on 0.02 s long segments (which is the sampling interval of the thermocouples and so the most interesting case). In this case, one simulation costs half the time with respect to the old time management (basically because one iteration lasts

half since only one direct problem per iteration is solved instead of two). If optimization is performed on larger segments, there is still an advantage in computational time but it is not so significant.
Note that calculating the temperature in two steps instead of one step leads to a smaller discretization error, so from a theoretical point of view, the previous time handling was more accurate. But considering all the other sources of error (Chapter 6), discretization error from the direct solver is negligible. Results obtained with this new time handling do not show a measurable deviation from the previous ones. As an example see Figure 4.2. The test case is the same of Figure 3.21 and also time optimization segments are equally long (0.02 s).

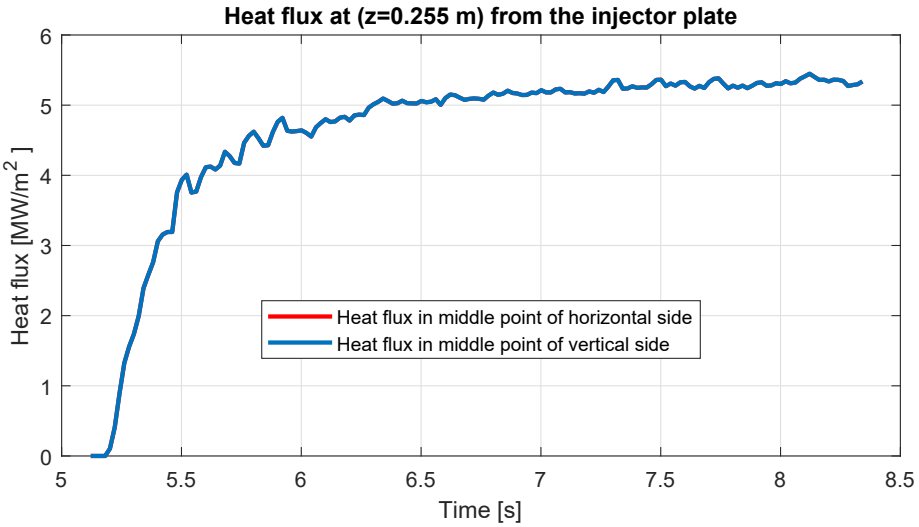


Figure 4.2: Heat flux at $z = 0.255$ m over time, with the new time transformation.

4.2 Smoothing measured temperatures

Apart from their accuracy, thermocouples measurements are affected by precision errors[1]. When a possibility to repeat the measures under unchanged conditions exists, it is possible to reduce precision errors (uncertainty in measurements) by calculating an average value for the measured data.
Unfortunately, since for the purpose of this thesis one has to deal with transient temperatures in the start up of a combustion chamber it is almost impossible to repeat the measurement under unchanged conditions.
Figure 4.3 gives an idea of the thermocouples precision. It refers to a time interval in

[1]Accuracy errors will be discussed in Section 6.1.2.

which the combustion chamber is off, so the temperature should be constant. Left side of the figure shows the measured temperatures over time. Right side of the figure shows temperature interval (x axis) and the number of data which correspond to the respective tempererature interval (y axis). One can note that the shape resembles a gaussian function according to the fact that the measurement is affect by random errors. The average value of this measurement is 292.7 and the standard deviation is about 0.1 K.

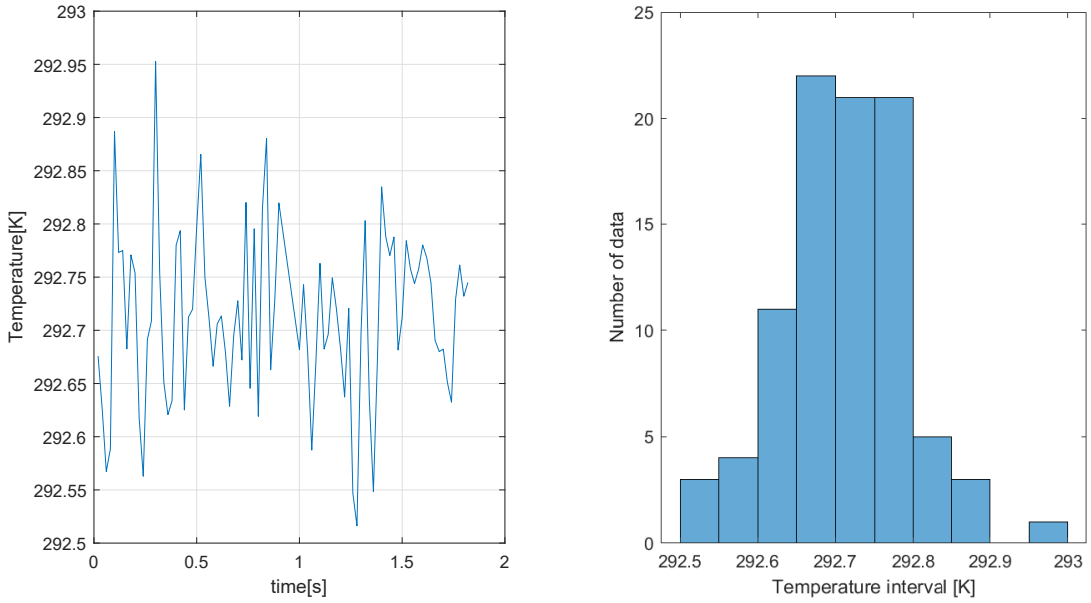


Figure 4.3: Measured temperatures before to start the combustion chamber test.

Repeated measurements allow not only to obtain a better idea of the actual value, but also enable to characterize the uncertainty of measurement.
Figure 4.4 shows a list of quantities that are very useful in data analysis. Here the value obtained from a particular measurement is x. The measurement is repeated N times. If N is small, usually no more than 5 to 10 formulas in Figure 4.4 are valid [17]:

Mean (x_{avg})	The average of all values of x (the "best" value of x)	$x_{avg} = \frac{x_1 + x_2 + \cdots + x_N}{N}$
Range (R)	The "spread" of the data set. This is the difference between the maximum and minimum values of x	$R = x_{max} - x_{min}$
Uncertainty in a measurement (Δx)	Uncertainty in a single measurement of x. You determine this uncertainty by making multiple measurements. You know from your data that x lies somewhere between x_{max} and x_{min}.	$\Delta x = \frac{R}{2} = \frac{x_{max} - x_{min}}{2}$
Uncertainty in the Mean (Δx_{avg})	Uncertainty in the mean value of x. The actual value of x will be somewhere in a neighborhood around x_{avg}. This neighborhood of values is the uncertainty in the mean.	$\Delta x_{avg} = \frac{\Delta x}{\sqrt{N}} = \frac{R}{2\sqrt{N}}$
Measured Value (x_m)	The final reported value of a measurement of x contains both the average value and the uncertainty in the mean.	$x_m = x_{avg} \pm \Delta x_{avg}$

Figure 4.4: Significative quantity for small set of measurements.

These formulas are valid for multiple measurements on the same phenomena under unchanged conditions. Since it is not possible to apply this concept for our purpose, the concept of 'moving average' is introduced. The idea is to calculate the average value of temperature at time t_i using informations coming from measurements at time t_{i-1},t_{i-2}...,t_{i-N} and t_{i+1},t_{i+2}...,t_{i+N} in order to get rid of random errors. The original code smoothed the thermocouple reading on 3 points using a moving average. This means that:

$$T_{s\,i} = \frac{T_{i-1} + T_i + T_{i+1}}{3} \tag{4.1}$$

where the subscript s refers to the smoothed measurement. For multiple measurements on the same phenomena the average value becomes more and more precise as the number of measurements increases. Although the uncertainty of any single measurement is always Δx ,the uncertainty in the mean Δx_{avg} becomes smaller (by a factor of $1/\sqrt{N}$) as more measurements are made.

This concept, extended to transient temperatures allows to think that the more points are used to smooth, the more the error on average value will decrease. The basic idea is right, but the way of smoothing has to be slightly changed. The smoothing parameters are basically two:

- Number of points used.

- Smoothing method.

As already mentioned, the starting version of the code used 3 points and the moving average method. Using more points, the moving average doesn't guarantee good results. One has to use others methods such as linear regression or quadratic regression. The risk of smoothing on many points, is to smooth physical phenomena instead of smoothing random errors and therefore a trade-off has to be made.

4.2.1 Skewness and Kurtosis for transient measurements

Measurements often have distributions that are nearly normal (or Gaussian) [1]. Figure 4.3 shows that also thermocouples measurements follow this behaviour. According to this, the more the temperature distribution resembles a gaussian distribution, the more one can be sure that the smoothing is operating on random phenomena. Two important quantities that give a measure on how much a cluster of measurements is near to a gaussian are:

- Skewness.
- Kurtosis.

Skewness, for a set of N measurements is defined as:

$$S = \frac{(1/N)\sum_{i=1}^{N}(x_i^3 - \bar{x}^3)}{\sigma^3} \tag{4.2}$$

where $\bar{x}$ is the average value of the measurements and σ is the standard deviation defined as:

$$\sigma = \sqrt{(1/N)\sum_{i=1}^{N}(x_i^2 - \bar{x}^2)} \tag{4.3}$$

Skewness is 0 for a Gaussian distribution and in general for symmetrical distributions. Kurtosis is defined as:

$$K = \frac{(1/N)\sum_{i=1}^{N}(x_i^4 - \bar{x}^4)}{\sigma^4} - 3 \tag{4.4}$$

Kurtosis is 0 for a Gaussian distribution.[2]
The aforementioned definitions are right for a set of measurements of the same phenomena under unchanged conditions. For a set of transient temperatures they have to be slightly modified. An example (Figure 4.5) can better clarify how these definitions have to be modified. Suppose to have measured the green temperatures and to have set a linear smoothing on 11 points. Figure 4.5 shows the smoothing process on temperature T_i. This

[2]Note that the fact that S and K are 0 is valid when $N \to \infty$.

means that the smoothed value assigned at temperature T_i will be the result of the linear regression of the set of temperatures from T_{i-5} to T_{i+5}. Note that since the smoothing is on eleven points, only the first five values before T_i and the first five values after T_i take part in the smoothing of temperature T_i. Other temperatures don't affect the smoothing process on the i-th temperature. Standard deviation, skewness and kurtosis are now calculated as follows:

$$\sigma_i = \sqrt{(1/N) \sum_{k=i-(N-1)/2}^{i+(N-1)} (T_{k\,m}^2 - T_{k\,s}^2)} \tag{4.5}$$

$$S_i = \frac{(1/N) \sum_{k=i-(N-1)/2}^{i+(N-1)/2} (T_{k\,m}^3 - Tk\ s^3)}{\sigma_i^3} \tag{4.6}$$

$$K_i = \frac{(1/N) \sum_{k=i-(N-1)/2}^{i+(N-1)/2} (T_{k\,m}^4 - T_{k\,s}^4)}{\sigma_i^4} - 3 \tag{4.7}$$

Here the subscript:

- i indicate the i-th temperature (temperature on which the smoothing process is operating). This index is fixed.
- s refers to the smoothed value.
- m refers to the measured value.
- N is the numbers of point on which the smoothing is performed (it is always an odd number otherwise the smoothing process will be asymmetrical).
- k refers to the k-th temperature and scrolls along the temperatures which take part in the smoothing process of temperature i-th.

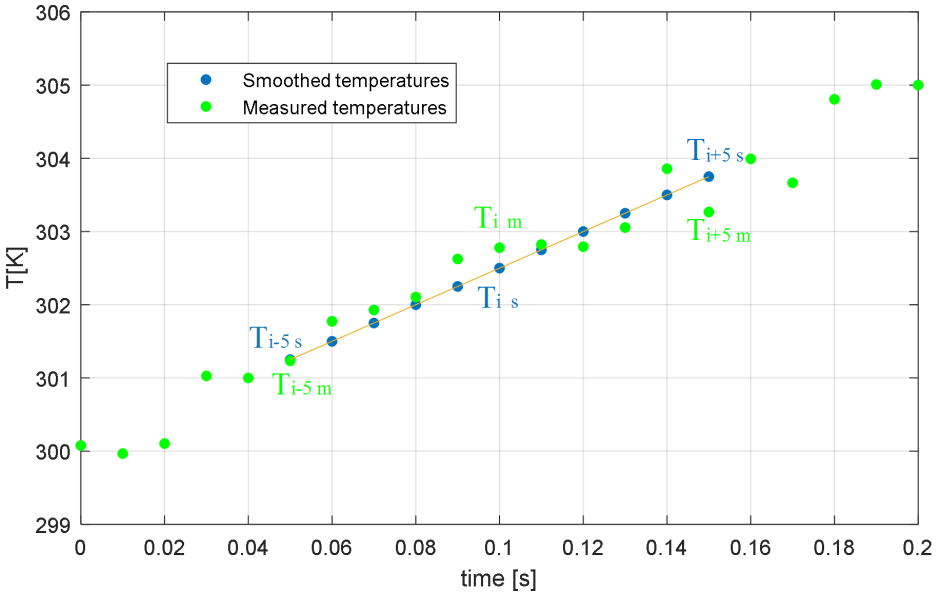


Figure 4.5: Smoothing example on temperature T_i.

Now that new versions of skewness and kurtosis have been created it is possibile to use them to verify if the smoothing is performing on random variables or not.

4.2.2 Skewness and kurtosis for small number of measurements

As previous mentioned, if the number of measurements tends to infinite, skewness and kurtosis are 0 if the measurements are affected by random errors. If the number of measurements is not infinite, even if they are affected only by random errors, values for skewness and kurtosis show a certain variability.
To have a measure of this variability, the process of Figure 4.6 has been used.
The process works as follows.
A vector of random measurements i is generated and skewness and kurtosis are calculated. In order to have a statistical idea on how skewness and kurtosis behave for a given number of measurements, one need to repeat this process many times (theoretically infinite times). In the algorithm the process is repeated 10000 times for each i. This process allows to have a measure of the variability of skewness and kurtosis when the number of considered points doesn't tend to infinity but it is small. Results from this process are shown in Table 4.1. It contains the average value for kurtosis and skewness for a given number of measurements, and also the standard deviation. The range in which kurtosis and skewness have to be included in order to be sure that the smoothing process is acting on random

phenomena are:

$$-2\sigma(S) + \bar{S} < S_i < \bar{S} + 2\sigma(S) \tag{4.8}$$

and

$$-2\sigma(K) + \bar{K} < K_i < \bar{K} + 2\sigma(K) \tag{4.9}$$

Values for $\sigma(S)$, $\bar{S}$, $\sigma(K)$, $\bar{K}$ are defined in Table 4.1 for different numbers of measurements (which in our case is equivalent to the number of points on which the smoothing process operates)

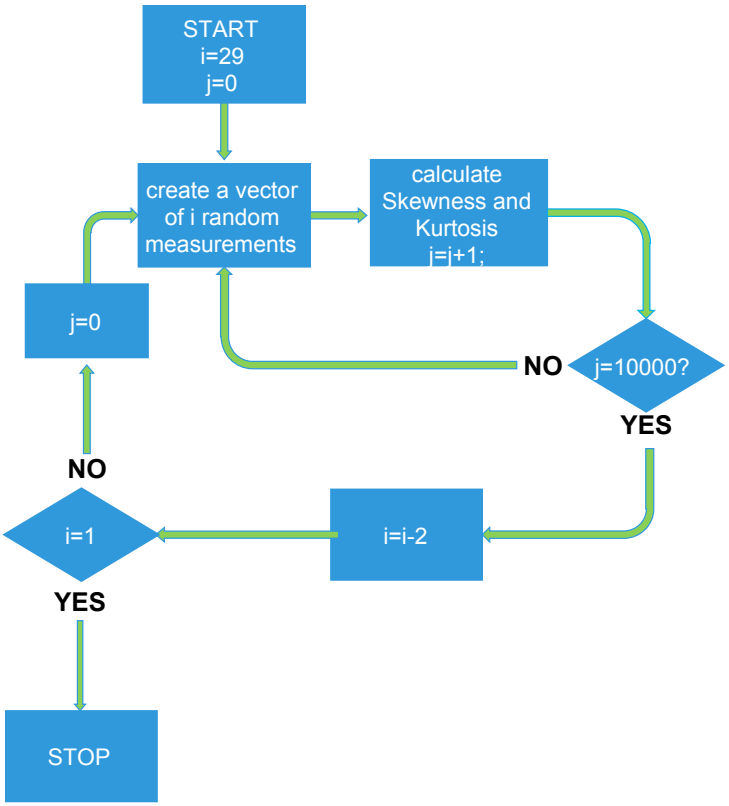


Figure 4.6: Process to estimate variability skewness and kurtosis.

N_{meas}	$\bar{S}$	σ (S)	$\bar{K}$	σ (K)
29	0,00008	0,2144	-0,9741	0,3203
27	0,00109	0,2326	-0,9520	0,4823
25	-0,00073	0,2527	-0,9336	0,5604
23	-0,00200	0,2778	-0,9134	0,5858
21	-0,00066	0,3037	-0,8960	0,5765
19	-0,00086	0,3330	-0,8837	0,5519
17	-0,00086	0,3644	-0,8798	0,5139
15	0,00244	0,3984	-0,8903	0,4720
13	0,00309	0,4301	-0,9282	0,4280
11	-0,00039	0,4612	-1,0003	0,3907
9	0,00049	0,4795	-1,1344	0,3552
7	-0,00300	0,4817	-1,1344	0,3230
5	0,00020	0,4370	-1,7217	0,2945

Table 4.1: Skewness and kurtosis for small number of measurements.

4.2.3 Smoothing cycle

At the beginning of Section 4.2 it was said that the smoothing parameters are basically two:

- Number of points .
- Method.

A cycle to optimize the use of the two has been arranged. In general smoothing on many parameters should reduce precision errors. But one needs to know which is the best smoothing method and also be sure to not smooth physical phenomena. In order to be sure to not smoothing physical phenomena, skewness and kurtosis calculated as explained in Section 4.2.2 should stay in the ranges defined in Eq 4.8 and 4.9.
Four method of smoothing have been used [3]:

- Moving average (Matlab command 'moving').
- Sgolay (a sort of moving average weigthed with the distance) (Matlab command 'sgolay').
- Linear regression (Matlab command. 'lowess')

[3]The Matlab function used to smooth is "smooth".

- Parabolic regression (Matlab command 'loes').

The smoothing process acts independently on all the 17 thermocouple measurements. So it is explained for a single thermocouple. Suppose to have I measurements in time for the considered thermocouple. The smoothing process is explained in flow chart in figure 4.7[4]. When the vector of measured temperatures of a single termocouple enters in the cycle, the situation is this:
$\mathbf{T} = (T_{1m}, T_{2m}, \ldots, T_{im}, \ldots, T_{(I-1)m}, T_{Im})$
where the subscript m indicates that the temperatures are the ones measured by the thermocouples. In an intermediate phase of the cycle the situation could be this:
$\mathbf{T} = (T_{1m}, T_{2m}, \ldots, T_{is}, \ldots, T_{(I-1)s}, T_{Im})$
where the subscript s means that the considered value of temperature is the smoothed one. Some temperatures reach the randomness requirement (skewness and kurtosis in the ranges defined in Eq. 4.8 and 4.9), for example smoothing on 15 points, while other need to be smoothed for example on 7 points to satisfy the randomness requirement. At the end of the smoothing cycle :
$\mathbf{T} = (T_{1s}, T_{2s}, \ldots, T_{is}, \ldots T_{(I-1)s}, T_{Is})$
Figures 4.8 and 4.9 show how smoothed temperatures look like. The comparison has been done:

- Between unsmoothed temperatures and smoothed on 3 points.
- Between smoothed temperatures on 3 points and smoothed temperatures using the cycle

. Now, it also possible to estimate the error on the smoothed value. Recalling that the uncertainty on the mean value, for a set of N measurements on the same phoenomenon is:

$$\Delta x_{avg} = \frac{x_{max} - x_{min}}{2\sqrt{N}} \tag{4.10}$$

Eq.(4.10) has been readapted, for a set of transient measurements in the following way:

$$\Delta T_{i\_smoothed} = \frac{max(T_{smoothed}(i-(N/2-1):i+(N/2-1) - T_{measured}(i-(N/2-1):i+(N/2-1)}{2\sqrt{N}} \tag{4.11}$$

[4]The entire process has been explained until smoothing points=7, since for 5 and 3 points a moving average has been used. And in particular, for 3 points no randomness analysis is done and if some temperatures haven't reached yet the randomness requirements, they are automatically smoothed on 3 points.

where N is the number of points on which the i-th temperature has been smoothed. $max(T_{smoothed}(i-(N/2-1):i+(N/2-1)-T_{measured})$ is the maximum difference between measured and smoothed value, of the set of measurements used to smooth the i-th measurement. For example in Figure 4.5 it is the maximum distance between blue and black dots.

Comparison between not smoothing, smoothing with cycle and smoothing on 3 points are shown in Figures 4.8, 4.9, 4.10, 4.11. In particular Figures 4.8 and 4.9 show how temperatures look when they are unsmoothed, smoothed on 3 points or smoothed with cycle. One can note that the difference in temperature between 3 points and cycle seems to be small. But looking at Figures 4.10 and 4.11 it is possible to see that a difference in heat flux over time exists. It is important to underline that since the error due to precision has been reduced using smoothing cycle, it has sense also to set a more restrictive stopping criteria for the optimization, in order to better match temperatures. In fact results in Figures 4.10 and 4.11 are obtained using standard deviation equal to 0.1 K in the calculation of stopping criteria(Eq. (3.39)).

No appreciable difference in results appears in heat flux over z at a given time.

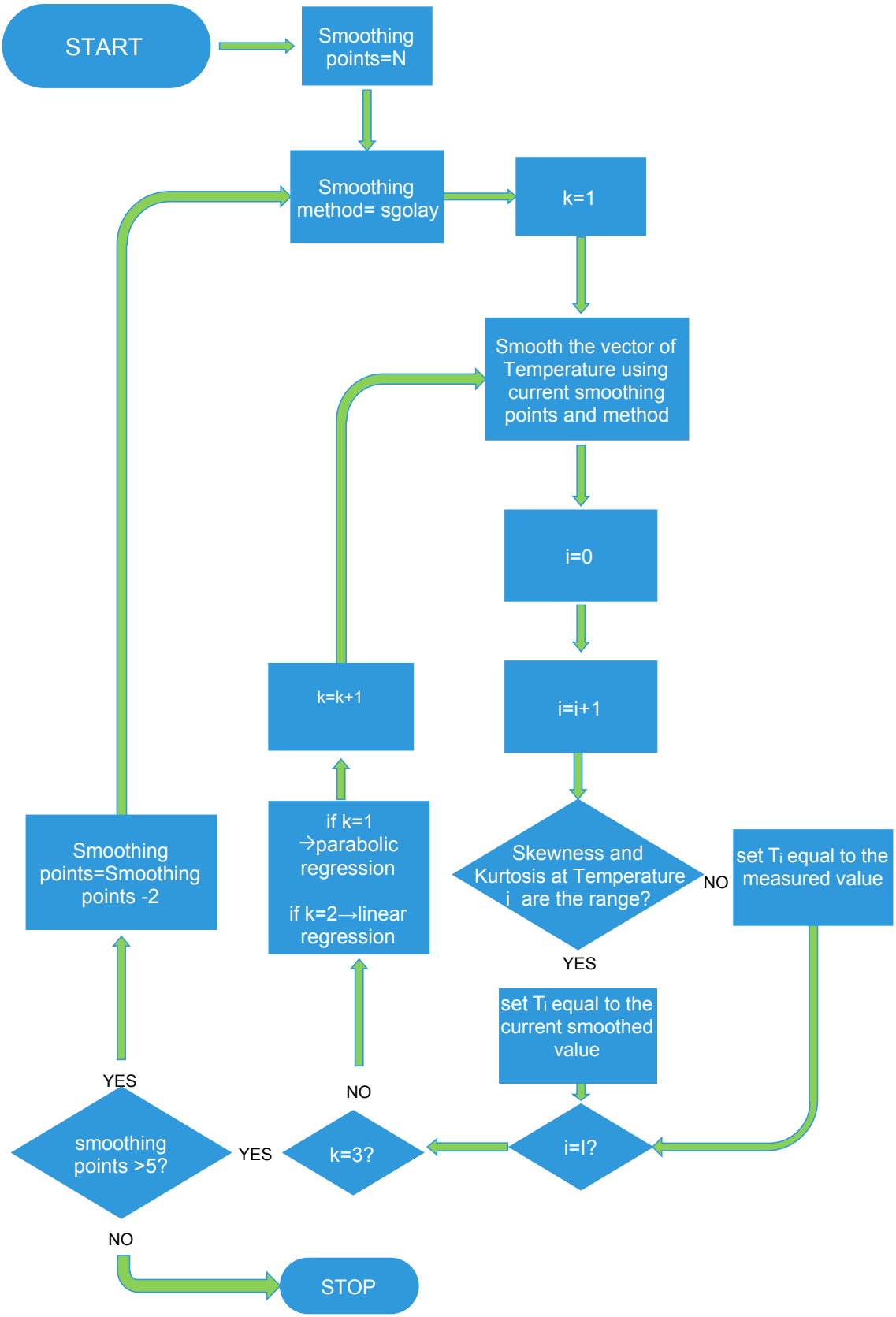


Figure 4.7: Smoothing cycle.

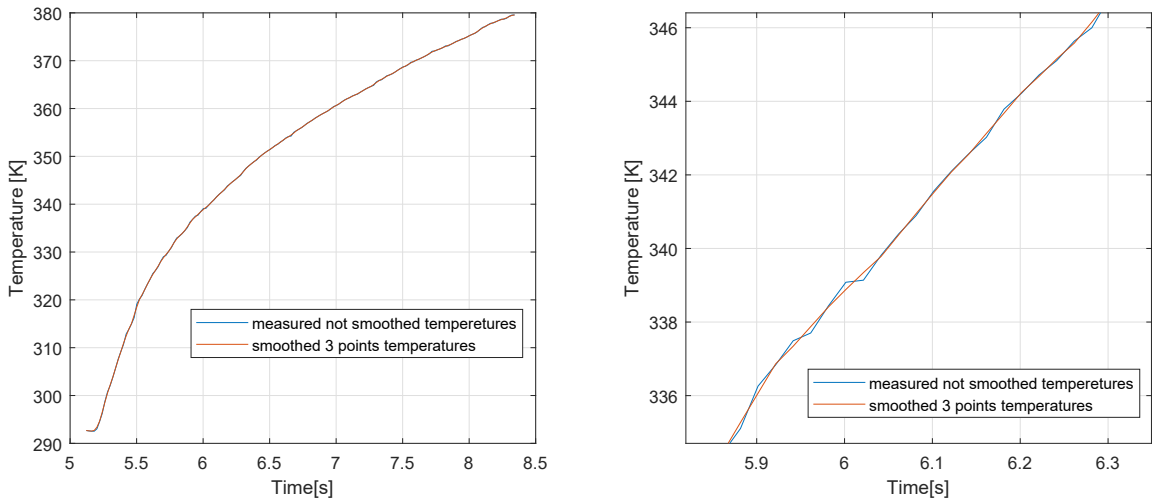


Figure 4.8: Comparison between unsmoothed temperatures ans smoothed ones on 3 points (right figure is a zoom). Thermocouples at $z = 0.1020$ m.

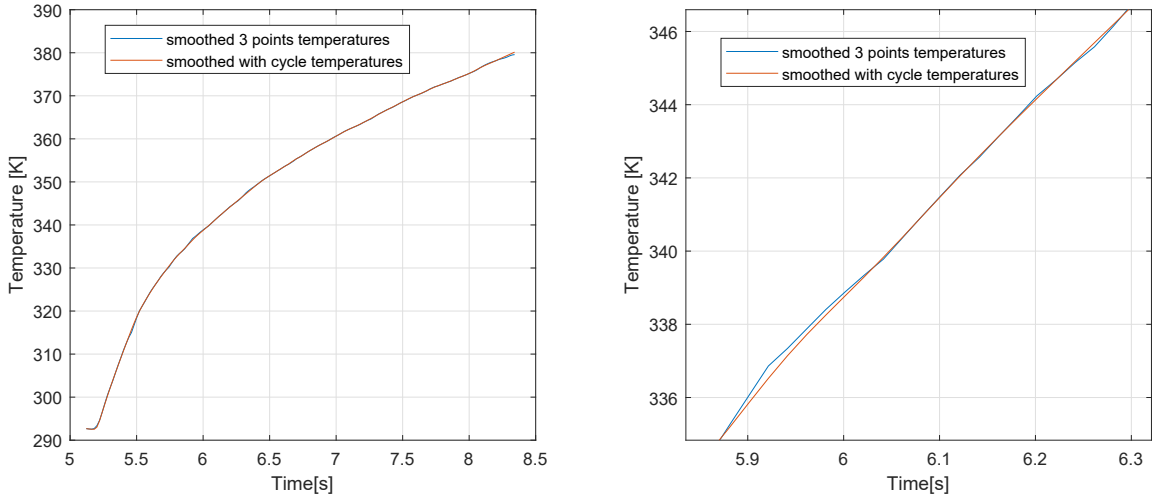


Figure 4.9: Comparison between smoothed temperatures using the cycle ans smoothed ones on 3 points (right figure is a zoom). Thermocouples at $z = 0.1020$ m.

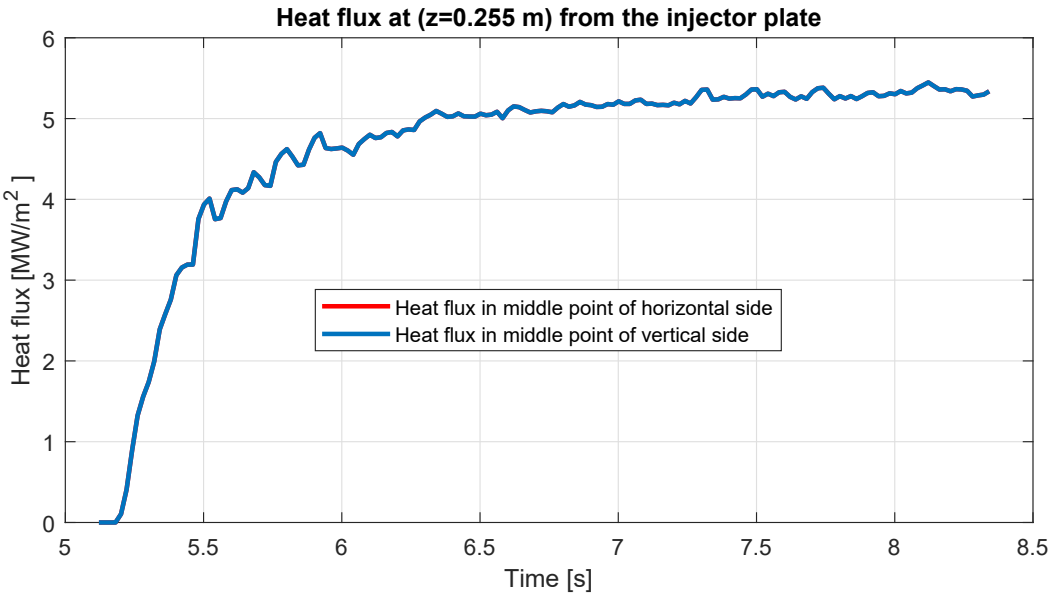


Figure 4.10: Heat flux at $z = 0.255$ m (smoothing on 3 points) over time, with $\sigma = 0.1$ K.

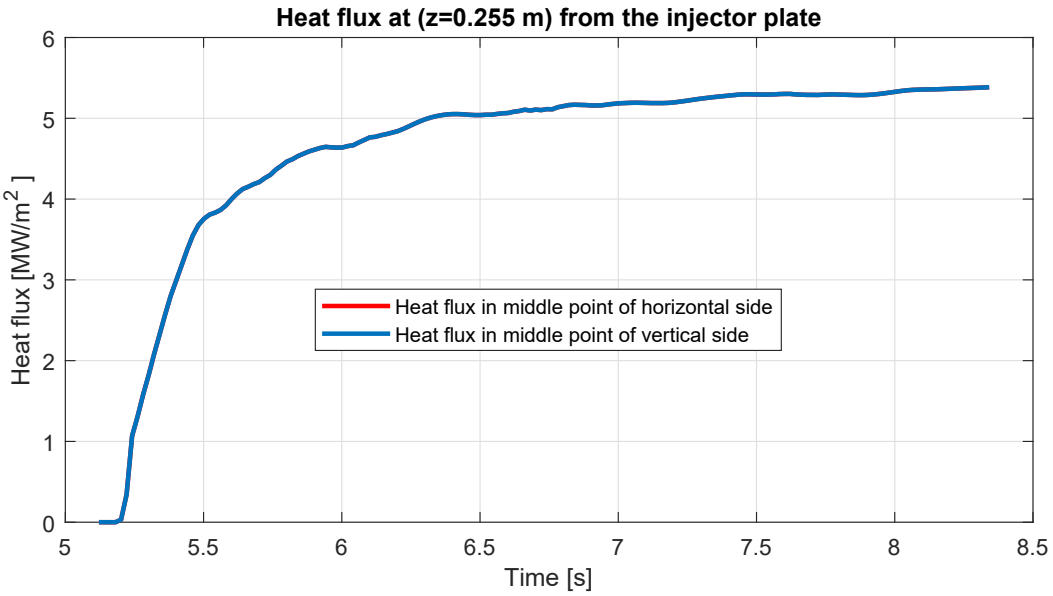


Figure 4.11: Heat flux at $z = 0.255$ m (smoothing using cycle) over time, with $\sigma = 0.1$ K.

4.3 Newton Rapshon method

Solving the inverse problem for a given time step, basically means to solve a non linear system. In fact the values of computational temperatures in thermocouples positions at time t^{k+1} depend on the values of heat flux parameters:
$\mathbf{T_c}^{k+1} = \mathbf{F}(\mathbf{P}^{k+1})$
where $\mathbf{P}^{k+1}$ is the vector of heat flux parameters which act in time segment $k+1$. In fact time segment $k+1$ starts in t^k and finishes in t^{k+1} and $\mathbf{T_c}^{k+1}$ is the result of an applied heat flux boundary condition on the initial condition temperature $\mathbf{T_c}^{k}$. In particular the function wanted to be 0 is:

$$\mathbf{T_c}^{k+1} - \mathbf{T_m}^{k+1} = \mathbf{F}(\mathbf{P}^{k+1}) = 0 \tag{4.12}$$

Where the subscript m means that the considered temperatures are the ones measured by thermocouples (they come from experiments and not from computation, note that as a matter of facts these temperatures have been smoothed before starting the inverse cycle). The system to be solved is:

$$\mathbf{F}(\mathbf{P}^{k+1}) = \mathbf{T_c}^{k+1}(\mathbf{P}^{k}) - \mathbf{T_m}^{k+1} = 0. \tag{4.13}$$

Such as system can be solved using the Newton Rapshon method [13]. The index k will be neglected, but all the following lines are referring to the solution of the inverse problem for a single time step. The solution is obtained by linearizing the equations starting from an initial guess and then iterating until the norm of the vector $\mathbf{F}(\mathbf{P})$ is smaller then a prescribed threshold value (defined by the stopping criteria).

$$\mathbf{F}(\mathbf{P}^{it}) + \underline{\mathbf{J}}_J^{it} \cdot (\mathbf{P}^{it+1} - \mathbf{P}^{it}) = 0 \tag{4.14}$$

Basically, for each iteration one need to solve the following linear system.

$$\underline{\mathbf{J}}_J^{it} \cdot (\mathbf{P}^{it+1}) = -\mathbf{F}(\mathbf{P}^{it}) + \underline{\mathbf{J}}_J^{it} \cdot (\mathbf{P}^{it}) \tag{4.15}$$

All the values with superscript it are known values coming from the solution of the previous iteration.

4.3.1 Jacobi matrix

$\underline{\mathbf{J}}_J$ is the Jacobi matrix, calculated in a **single time step**:

$$\underline{\mathbf{J}}_J^{it}(\mathbf{P}^{it}) = \begin{bmatrix} \frac{\partial T_1}{\partial P_1^{it}} & \frac{\partial T_1}{\partial P_2^{it}} & \frac{\partial T_1}{\partial P_3^{it}} & \cdots & \frac{\partial T_1}{\partial P_N^{it}} \\ \frac{\partial T_2}{\partial P_1^{it}} & \frac{\partial T_2}{\partial P_2^{it}} & \frac{\partial T_2}{\partial P_3^{it}} & \cdots & \frac{\partial T_2}{\partial P_N^{it}} \\ \frac{\partial T_3}{\partial P_1^{it}} & \frac{\partial T_3}{\partial P_2^{it}} & \frac{\partial T_3}{\partial P_3^{it}} & \cdots & \frac{\partial T_3}{\partial P_N^{it}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial T_N}{\partial P_1^{it}} & \frac{\partial T_N}{\partial P_2^{it}} & \frac{\partial T_M}{\partial P_3^{it}} & \cdots & \frac{\partial T_N}{\partial P_N^{it}} \end{bmatrix}$$

It is computed similarly to the sensitivity matrix defined in Section 3.3. The fundamental difference is that here the temperature T_i is the temperature at time t^{k+1} resulting from applying the heat flux parameters to the boundary walls from time t^k to time t^{k+1}. (For a better understanding on how sensitivity matrix and Jacobi matrix are computed see Section 4.3.2) Note that for this thesis $N = M$ (number of thermocouples is equal to number of parameters, usually 17). Because of this $\underline{\mathbf{J}}_J$ is usually a 17x17 matrix.

$$J_{J\,ij} = \frac{T_i(P_1^{it},P_2^{it},P_3^{it},\ldots,P_j^{it}+\epsilon P_j^{it}+\ldots+P_N^{it}) - T_i(P_1^{it},P_2^{it},P_3^{it},\ldots,P_j^{it}+\ldots+P_N^{it})}{\epsilon P_j^{it}} \quad (4.16)$$

One can note that in general the Jacobi matrix depends on $\mathbf{P}^{it}$ and so a new Jacobi matrix should be calculated at each iteration. But, for the case of interest of this thesis, studies on Jacobi matrix have shown that, not only the sensitivity matrix is almost constant when varying the heat flux parameters in a single time step, but it is also almost constant for different time segments. Since it is independent from time segment and from heat fluxes, it can be calculated once, out of the inverse cycle, using a constant heat flux. Figure 4.12 shows one element of the Jacobi matrix computed at each time step, using the real starting condition of temperatures for each time step and real converged heat fluxes, compared to the sensitivity matrix computed one time, using constant heat fluxes and temperatures at t_0 as starting condition. One can note that, once the combustion has started, the two Jacobi matrixes are almost equal. This allows to calculate the Jacobi matrix one time and to use it to solve the inverse problem in all the time steps.
It was found that the solution of the inverse problem, using the Newton Rapshon method, costs a lot less time than using the conjugate gradient method.

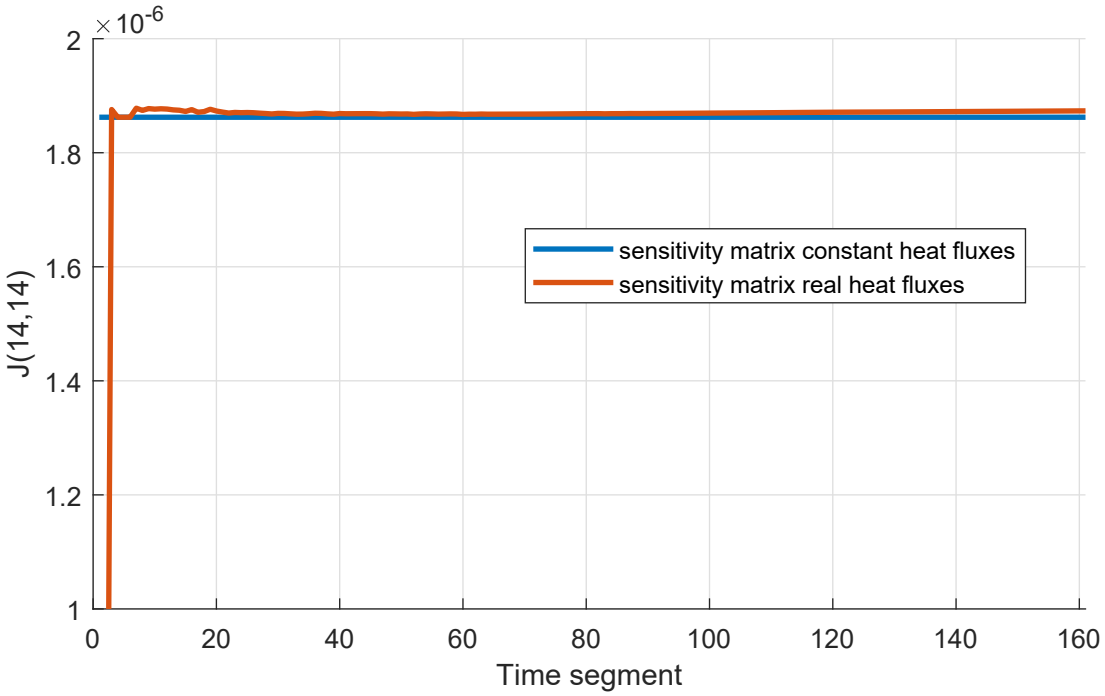


Figure 4.12: Jacobi matrixes comparison.

4.3.2 Differences in computation of Jacobi and Sensitivity Matrix

Two different sensitivity matrixes have been defined in Sections 4.3.1 and 3.3.1. The one defined in 4.3.1 will be called '**Jacobi matrix**' and the one defined in 3.3.1 will be called '**Sensitivity Matrix**'.
This is only for convenience since the two terms are general synonyms.

Jacobi matrix will be indicated as $\underline{\mathbf{J}}_J$.
Sensitivity matrix will be indicated as $\underline{\mathbf{J}}_S$.

Jacobi matrix for time segment $k+1$ is computed in the following way:

- T_c^k is the starting condition temperature of the time segment.
- The direct problem is computed applying the converged heat flux for the given time segment and temperatures T_C^{k+1} are calculated.
- The converged heat flux parameters are perturbed one at a time and the direct problem is solved N times with perturbed parameters (where N is the number of thermocouples equal to the number of parameters).

The **Jacobi matrix** for time segment $k+1$ is computed as follows:

$$J^{k+1}_{J\,ij} = \frac{T_i^{k+1}(P_1^{k+1},P_2^{k+1},P_3^{k+1},\ldots,P_j^{k+1}+\epsilon P_j^{k+1}+\ldots+P_N^{it}) - T_i^{k+1}(P_1^{k+1},P_2^{k+1},P_3^{k+1},\ldots,P_j^{k+1}+\ldots+P_N^{k+1})}{\epsilon P_j^{k+1}} \tag{4.17}$$

- The superscript $k+1$ is referred to the time segment.
- Themperatures T_i are computational temperatures calculated at i-th thermocouple location.
- Paramters P_j are heat flux parameters at j-th paramter point.

Figure 4.13 shows in a schematic way how Jacobi matrix is calculated. Here : $T^{k+1}_{pj} = T^{k+1}(P_1^{k+1},P_2^{k+1},P_3^{k+1},\ldots,P_j^{k+1}+\epsilon P_j^{k+1}+\ldots+P_N^{k+1})$ are the computational temperatures in the domain resulting from solving the direct problem from t^k to t^{k+1} perturbing the j-th heat flux parameter flux and keeping the others constant.

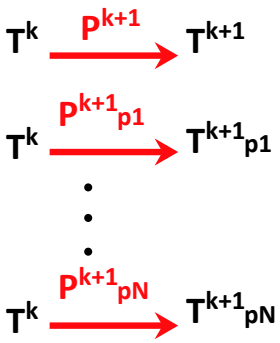


Figure 4.13: Computation of Jacobi matrix

Note that the computation of Jacobi matrix for each time step cost $N+1$ solutions of the direct problem. It is important to note that **the starting condition for computing the Jacobi matrix are the real converged temperatures at time** t^k.

The **sensitivity matrix** for time segment $k+1$ is computed as follows:

- T^0 is the starting condition.
- The direct problem is computed step by step from t^0 to t^{k+1} using converged heat flux parameters for each time step.

- The converged heat flux parameters are perturbed one at a time (in total N times) and the direct problem is solved from t^0 to t^{k+1} N times with perturbed parameters (where N is the number of thermocouples equal to the number of parameters). The parameters are perturbed from the beginning of the solution to the end.

Also in this case the sensitivity matrix is computed according to Eq. (4.17). But now the fundamental difference is that temperatures obtained perturbing heat fluxes are the result of this perturbation from the start of the calculation to the end and not only the result of perturbation in a single time step. Figure below can help to understand this difference:

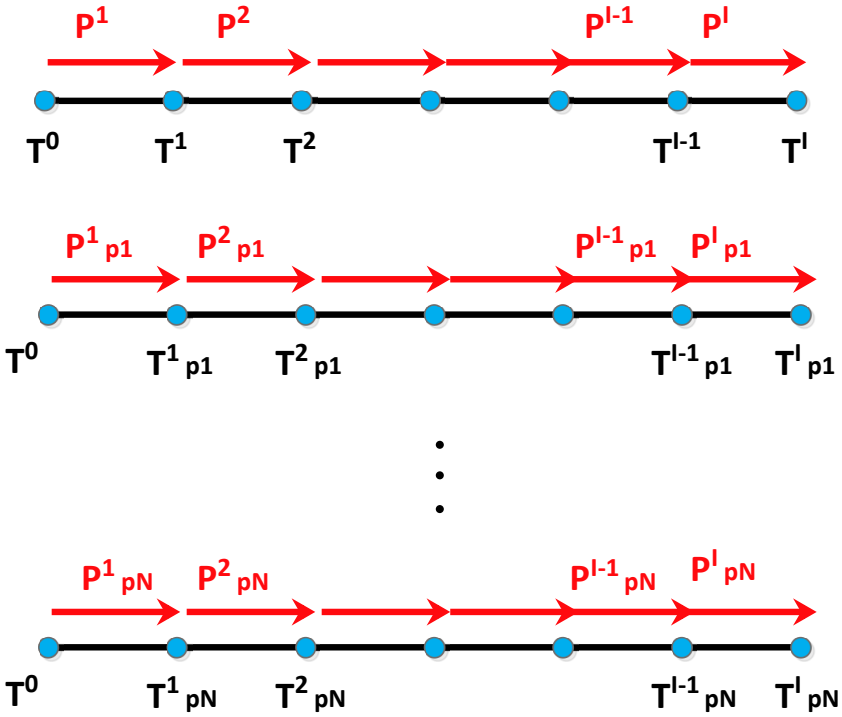


Figure 4.14: Computation of sensitivity matrix

Here :
T_{pj}^{k+1} are the computational temperatures in the whole domain resulting from solving the direct problem from t^0 to t^{k+1} perturbing the j-th heat flux parameters and keeping the others constant.
It has been shown in Section 4.3.1 that the Jacobi matrix is almost independent on heat fluxes and time step. So it can be calculated only one time out of the cycle.
Sensitivity matrix is almost independent on heat flux but it is dependent on the time step in which it is calculated. Figure 4.15 shows as an example (all the other diagonal elements behave in the same way) both diagonal elements of sensitivity matrix (computing using converged heat fluxes or constant heat fluxes) versus time. It can be noted that the

sensitivity matrix has small values at the beginning (in the first time segment it coincides with the Jacobi matrix), increases its values with a big slope in the first time segments and then continues increasing with a smaller slope.
Figure 4.16 shows the element of a row of the sensitivity matrix over time. In particular it is row 3. Here $J_S(3,2)$ and $J_S(3,4)$ are perfectly overlying and one can note that basically, only diagonal, upper diagonal and lower diagonal terms are non-zero. This means that a perturbation in heat flux (over the all experimental time) in position i, only affect temperatures in position $i-1$, i and $i+1$.
The sensitivity matrix is a measure of the ill-posedness of an inverse problem (the smaller it is, the more the inverse problem is ill-posed). Figure 4.15 shows that at the beginning the inverse problem is very ill-posed and then things improve as the time increases.

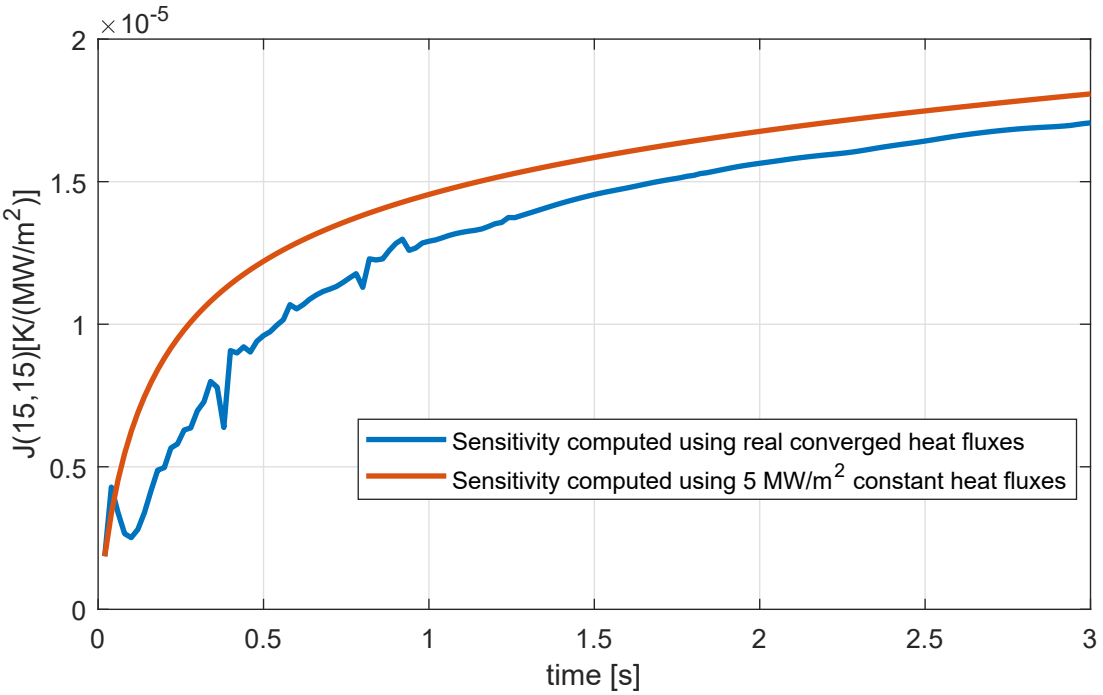


Figure 4.15: Sensitivity matrix over time.

The **Jacobi matrix** is used in:

- Newton Rapshon method (Section 4.3).
- Estimating errors on heat flux due to thermocouples precision errors (Section 6.1.1).

The **sensitivity matrix** is used in:

- Estimating errors on heat flux due to thermocouples accuracy errors (Section 6.1.2).
- Estimating errors on heat flux due to thermocouples locations errors (Section 6.1.3).

- Estimating errors on heat flux due to material properties errors (Section 6.1.4).
- Estimating errors on hot gas wall temperatures (Section 6.2)

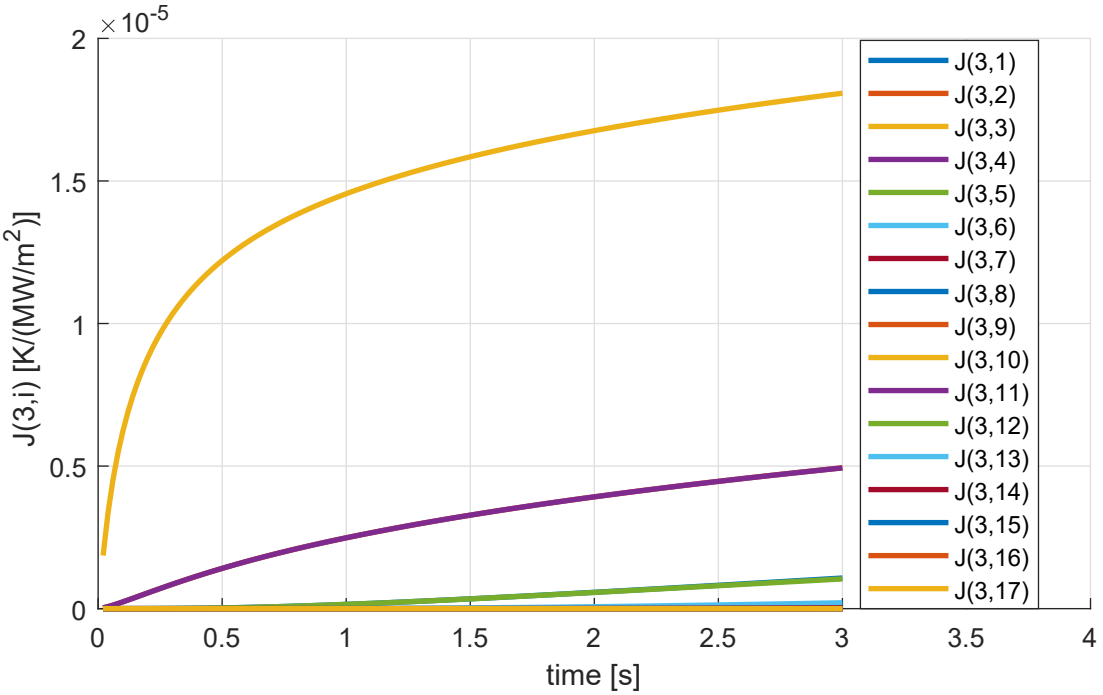


Figure 4.16: Elements of row 3 sensitivity matrix over time.

4.3.3 Comparison with conjugate gradient method

Figure 4.17 shows the number of iterations needed to reach the convergence for each time step. Conjugate gradient method and Newton Rapshon method are compared. One can note that the Newton Rapshon method is much faster.
A simulation using $\sigma = 0.3\ K$ in the stopping criteria definition lasts less than half the time using the Newton Rapshon method compared to the conjugate gradient method. Figure 4.17 refers to a test in which $\sigma = 0.3\ K$. The more the stopping criteria becomes restrictive, the larger advantage in terms of computational time Newton Rapshon method has compared to the conjugate gradient method (see Figure 4.17 and 4.18). So using the Newton Rapshon method it is possible to set more restrictive stopping criteria without too much time cost (which means matching the measured temperatures with higher accuracy). Since the smoothing cycle (Section 4.2.3) has reduced the precision errors, it is important to reduce the standard deviation when defining the stopping criteria in order to better exploit the smoothing cycle.

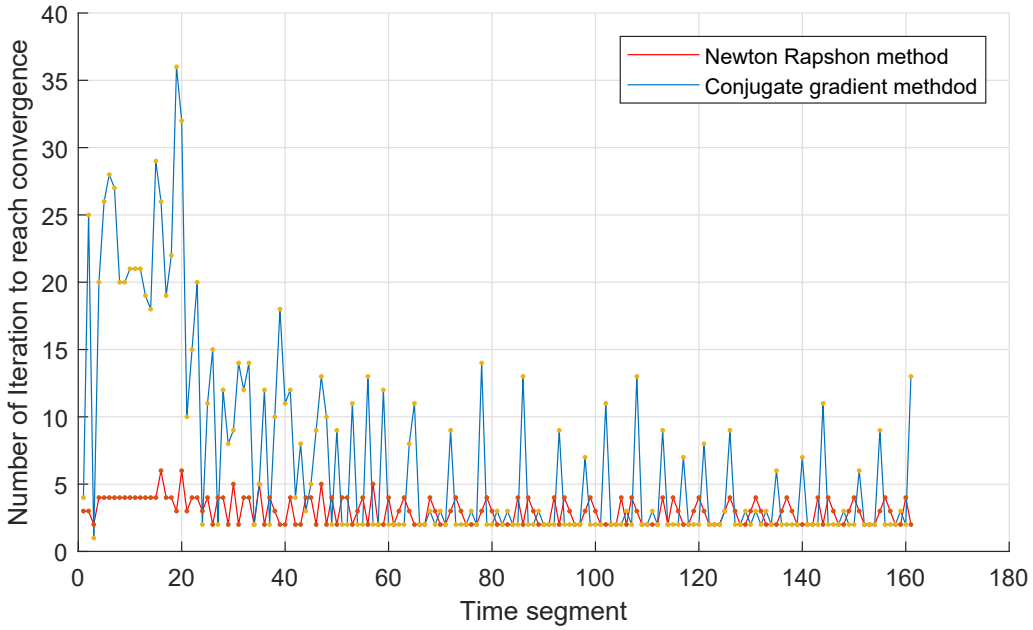


Figure 4.17: Optimization methods comparison, $\sigma = 0.3$ K.

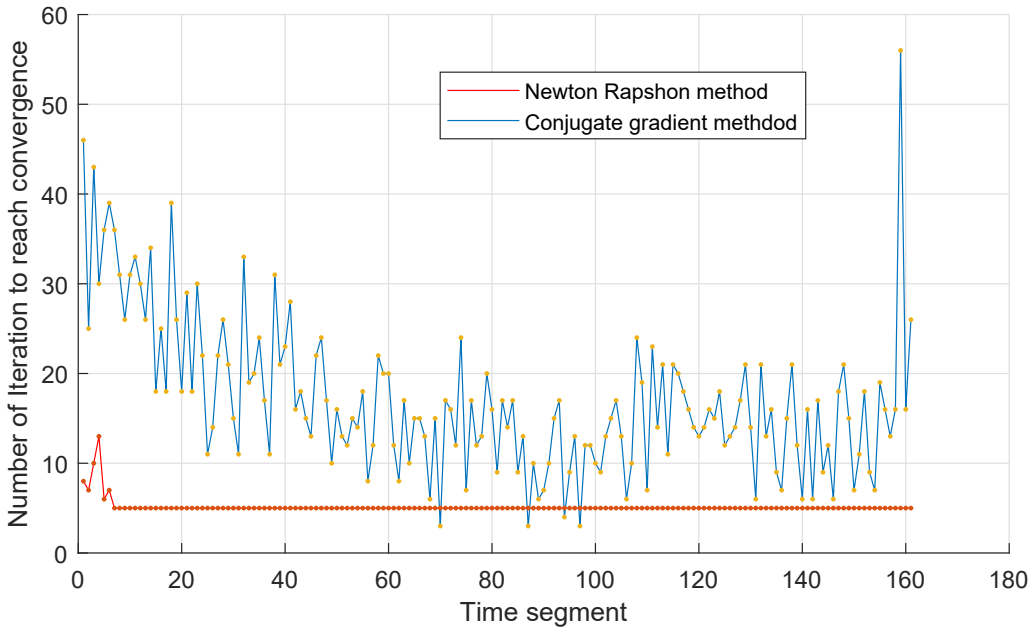


Figure 4.18: Optimization methods comparison $\sigma = 0.1$ K.

4.3.4 Initializing heat flux using linear interpolation

Since one hypotesis for heat flux oscillations over z is the effect of the igniter, it was possible to think that initializing the heat flux in each new time segment with the converged one

from previous segment, could lead to propagation of heat flux oscillations from previous time segments. Since the Newton Rapshon method is faster, it was able to initialize the heat flux in each new segment using a linear interpolation as in Figure 3.16. However, this initialization led to the same converged heat flux, indicating that the type of inizialization for each new time segment is not the cause of oscillations along z.

4.4 Post smoothing heat flux over z

In order to get rid of the oscillation over z, it has been tried to smooth the converged heat fluxes over z, after the inverse cycle and recalculate the temperatures using these new smoothed heat fluxes. The procedure can be summarized as follows:

- Solve the inverse problem.
- Smooth the converged heat flux over z.
- Recalculate the computational temperatures using these new smoothed heat fluxes.
- Evaluate the temperature differences between new computational temperatures and measured temperatures.

In particular, the smoothing process of heat flux over z, has been started 0.5 s after combustion is started, since at the beginning, oscillations over z could be physical, due to the effect of the igniter.
Results are depicted in Figures 4.19 to 4.21. The errors in temperatures are very big and they are not certainly comparable with the accuracy of the thermocouples. A possible source of this deviation could be an error in positioning of the thermocouples as shown in Section 6.1.3. One has also been tried to smooth less along z (Figure 4.22 and 4.23). Oscillations start to appear again and error in temperature is still too high.

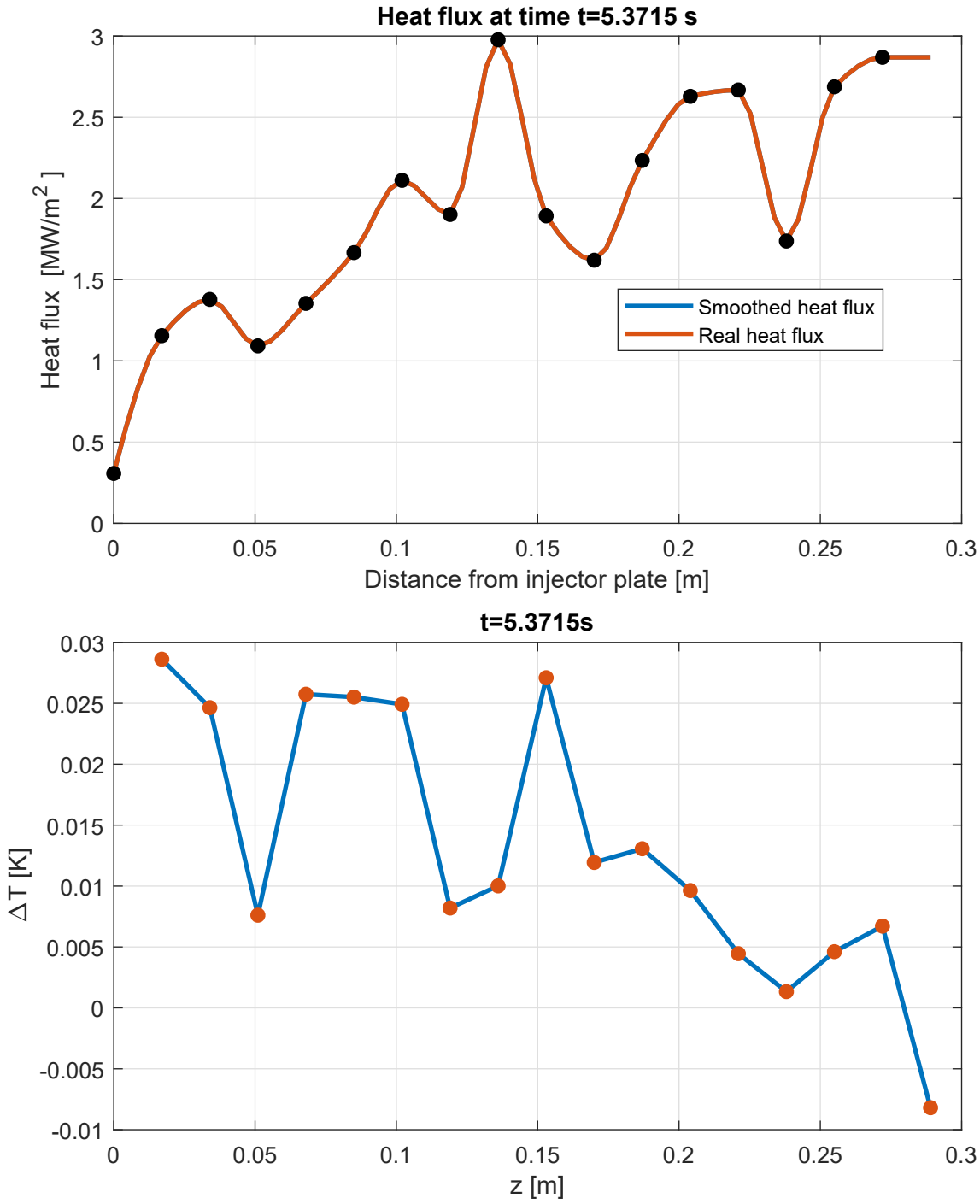


Figure 4.19: Converged heat flux and smoothed heat flux (up figure), temperature differences (down figure) t=5.3715 s.

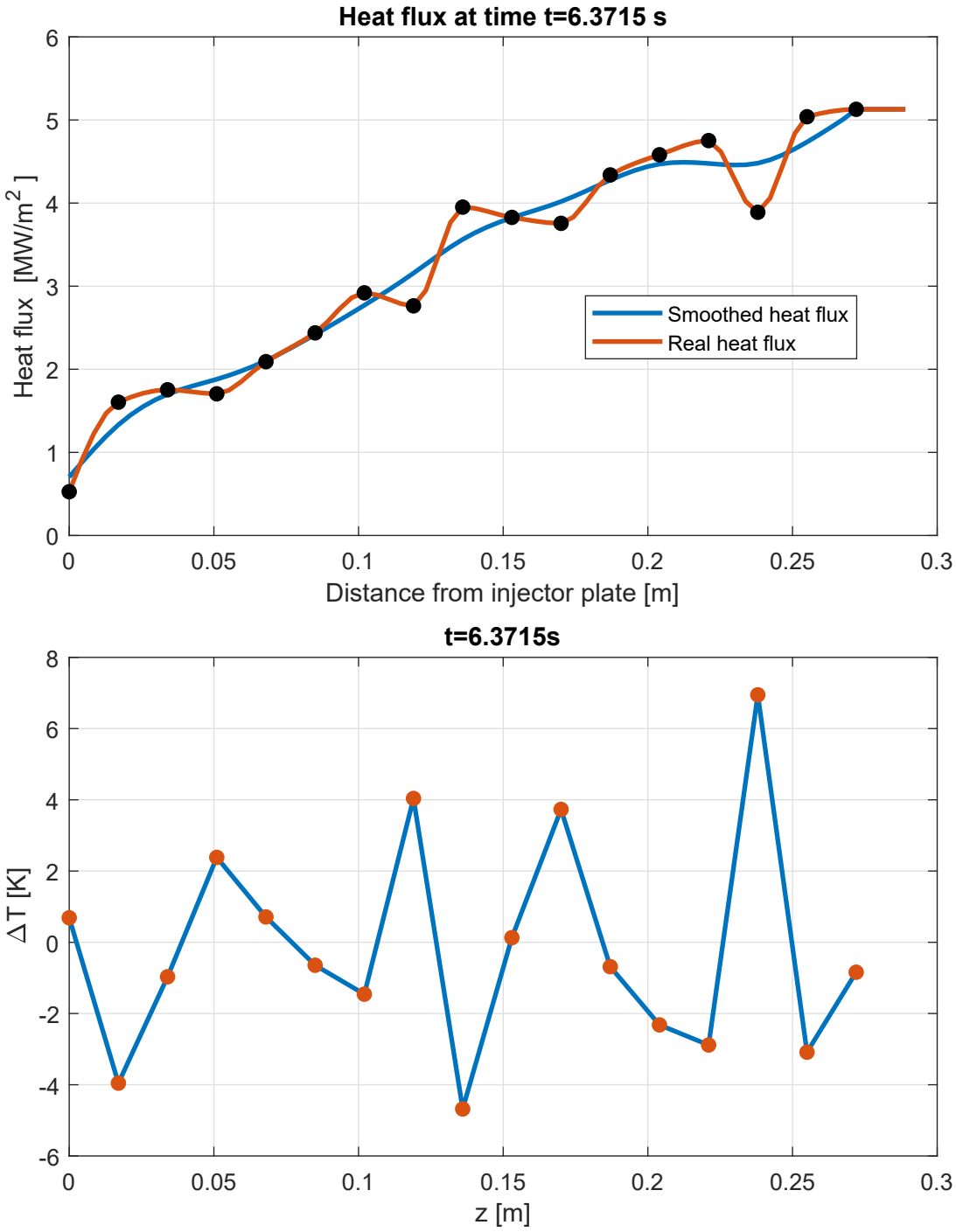


Figure 4.20: Converged heat flux and smoothed heat flux (up figure), temperature differences (down figure) t=6.3715 s.

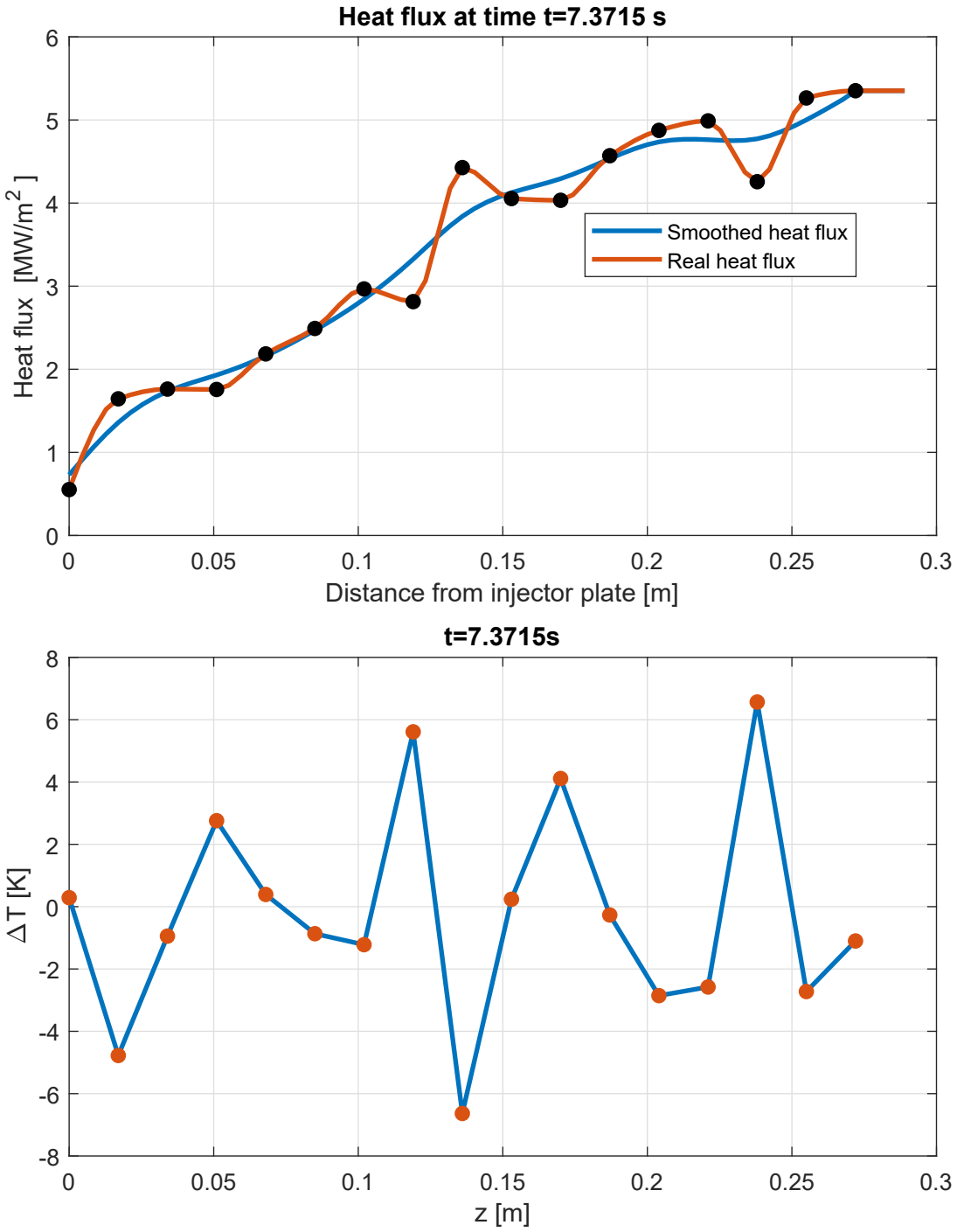


Figure 4.21: Converged heat flux and smoothed heat flux (up figure), temperature differences (down figure) t=7.3715 s.

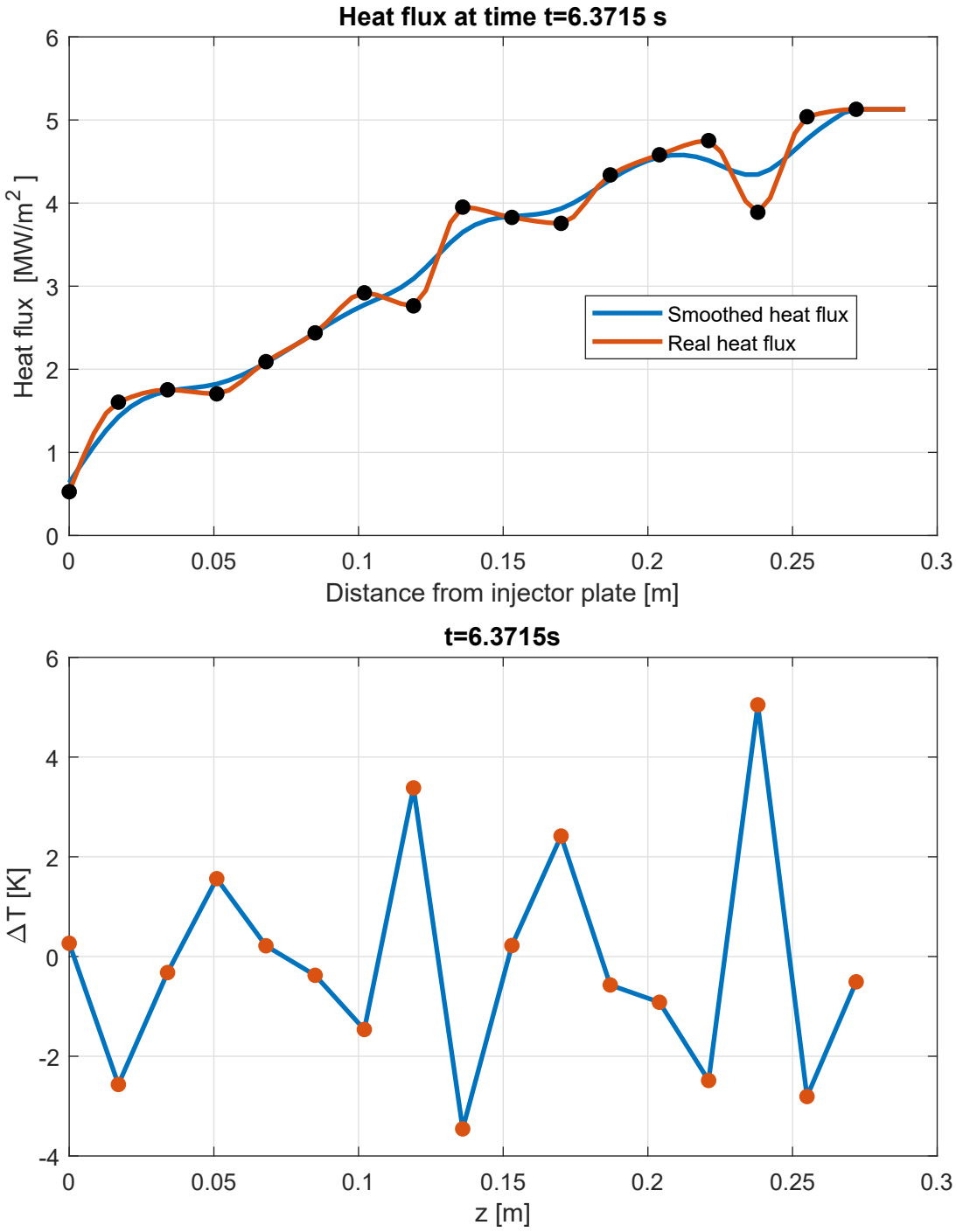


Figure 4.22: Converged heat flux and smoothed heat flux (up figure), temperature differences (down figure) smooting reduced t=6.3715 s.

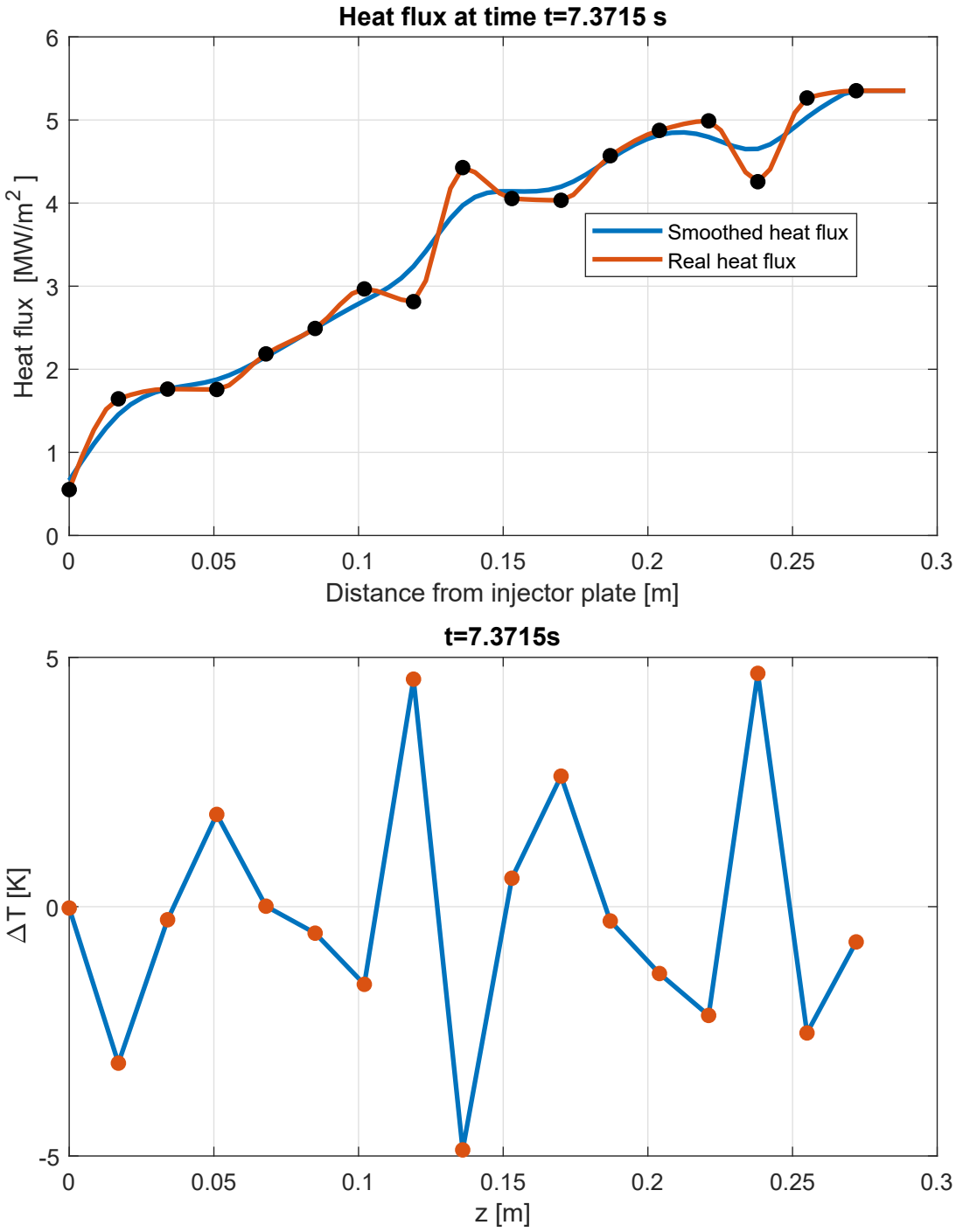


Figure 4.23: Converged heat flux and smoothed heat flux (up figure), temperature differences (down figure) smoothing reduced t=7.3715 s.

Chapter 5

Validation of RoqFITT

5.1 Inverse solution on self created experimental data

In order to identify if the oscillations are an artefact of the inverse code or due to experimental data, a validation of the inverse code has been performed. The major issue of the inverse transient heat method in this thesis is that boundary condition heat fluxes are not known and so it is diffucult to evaluate if the code is working as expected, that is if it is estimating with certain reasonable limits the boundary condition.
In order to validate RoqFITT, some 'computational experiments' with a known boundary condition have been performed.
The procedure can be schematized as follows:

- Assign a known boundary condition on the hot gas wall for each time step.
- Assign a starting condition temperature for the all domain.
- Solve the direct problem (now that boundary condition is known it is possible) for each time step.
- Evaluate from computational temperatures at each time, temperatures in thermocouples position.
- Perturb the extrapolated thermocouples temperatures with a standard deviation $\sigma = 0.15\ K$.
- Use this temperature in thermocouples locations to solve the inverse problem.
- Evaluate heat flux boundary conditions obtained solving inverse problem and check if they match the assigned boundary conditions used to solve the direct problem.

The procedure is also schematized in Figure 5.1

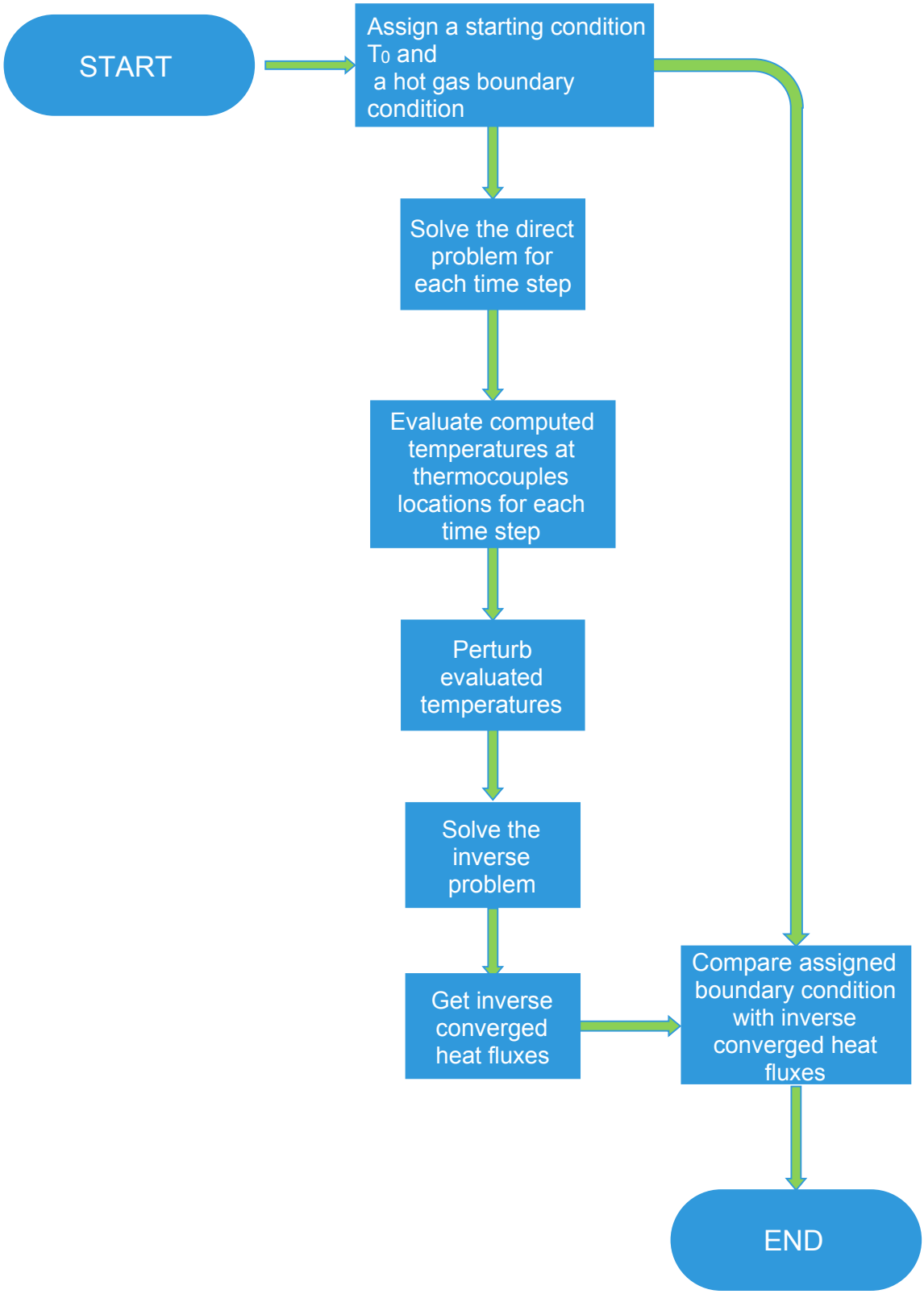


Figure 5.1: Algorithm for RoqFITT validation.

The heat fluxes (direct heat fluxes and ones obtained using inverse method) are depicted in Figure 5.2. An oscillation has been introduced in the first time instants (for about 0.5 s) and then this oscillation suddenly disappears. This has been done to see the effect of a suddenly disappearing oscillation on the inverse code. In fact as already Said in Section 3.4.11 one of the hypotesis made about oscillations over z is that they could be physical at the beginning of the experiment (because of the effect of the igniter), but these oscillations should disappear as experiment goes forward (at least when averaged in time).
Figure 5.3 shows some temperatures at 1 mm from hot gas wall resulting from the solution of the direct problem. The change in slope of temperatures at z=0.1870 m and z=0.2210 m is the result of the disappearing of the oscillations over z.
Figure 5.4 shows also the perturbed temperatures (that are the input of the inverse problem). One can note that the deviation is very small as in the case of real measurements. Figures 5.5 to 5.10 show the comparison between applied heat fluxes and obtained from the inverse method ones at different times. Note that they are matched quite good (especially in the parameters point that are the point in which inverse code optimize the parameters). This is a further proof that the inverse code works as expected and does not introduce oscillations when input information (material properties and thermocouples locations) are given precisely. The precision of thermocouples doesn't affect too much the estimation of heat fluxes. It is important to recall that also the errors from thermocouples accuracy are not big as explained in Section 6.1.2.

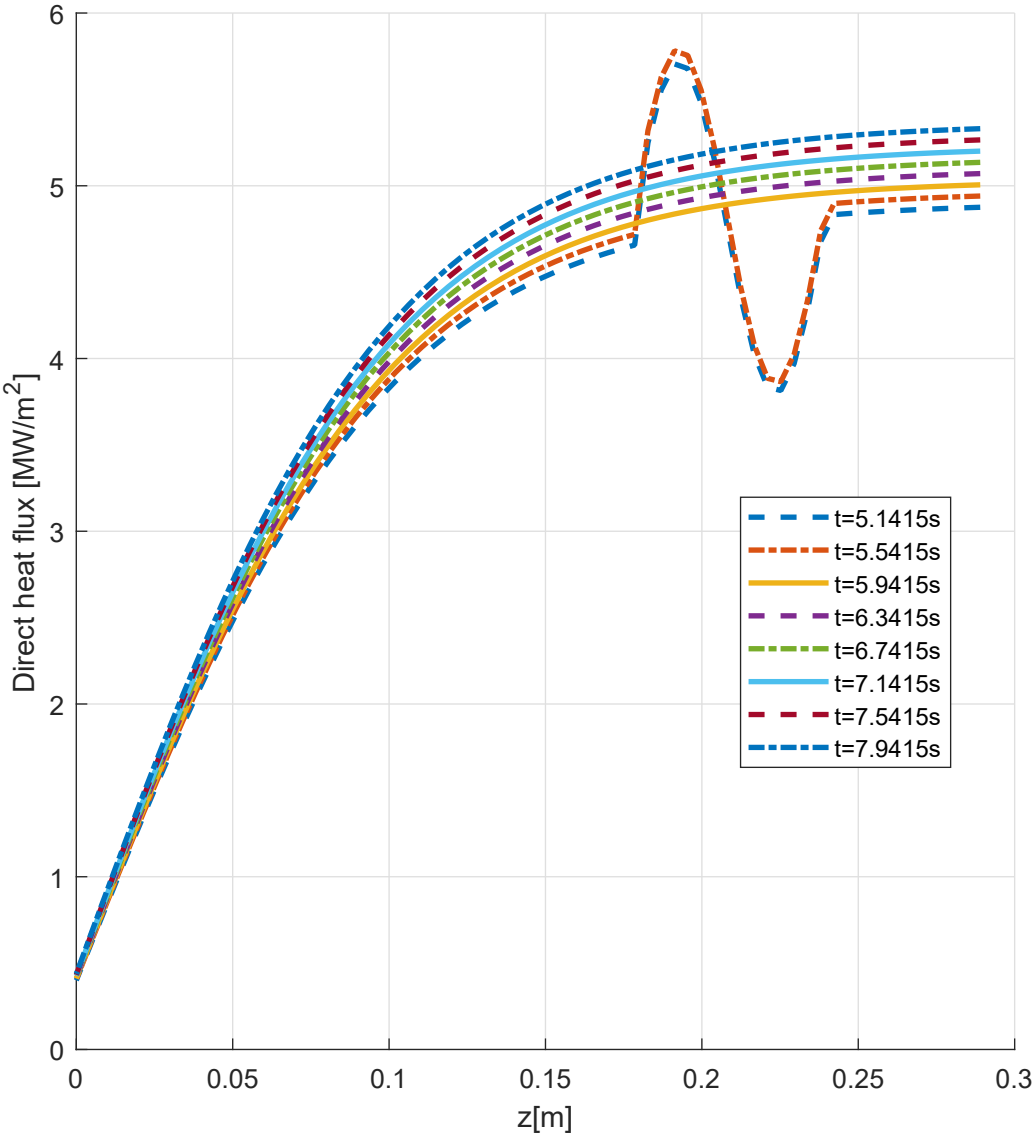


Figure 5.2: Known boundary hot gas wall heat fluxes (used to solve direct problem).

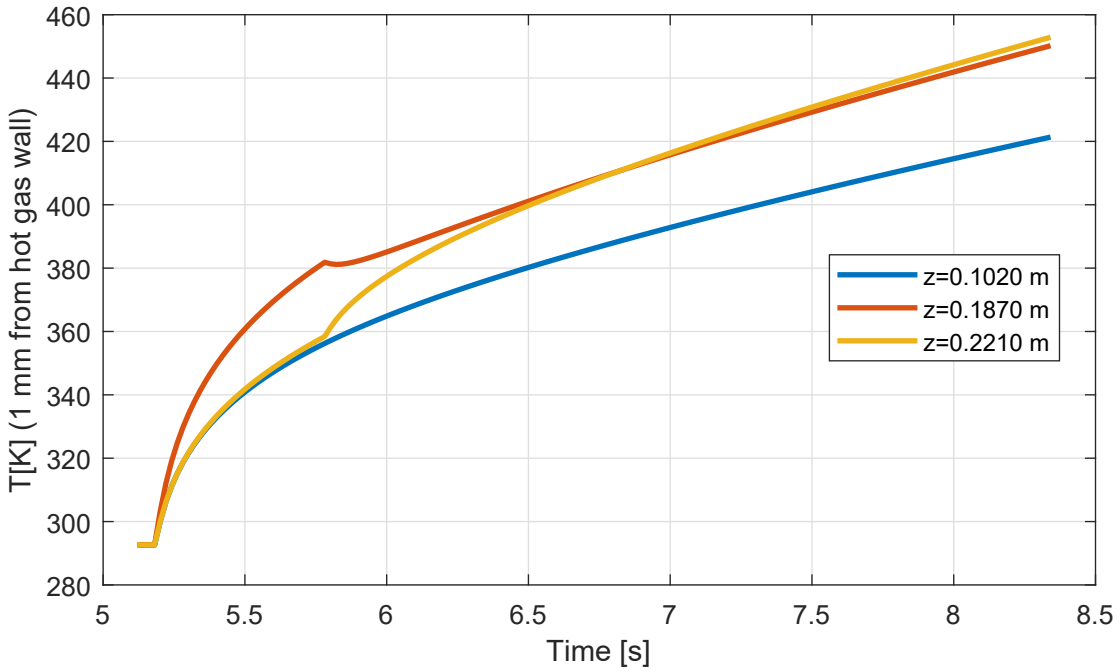


Figure 5.3: Temperatures resulting from the solution of the direct problem with known heat fluxes (1 mm from hot gas wall) vs time.

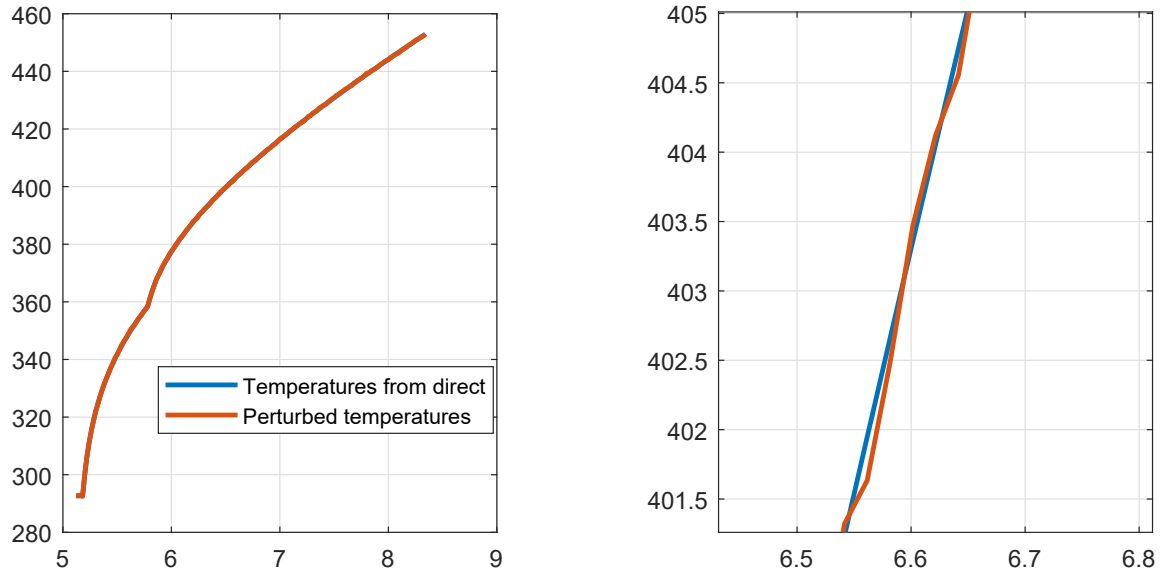


Figure 5.4: Temperatures from direct solution and perturbed ones for a fixed location (1 mm from hot gas wall and $z = 0.1870$ m). Right figure is a zoom.

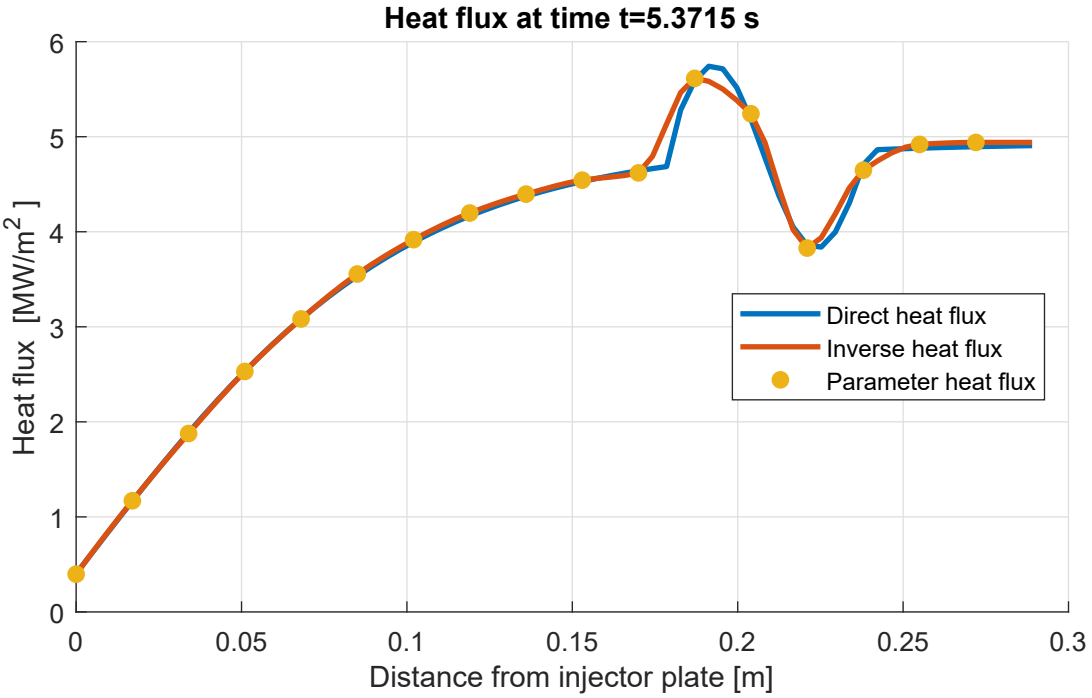


Figure 5.5: Known heat fluxes and obtained from inverse code heat fluxes along z ($t = 5.3715$ s).

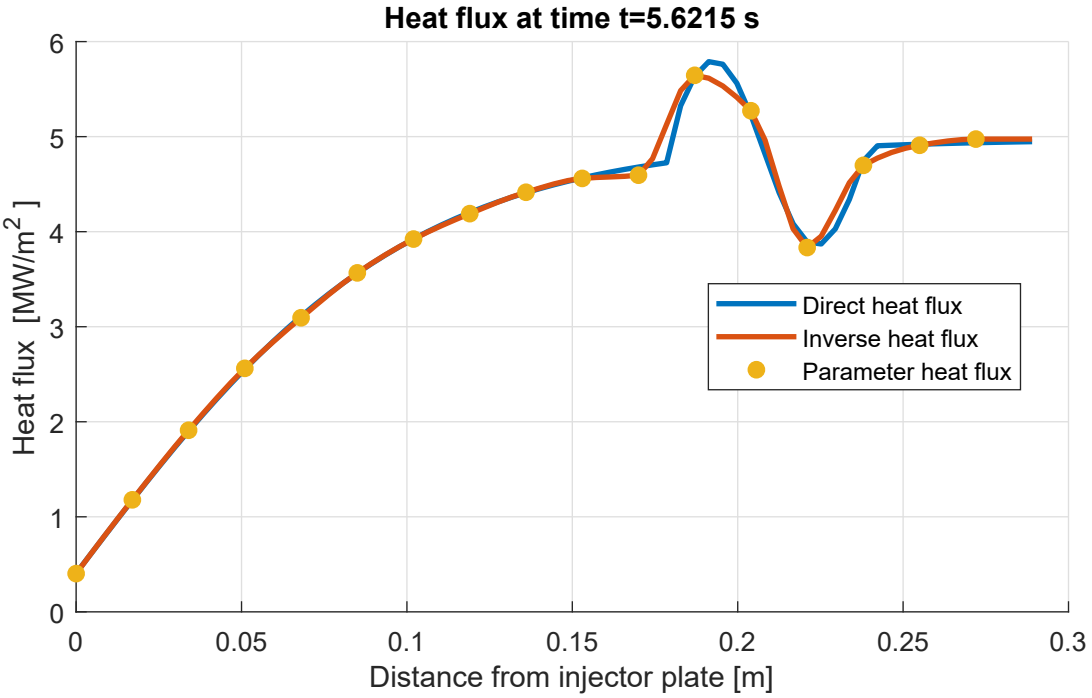


Figure 5.6: Known heat fluxes and obtained from inverse code heat fluxes along z ($t = 5.6215$ s).

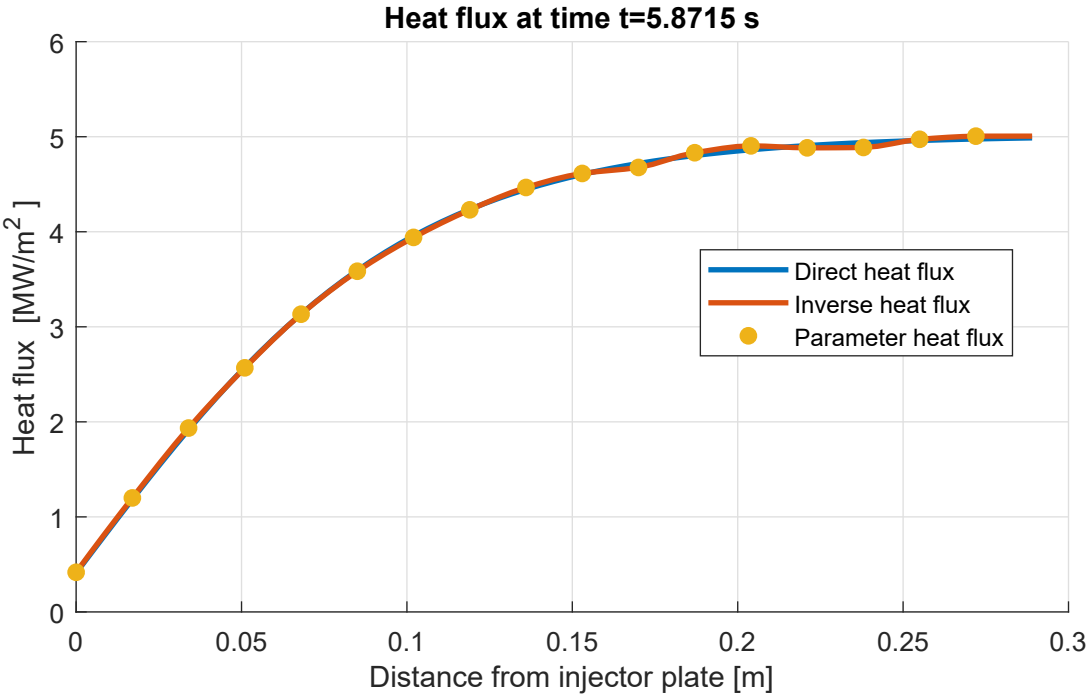


Figure 5.7: Known heat fluxes and obtained from inverse code heat fluxes along z ($t = 5.8715$ s).

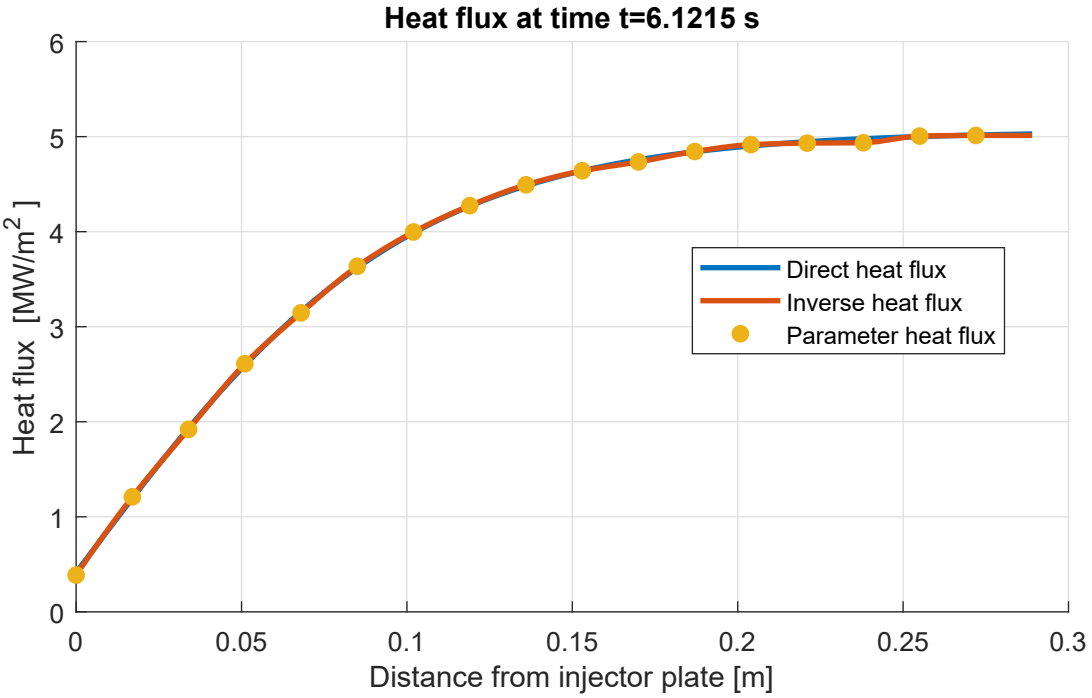


Figure 5.8: Known heat fluxes and obtained from inverse code heat fluxes along z ($t = 6.1215$ s).

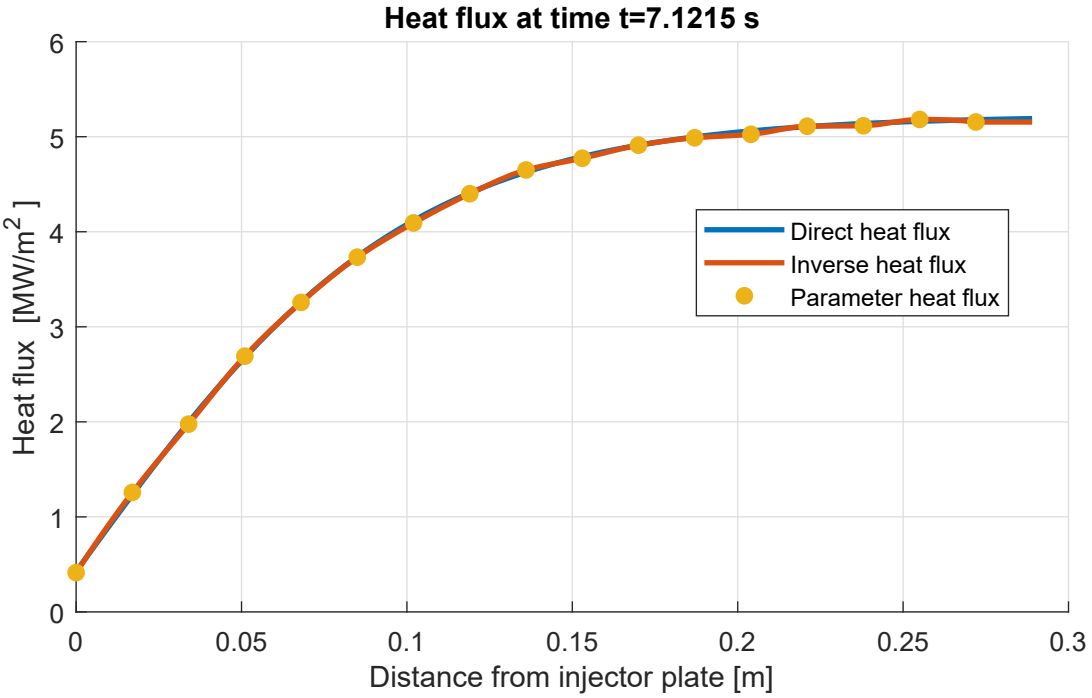


Figure 5.9: Known heat fluxes and obtained from inverse code heat fluxes along z ($t = 7.1215$ s).

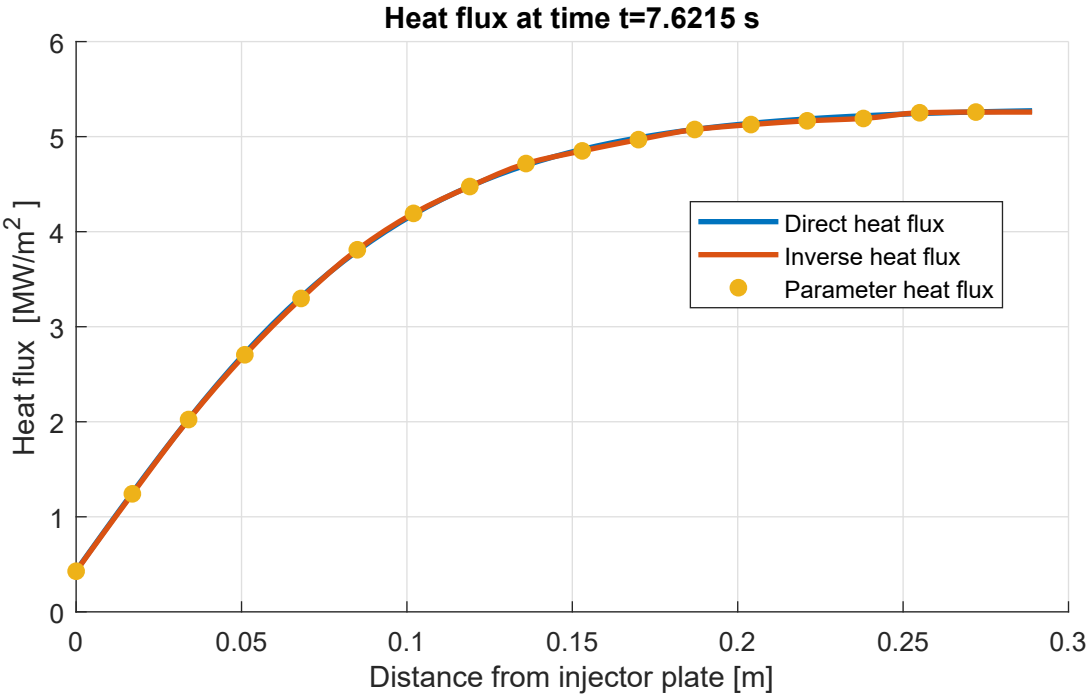


Figure 5.10: Known heat fluxes and obtained from inverse code heat fluxes along z ($t = 7.6215$ s).

5.2 Simplified inverse method

In order to evaluate the behavior of heat fluxes along z, without solving the entire inverse problem, a "simplified inverse method" has been implemented. It is essentially based on the discretization of energy equation (using a finite volume approach) in a simplified domain.
The approach is 2 dimensional (the equation is solved in the y-z plane[1]). The computational domain is shown in Figure 5.11.

- y=0 mm is the hot gas wall.
- z=0 mm is at the face plate and z=0.289 m is at the surface in contact with the nozzle.
- $\Delta y = 2\ mm$. It is 2 times the distance of thermocouples from hot gas wall.
- $\Delta z = 1.7\ cm$. It is the distance between thermocouples.
- The cell centers are located in the thermocouples locations.
- For y=2 mm it has been set an adiabatic boundary condition (this is equivalent to neglecting thermal conduction along y. Note that this is wrong since conduction along y exists).
- For z=0.289 m it has been set an adiabatic boundary condition (as in the case of the inverse code).
- For the first cell (i=1), in order to simulate the heat sink, the heat flux from thermal conduction coming in from cell i=2 is supposed to be equal to the heat flux coming out towards the heat sink.
- Conduction along z is a parameter analysis. It has been taken into account in one case and neglected in the other case.
- Heat flux hot gas wall (q_1,q_2,... q_{17}) are the unkown boundary condition of the analysis.

[1]Note that since 2D approach means $\frac{d}{dx} = 0$, this is equivalent to neglecting heat conduction along x.

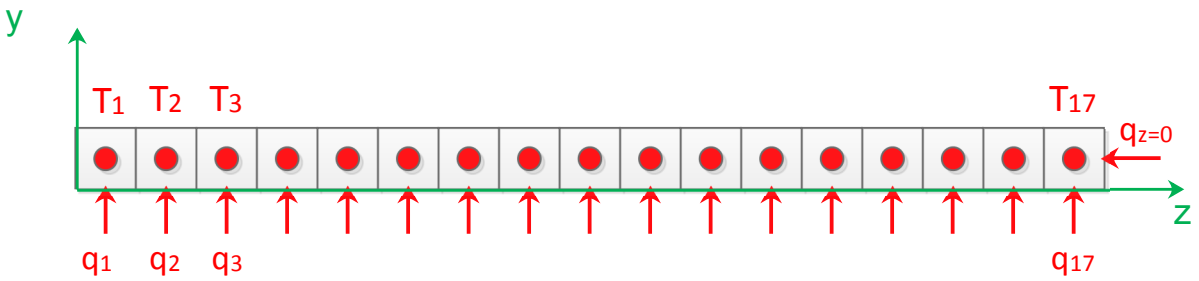


Figure 5.11: Simplified inverse method domain.

5.2.1 Equation discretized taking into account thermal condution along z

To understand how the equations have been discretized it is helpful to look Figure 5.12

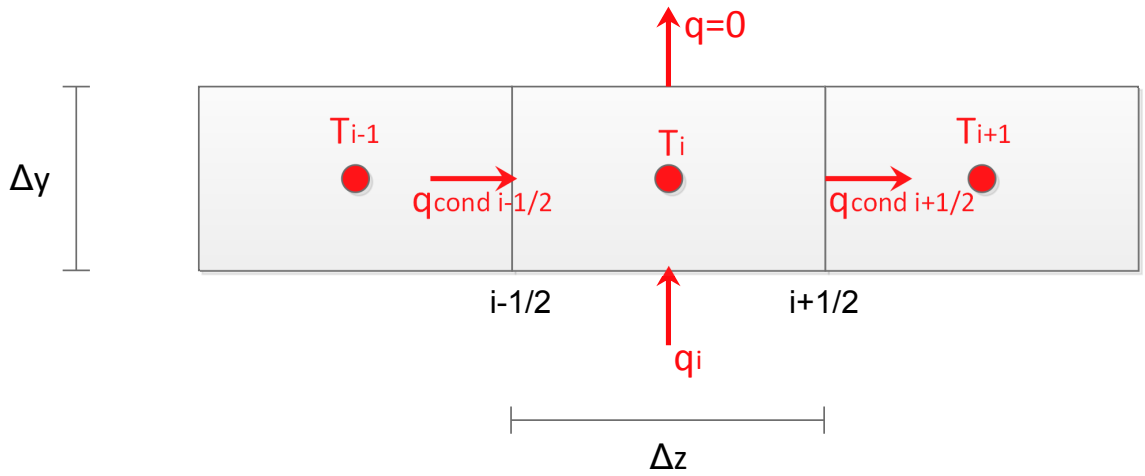


Figure 5.12: Simplified inverse method equation discretization.

Heat flux equation 2D discretized using a finite volume approach for the i-th cell control volume establish that:

$$q_i^{k+1}\Delta z + q_{cond\ i-1/2}^{k+1}\Delta y - q_{cond\ i+1/2}^{k+1}\Delta y = \rho c_p \frac{(T_i^{k+1} - T_i^k)}{\Delta t}\Delta y \Delta z \tag{5.1}$$

where:

- q_i^{k+1} is the hot gas heat flux wall at location i at time $k+1$ (supposed to be constant for time step from k to $k+1$).
- $q_{cond\ i-1/2}^{k+1}$ and $q_{cond\ i+1/2}^{k+1}$ are the thermal conduction heat fluxes along z at cell surface $i-1/2$ and $i+1/2$.

- T_i^{k+1} and T_i^k are the cell centers temperatures at time instants $k+1$ and k (in this case this temperature coincides with thermocouples temperatures).

Fourier law is valid (Eq. (3.4)) and the first derivatives of temperatures at cell surface $i-1/2$ and $i+1/2$ can be discretized using a centered in space scheme. Using an implicit method in time one obtains:

$$q_{cond-i-1/2} = -\lambda \frac{T_i^{k+1} - T_{i-1}^{k+1}}{\Delta z} \tag{5.2}$$

$$q_{cond-i+1/2} = -\lambda \frac{T_{i+1}^{k+1} - T_i^{k+1}}{\Delta z} \tag{5.3}$$

Substituting Eq. (5.2) and Eq. (5.3) into Eq. (5.1) one obtains:

$$q_i^{k+1} = \Delta y \lambda \left[\frac{1}{\alpha} \frac{T_i^{k+1} - T_i^k}{\Delta t} - \frac{T_{i-1}^{k+1} - 2T_i^{k+1} + T_{i+1}^{k+1}}{\Delta z^2} \right] \tag{5.4}$$

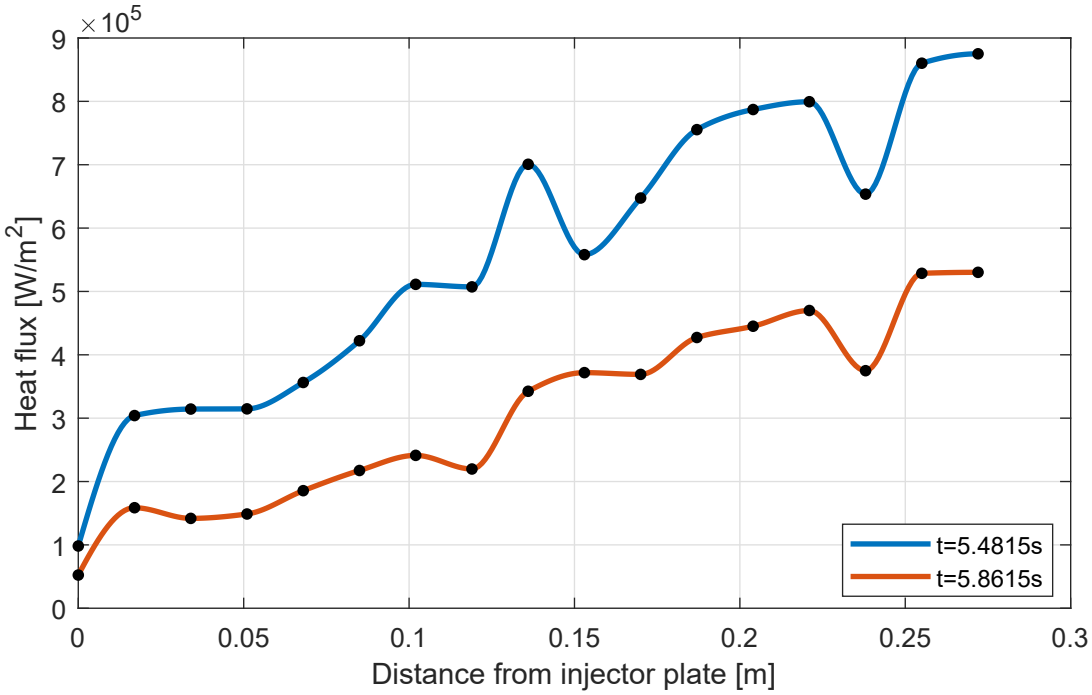
Eq. (5.4) is valid for internal cells. For boundary cells it has to be slightly modified in order to take into account the boundary conditions along z.
For the first cell, $i = 1$, it is valid:

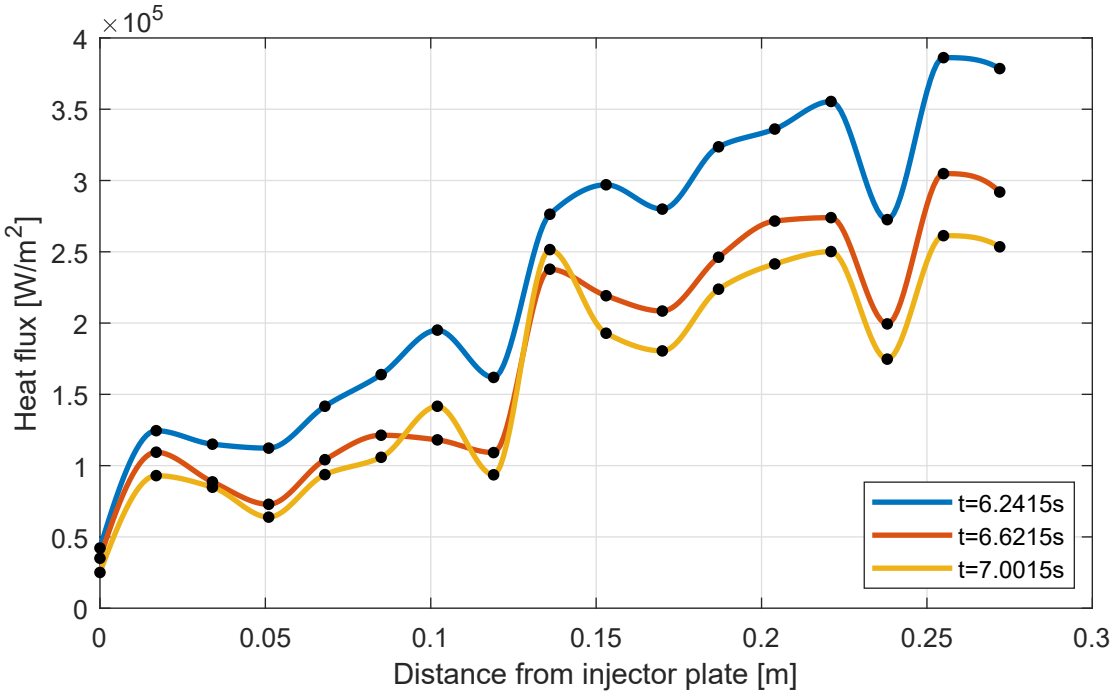
$$q_1^{k+1} = \Delta y \lambda \frac{1}{\alpha} \frac{T_1^{k+1} - T_1^k}{\Delta t} \tag{5.5}$$

For the last cell, $i = 17$, it is valid:

$$q_1^{k+1} = \Delta y \lambda \left[\frac{1}{\alpha} \frac{T_{17}^{k+1} - T_1 7^k}{\Delta t} + \frac{T_{17}^{k+1} - T_{16}^{k+1}}{\Delta z^2} \right] \tag{5.6}$$

Results from this analysis are depicted in Figure 5.13.
The absolute values obtained are no significant since thermal conduction in two direction is neglected. But it is interesting to look at the shape of these profiles. Also in this case, where no optimization inverse method has been applied, but only an approximative calculation of heat flux from measured temperatures, oscillations over z exist also after the shutdown of the igniter so they are not introduced by the optimization method.


×10^5
9
8
7
6
5
4
3
2
1
0
Heat flux [W/m^2]
0
0.05
0.1
0.15
0.2
0.25
0.3
Distance from injector plate [m]
t=5.4815s
t=5.8615s


×10^5
4
3.5
3
2.5
2
1.5
1
0.5
0
Heat flux [W/m^2]
0
0.05
0.1
0.15
0.2
0.25
0.3
Distance from injector plate [m]
t=6.2415s
t=6.6215s
t=7.0015s

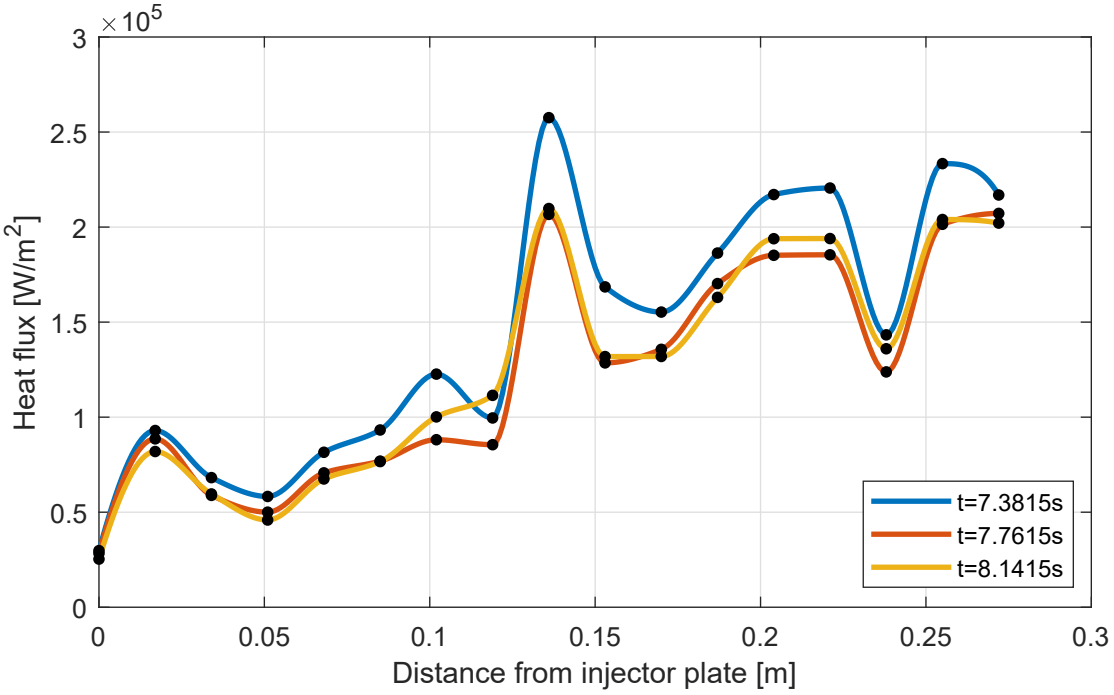


Figure 5.13: Approximated heat flux considering thermal conduction along z.

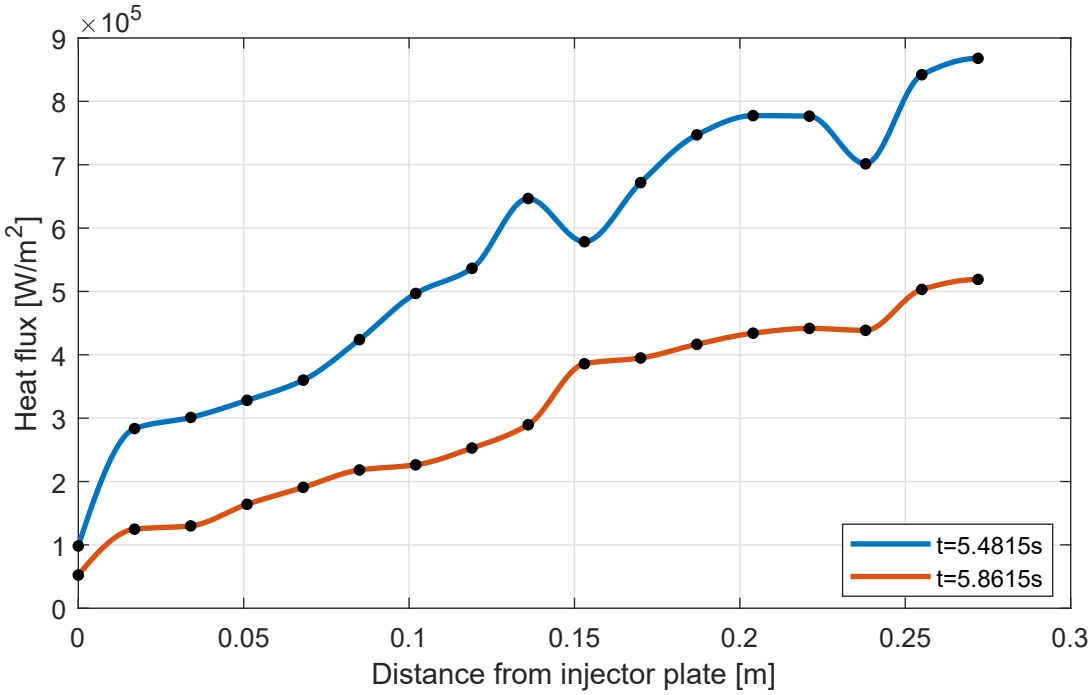
5.2.2 Equation discretized neglecting thermal condution over z

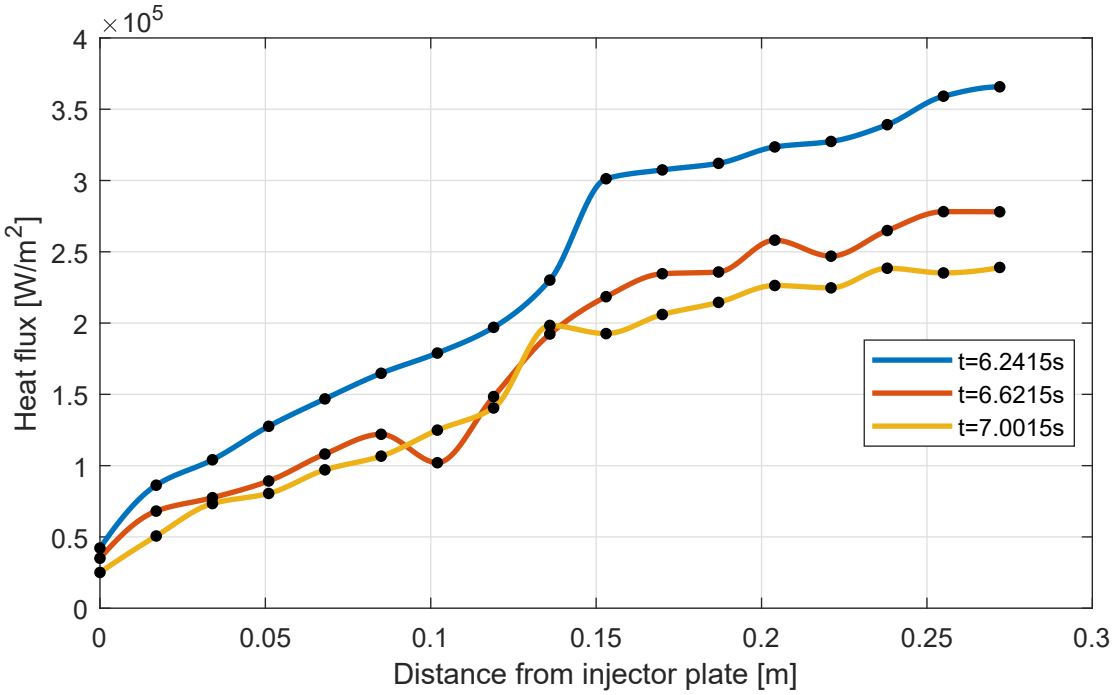
Neglecting thermal conduction along z, the equations are discretized as follows:

$$q_i^{k+1} = \Delta y \lambda \frac{1}{\alpha} \frac{T_i^{k+1} - T_i^k}{\Delta t} \tag{5.7}$$

This espression is valid for all the cells. Note that it is equivalent to setting an adiabatic boundary condition on all the cell surfaces except for the one in contact with the hot gas wall. Results from this analysis are depicted in Figure 5.14.

Also in this case, as previous, the absolute values for the heat flux are no significant. Some oscillations over z seem to disappear at certain time instants (oscillation between 0.1 and 0.15 m is still present). But note that, neglecting also thermal conduction along z, one is going far from the physics of the problem. So although the oscillation at z=0.238 m seems to disappear at the final instants of the experiment, it is an error to state that oscillation don't exist in the physics and they are introduced by the inverse method.


×10^5
Heat flux [W/m^2]
Distance from injector plate [m]
t=5.4815s
t=5.8615s


×10^5
Heat flux [W/m^2]
Distance from injector plate [m]
t=6.2415s
t=6.6215s
t=7.0015s

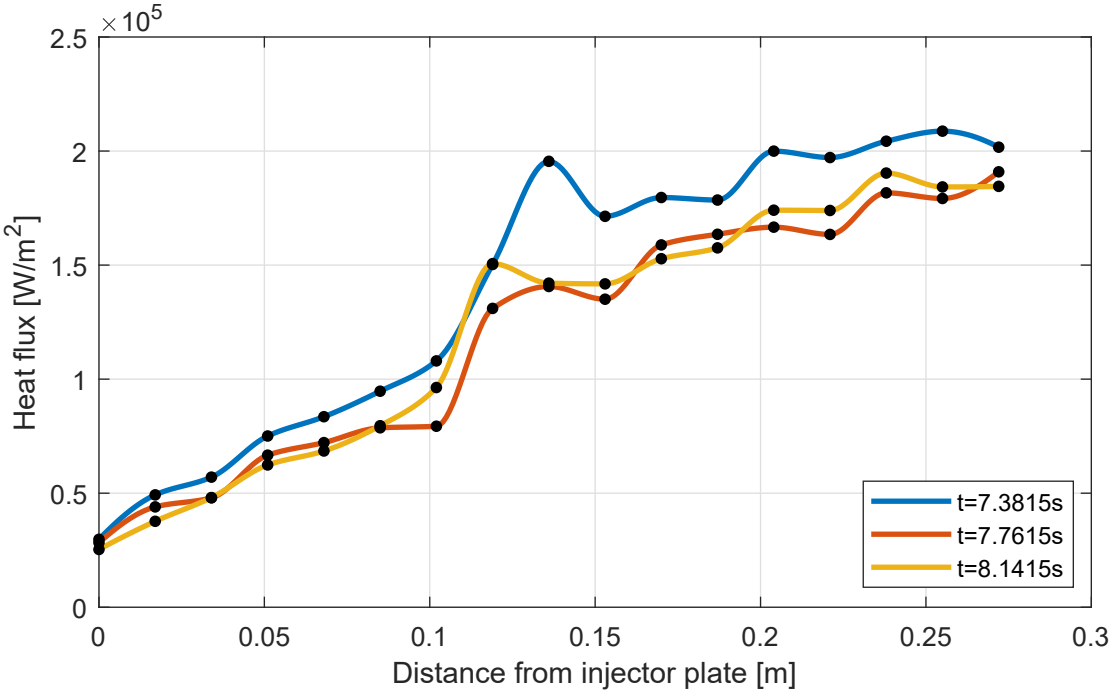


Figure 5.14: Approximated heat flux neglecting thermal conduction along z.

5.3 Validation of boundary conditions (Effect of heat sink length)

Heat sink length is not well defined, since it is only a modeling of the copper block around the injector. Because of this, the length of the heat sink is a parameter analysis and it is important to see how the inverse heat flux changes with respect to a change in the heat sink length.

It is important to underline that a heat sink length equal to 0 m (i.e. the absence of the heat sink) doesn't allow to match measured temperatures with computed ones, since the physical situation has a heat sink and so it has no sense to eliminate it in the computation.

Figures 5.15 to 5.17 show that only the first heat flux parameter (the one attached to the faceplate) is affected by a variation of the heat sink length.

Figure 5.18 shows the heat flux at z=0 for different heat sink lengths and for different times. It is possible to note that a difference in heat flux due to different length starts to exit at time larger than 5.7415 s. It is also possible to note that a large difference exists between length=0.086 m and other lengths. The difference in heat flux between this length and other lengths is about 0.2 MW. Heat flux vs length reaches an asymptotic value after length equal to 0.258 m. For length=0.086 m the heat flux decreases its value

as time increases (this could be a signal that also in this case, as in the case of the absence of the heat sink, one is simulating something not responding to the reality).

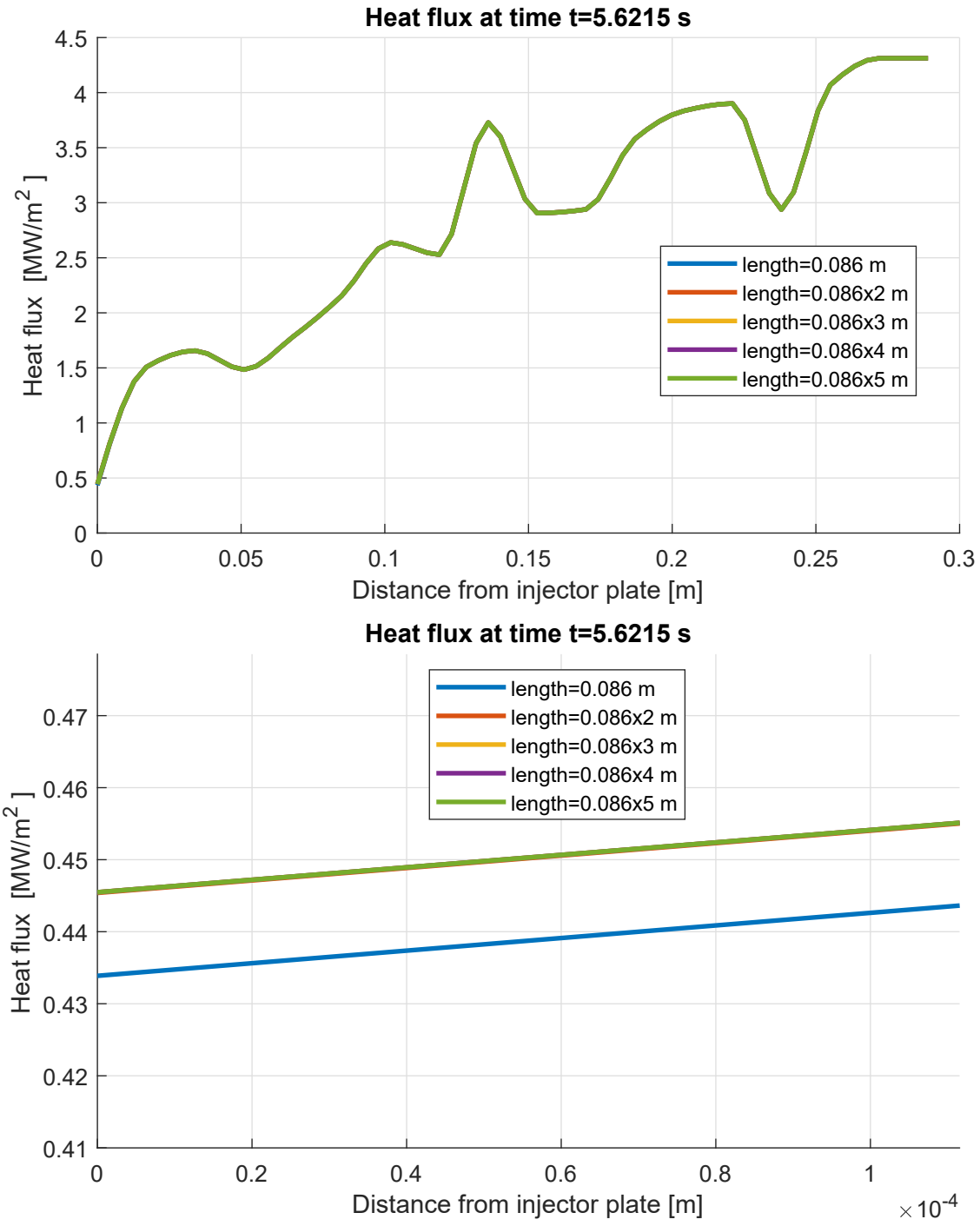


Figure 5.15: Heat flux over z for different length of heat sink $t = 5.6215$ s. Bottom figure is a zoom.

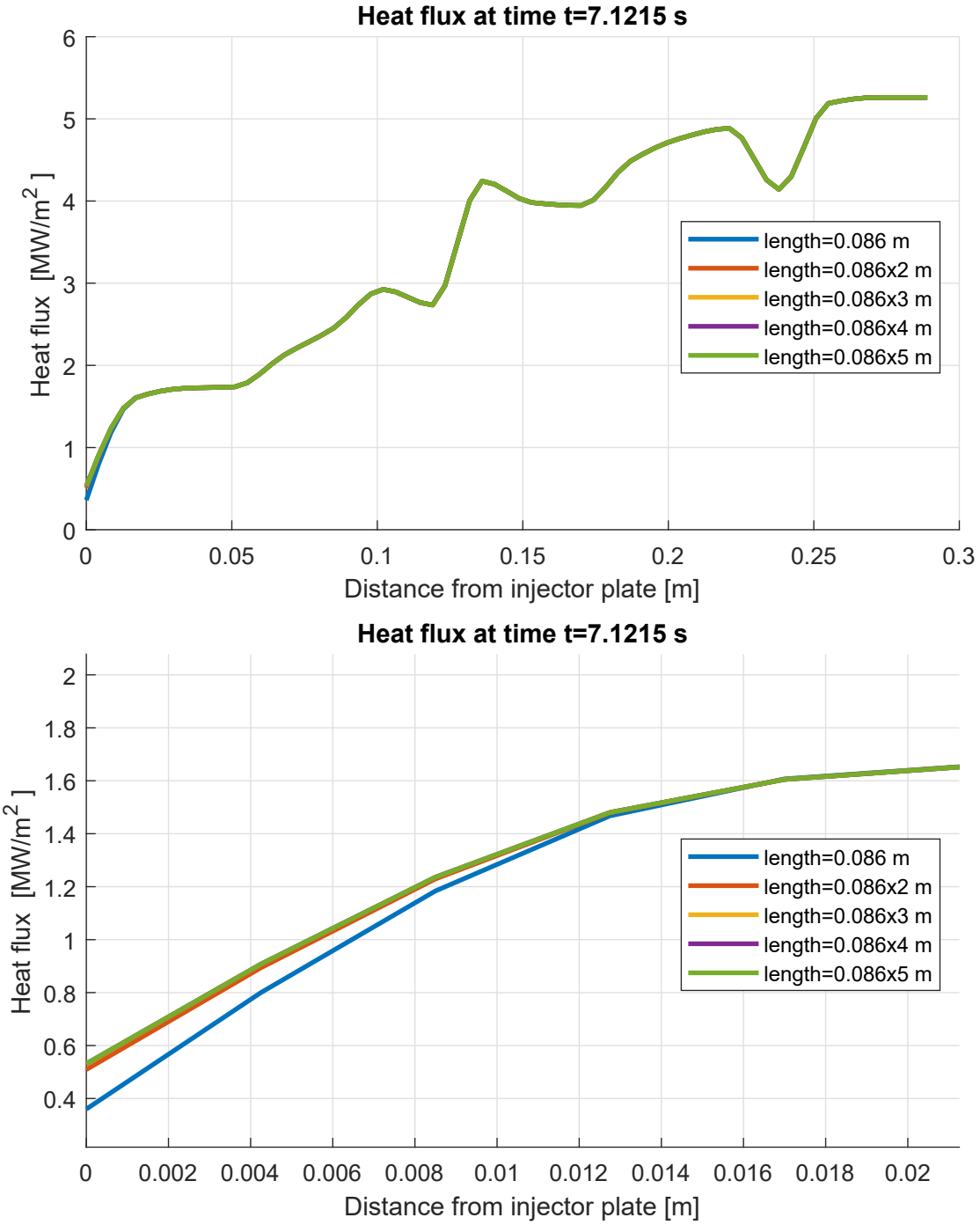


Figure 5.16: Heat flux over z for different length of heat sink $t = 7.1215$ s. Bottom figure is a zoom.

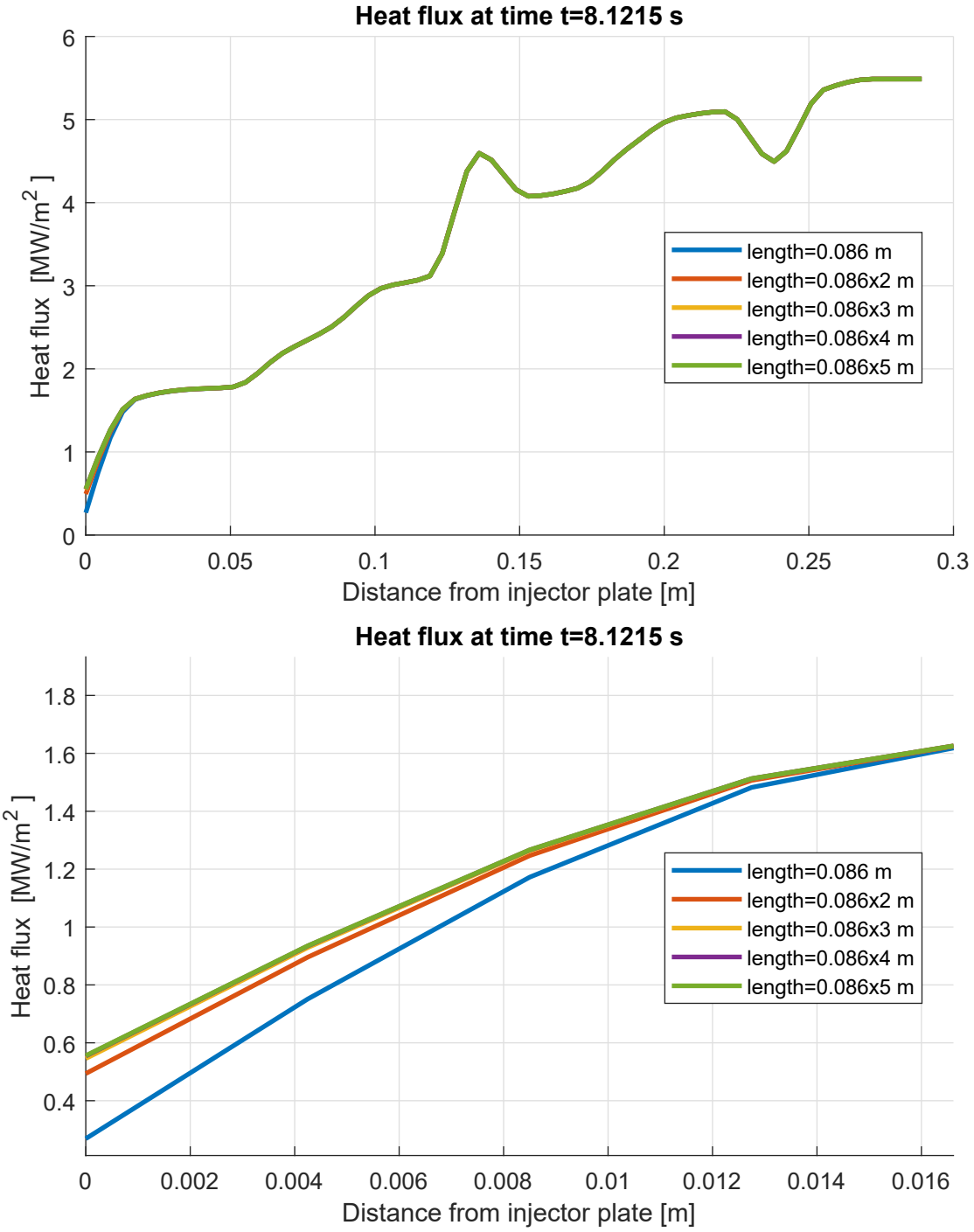


Figure 5.17: Heat flux over z for different length of heat sink $t = 8.1215$ s. Bottom figure is a zoom.

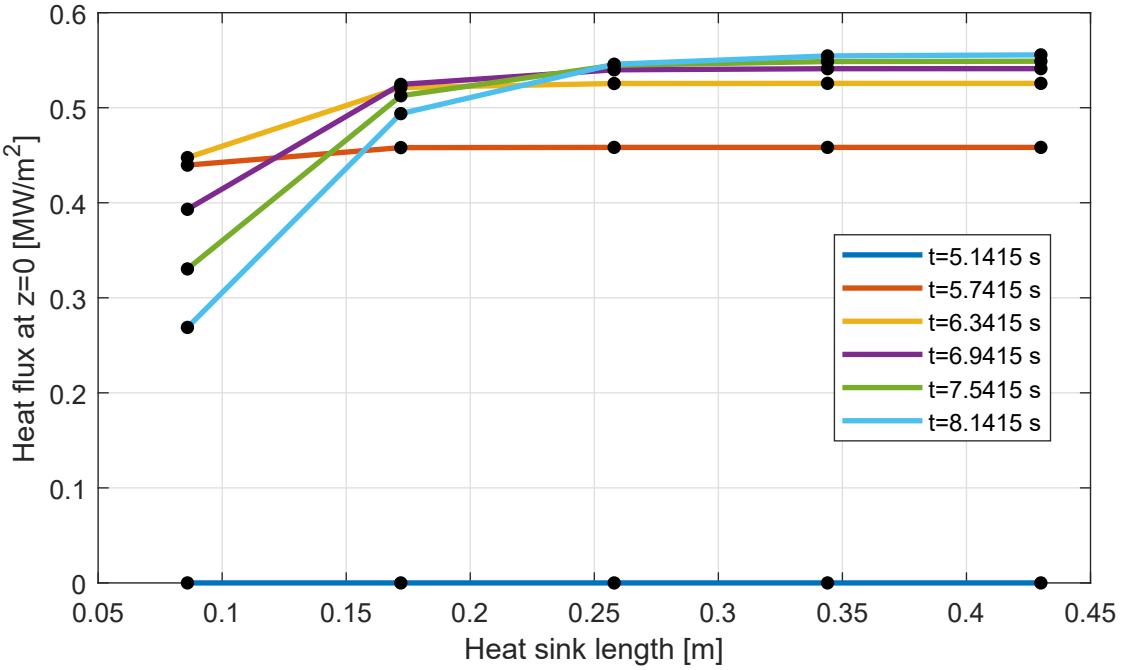


Figure 5.18: Heat flux at z=0 vs heat sink length for different times.

Chapter 6

Errors estimation

In this chapter errors on heat flux and hot gas temperature will be estimated.
In Section 6.1 errors on heat flux will be estimated. In particular:

- In Section 6.1.1 errors on heat flux due to thermocouples precision will be estimated.
- In Section 6.1.2 errors on heat flux due to thermocouples accuracy will be estimated.
- In Section 6.1.3 errors on heat flux due to wrong positioning of thermocouples will be estimated.
- In Section 6.1.4 errors on heat flux due to wrong input material properties will be estimated.
- In Section 6.1.5 the total amount of errors on heat flux due to the four sources listed before will be estimated.

Once, errors on heat flux are known, is possible, through a sensitivity analysis to estimate errors on temperatures in the domain (in particular on the hot gas wall temperatures). In fact, basically, temperature in the domain are a function of the heat flux boundariy conditions. The estimation of errors on hot gas wall temperatures is presented in Section 6.2.

6.1 Errors on heat flux estimation

In this chapter the procedure used to estimate the errors on heat fluxes is shown. Then some plots of the estimated errors are presented. Since the raw data, on which the calculation of heat flux is based are the **thermocouples measurements** it is simple to

understand that the main source of error on the heat flux will be error on thermocouples measurements. Talking about measurement errors one has to distinguish between precision and accuracy.

- **accuracy**: in the fields of science and engineering, the accuracy of a measure is the degree of closeness of the measure to a quantity to that quantity's "true value[1]" [21].
- **precision**: the precision of a measurement system is related to reproducibility and repeatability, is the degree to which repeated measurements under unchanged conditions show the same results [21].

Figure 6.1 shows in a more intuitive way what is precision and what is accuracy. **Precision** is essentially a measure of the deviation from the middle value that the instruments measures if the measurements are repeated many times. **Accuracy** is essentially the difference between this middle value and the real value. Since the all experiments, here, are based on transient temperatures measurements inside a combustion chamber wall it is impossible to exactly recreate the same conditions. To evaluate the precision of thermocouples one has to refer to standard deviation which is obtained on a moving average instead of an average of the same measure repeated many times (see Section 4.2).

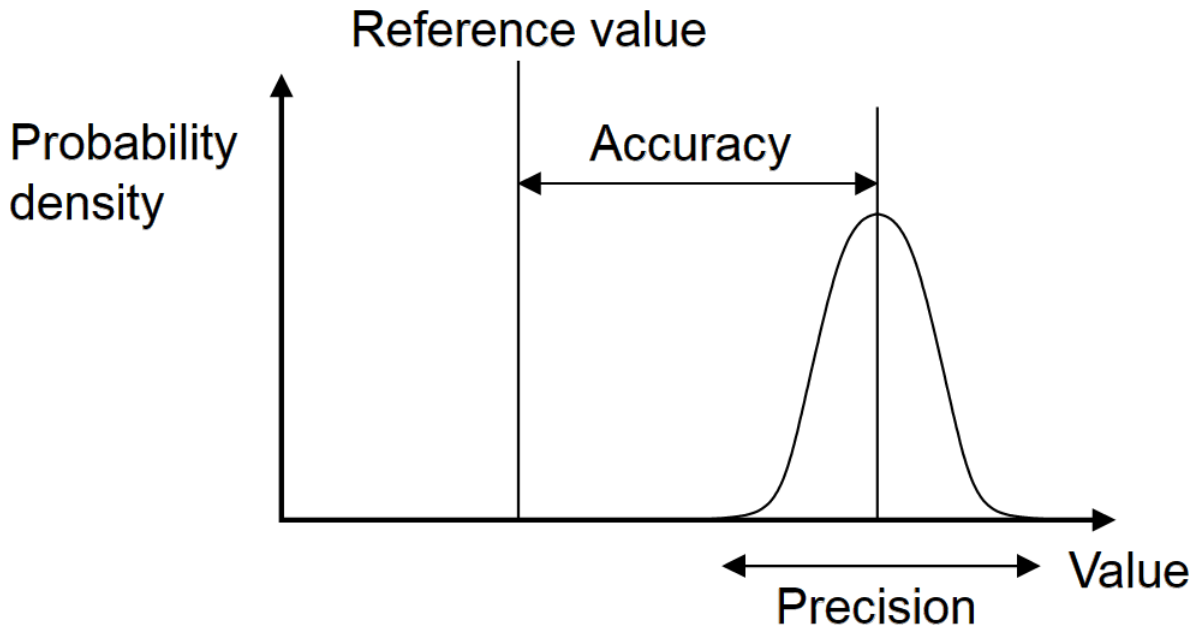


Figure 6.1: Precision and accuracy.

Other sources of errors are **material properties** values and **thermocouples locations** that if not known exactly introduce errors in evaluating heat fluxes.

[1]As a matter of fact the true value doesn't exist since no instrument is perfect. What we mean with true value is the measure of a higher order quality instrument with respect to the considered one.

6.1.1 Precision errors

In this Section, only random errors will be treated and other sorces of errors are supposed to be absent. In Section 4.2.3 errors on the smoothed values of temperatures have been estimated (Eq. (4.11)). Before looking into how random errors in temperatures propagate in heat fluxes, one has to note that computational temperatures don't match perfectly the smoothed ones. The difference between calculated temperatures and smoothed temperatures is defined by the stopping criteria.
When solving the inverse problem in time segment k^{th} which goes from t^{k-1} to t^k, the unkown of each direct problem that is solved at each iteration are computational temperatures T^{k+1} , while the starting condition are temperatures T^k.
The Jacobi matrix defined in Section 4.3.1 expresses the variation of temperatures T^{k+1} in thermocouples position with respect to a variation in heat flux parameters during the k^{th} time segment.
Essentially $\underline{\mathbf{J}} = [\frac{\partial \mathbf{T}}{\partial \mathbf{P}}]$. Definining the error on temperatures in thermocouples locations as:

$$\Delta \mathbf{T}^{k+1} = \mathbf{T}^{k+1} - \mathbf{T}^{k+1}_{\mathbf{real}} \tag{6.1}$$

And linearizing the function $\mathbf{T}^{k+1}(\mathbf{P}^{k+1})$ it is possible to write:

$$\Delta \mathbf{T}^{k+1} = \underline{\mathbf{J}}_J \Delta \mathbf{P}^{k+1} \tag{6.2}$$

where $\Delta \mathbf{T}^{k+1}$ are the errors at t^{k+1}, $\Delta \mathbf{P}^{k+1}$ are the errors on parameters heat flux at the time segment $k+1$. $\underline{\mathbf{J}}_J$ has no superscript since it has been demonstrated that it is constant in Section (4.3.1).
So to evaluate the errors on heat flux for a given time segment, one has to solve a linear system in which $\underline{\mathbf{J}}$ is the matrix of coefficients , $\Delta \mathbf{P}^{k+1}$ is the unkown vector and $\Delta \mathbf{T}^{k+1}$ is the vector of known terms. $\Delta \mathbf{T}^{k+1}$ is defined as:

$$\Delta \mathbf{T}^{k+1} = \sqrt{(\Delta \mathbf{T}^{k+1}_{\mathbf{s}})^2 + (\mathbf{T}^{k+1}_{\mathbf{c}} - \mathbf{T}^{k+1}_{\mathbf{s}})^2 + (\Delta \mathbf{T}^{k}_{\mathbf{s}})^2 + (\mathbf{T}^{k}_{\mathbf{c}} - \mathbf{T}^{k}_{\mathbf{s}})^2} \tag{6.3}$$

where:

- $\Delta \mathbf{T}^{k+1}_{\mathbf{s}}$ is the vector of the errors on smoothed values in all thermocouples positions at time t^{k+1}. It has dimension equals to the number of thermocouples.
- $\mathbf{T}^{k+1}_{\mathbf{c}}$ is the vector of converged computational temperatures at time t^{k+1}. It has dimension equal to the number of thermocouples.

- $\mathbf{T_s}^{k+1}$ is the vector of smoothed thermocouples measurements at time t^{k+1}. It has dimension equals to the number of thermocouples.

Terms relative to time t^k have been inserted since also the starting condition is affected by the same type of errors and this errors can propagate in the computation of temperatures at time t^{k+1}.
Figure 6.2 shows the values of all the 17 elements of the first row of the Jacobi matrix. One can note that diagonal elements are several orders of magnitude higher than non-diagonals elements. This means that an error in temperature at thermocouple n, basically only affects the calculation of parameter heat flux n.
Figure 6.3 shows the diagonal elements of the Jacobi matrix. They have almost the same value, but a little difference can be seen at boundaries thermocouples, because of the presence of a boundary condition.
Figure 6.4 shows the error on smoothing values and errors due to not matching computational temperatures with smoothed one. In particulary it refers to the thermocouple at z=0.2210 m. One can note that errors due to precision are bigger than errors due to not matching (note that this last source of error is dependent on the chosen threshold value defined by the stopping criteria).
Figure 6.5 shows the errors on heat flux estimation, due to the precision errors of thermocouples vs time for the parameter point located at z=0.2210 m. One can note that errors have a peak of 0.5 MW at the beginning of the simulation but then they reduce to about 0.1 MW. These are very small relative errors. In fact in Figure 6.6 one can note that errors due to precision are negligible if compared to the absolute values.

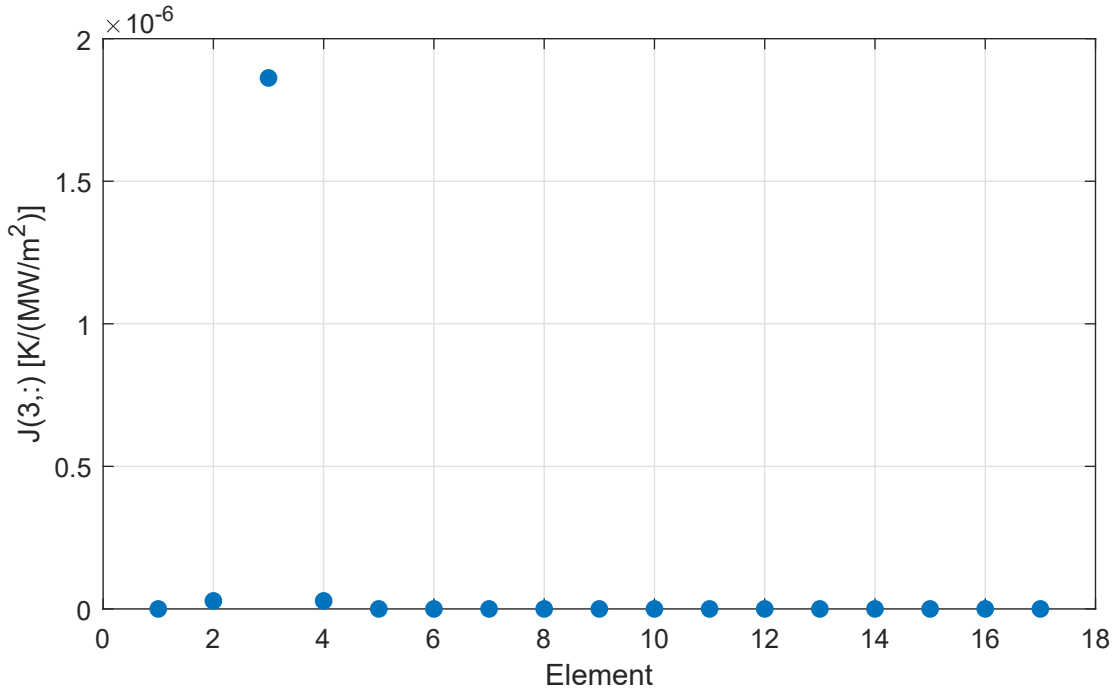


Figure 6.2: Elements of the first row of the Jacobi matrix.

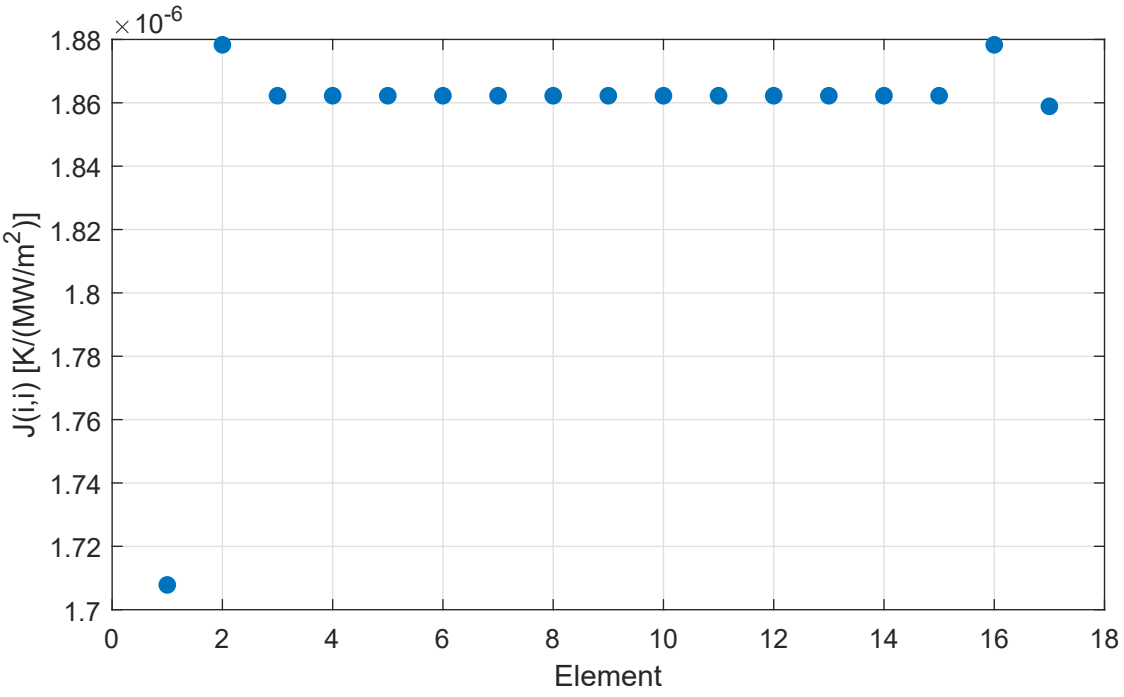


Figure 6.3: Diagonal elements of the Jacobi matrix.

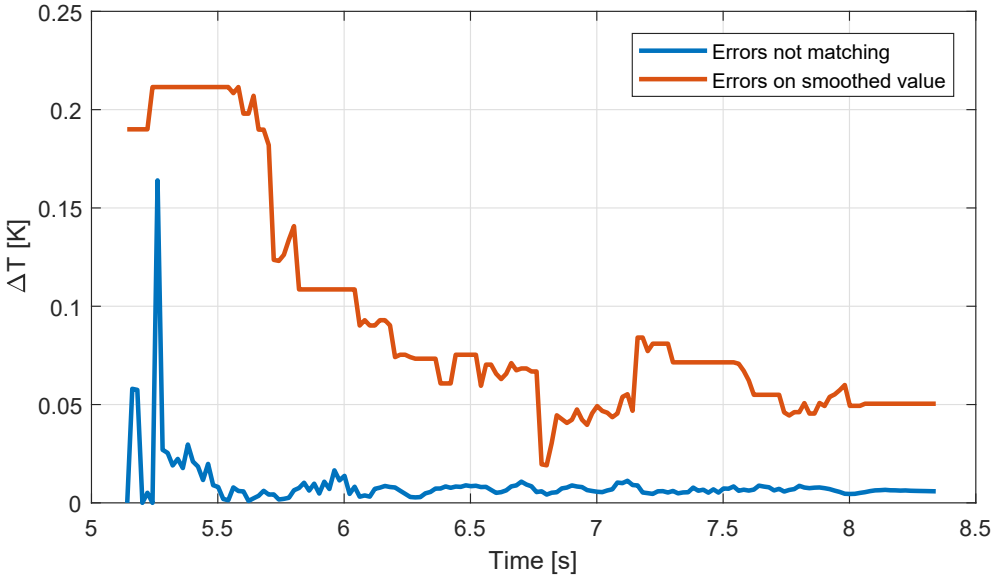


Figure 6.4: Random errors in temperatures vs time for thermocouples at $z = 0.2210$ m.

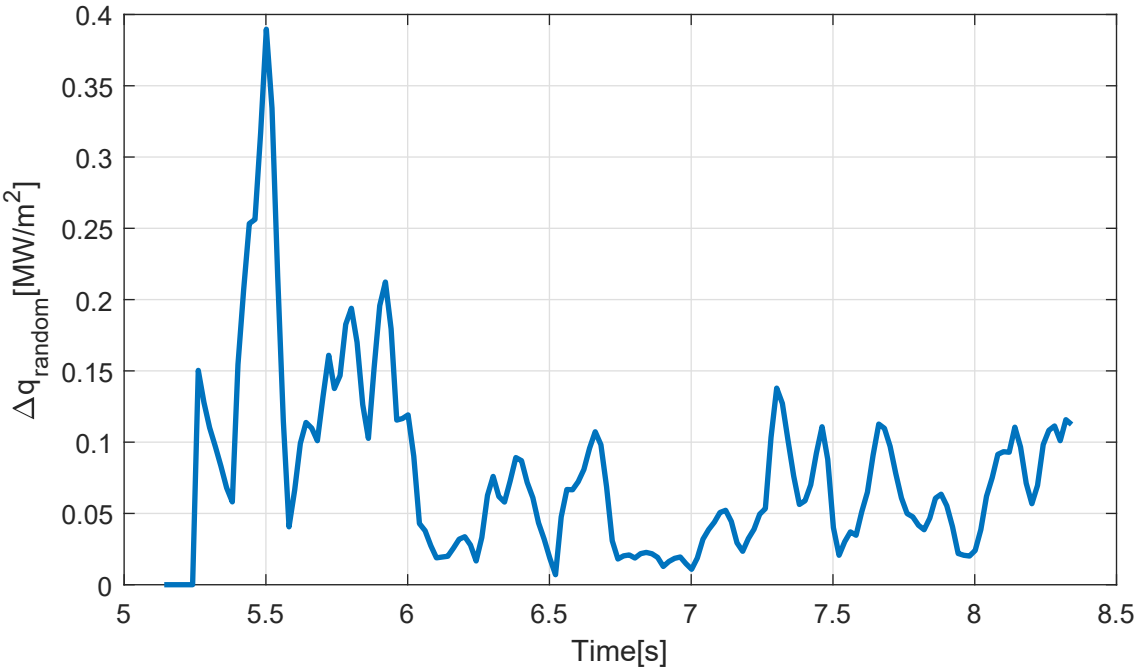


Figure 6.5: Random errors heat flux vs time at $z = 0.2210$ m.

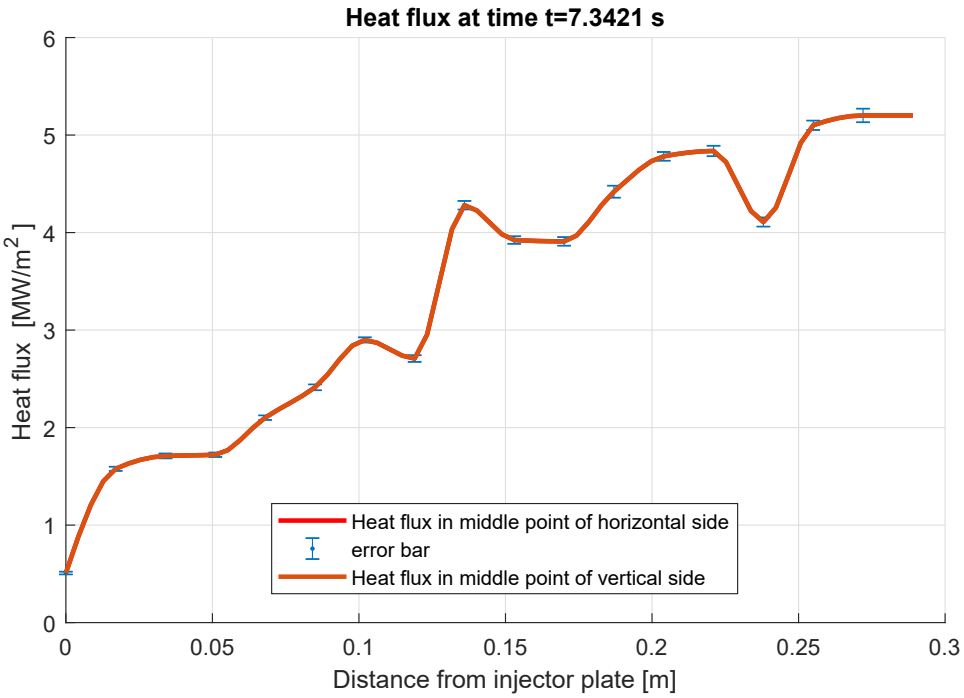


Figure 6.6: Random errors heat flux over z at evaluation time.

6.1.2 Accuracy errors

In this section accuracy errors will be examined and other sources of errors will be neglected (also precision errors). If there are no random errors, accuracy is here intended as the distance between real value and measured value. Accuracy of type T thermocouples is 1^oC [6]. Note that since heat flux equations deal with temperature differences and not absolute temperatures, if the accuracy of thermocouples is constant, no accuracy errors in the evaluation of heat fluxes are present.

Errors due to accuracy are only present if a change in accuracy occurs during transient measurements. It is reasonable to assume that accuracy decreases as temperature increases. To be more conservative the following hypothesis is done:

when there is no combustion, thermocouples have 1^oC of error in a certain direction. When combustion starts and thermocouples start measuring transient temperatures, errors in thermocouples will change linearly and at a certain time instant of the experiment will become 1^oC but in the opposite direction with respect to the beginning. This is equivalent to having 2^oC of error instead of 1^oC, since as already mentioned, temperatures differences are important and not absolute values.

Figure 6.7 shows the hypothesis made for accuracy errors. The real behaviour should follow one of the lines in the figure. One ca note that a linear change in accuracy over

time has been assumed. Different lines in figure, represent all the possible behaviors that thermocouples accuracy could have. Bordeline cases are:

- A rapid change in accuracy in a very small time and then a constant accuracy (the line with the higest slope).
- A slow change in accuracy which occurs over the all experimental time (the line with the lowest slope).

All other lines represent intermediate cases. One can note that in this way, measured temperatures systematically go far from the real temperatures. Such a situation can't be bordered to a single time step. In each time step there is a **systematic error** which propagates during the solution of the direct problem along time, since the measurement affected by the error is the starting condition for the solution of direct problem in the next time step.
It is easy to understand that the Jacobi matrix is not able to estimate this types of errors. The best way to estimate these errors is to use the sensitivity matrix according to how it has been computed (see Section 4.3.2). In fact sensitivity matrix simulates heat flux perturbations (that linearizing can be intended as the errors) which lead to a perturbed solution in temperature (linearizing the perturbation in temperature can be seen as temperatures errors) and this perturbed solution is kept as starting condition for the solution of the next time step. Figure 4.15 shows that sensitivity matrix computed using constant heat fluxes is bigger than the one computed using real heat fluxes. A bigger sensitivity matrix leads to a smaller error estimation. Because of this, when calculating errors using sensitivity matrix, a corrective factor of 1.1 is applied.

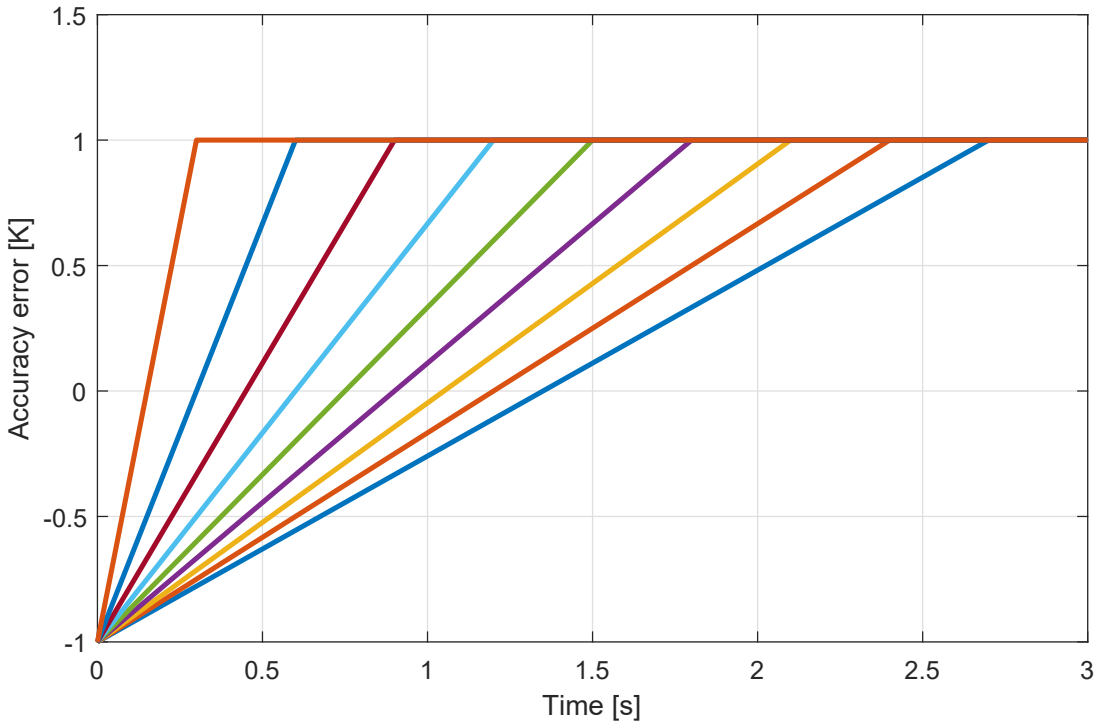


Figure 6.7: Accuracy errors hypotesis.

According to this, errors on estimated heat fluxes, due to thermocouple accuracy errors are computed solving the following linear system:

$$2\Delta\mathbf{T}_{accuracy} = \underline{\mathbf{J}}_S^{k+1}\Delta\mathbf{P}_{accuracy}^{k+1} \tag{6.4}$$

where $\underline{\mathbf{J}}_S^{k+1}$ is the sensitivity matrix in which thesuperscript $k+1$ indicates that this sensitivity matrix is a function of the time step as shown in Figure 4.15.
Figure 6.8 shows errors on heat flux due to accuracy vs time for the parameter point at z=0.2210 m. One can note that errors decrease in time, according to the behaviour of the sensitivity matrix. In fact, an increase in value of the sensitivity matrix means a smaller perturbation in heat flux due to the same perturbation in temperature.
Since sensitivity computed using constant heat flux is similar to sensitivity computed using real converged heat flux, it is possible to compute it only one time and save for estimating accuracy errors in all the tests made on the same chamber, if geometry and material properties don't change.

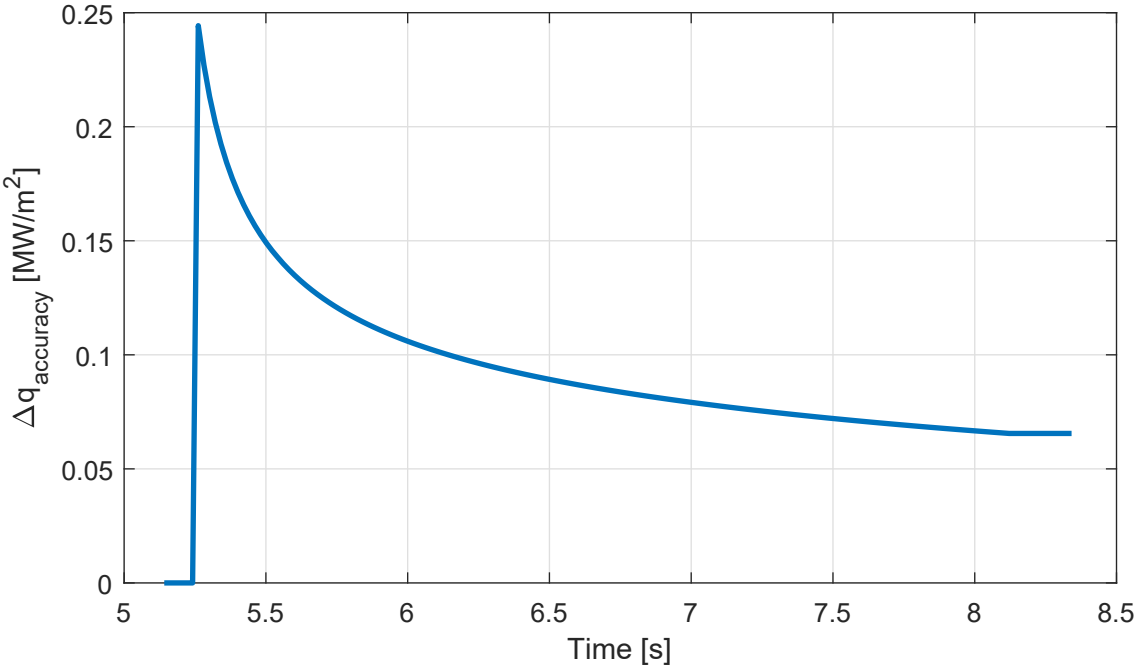


Figure 6.8: Errors on heat flux extimation due to thermocouples accuracy errors ($z = 0.2210$ m).

Figure 6.9 shows an example of errors on estimated heat flux due to thermocouples accuracy over z at evaluation time. One can note, that the errors coming from accuracy are much bigger than precision errors, but still too small to explain the oscillations over z. In fact, errors due to accuracy are about $0.1\ MW$ while oscillations over z are more than $0.5\ MW$.

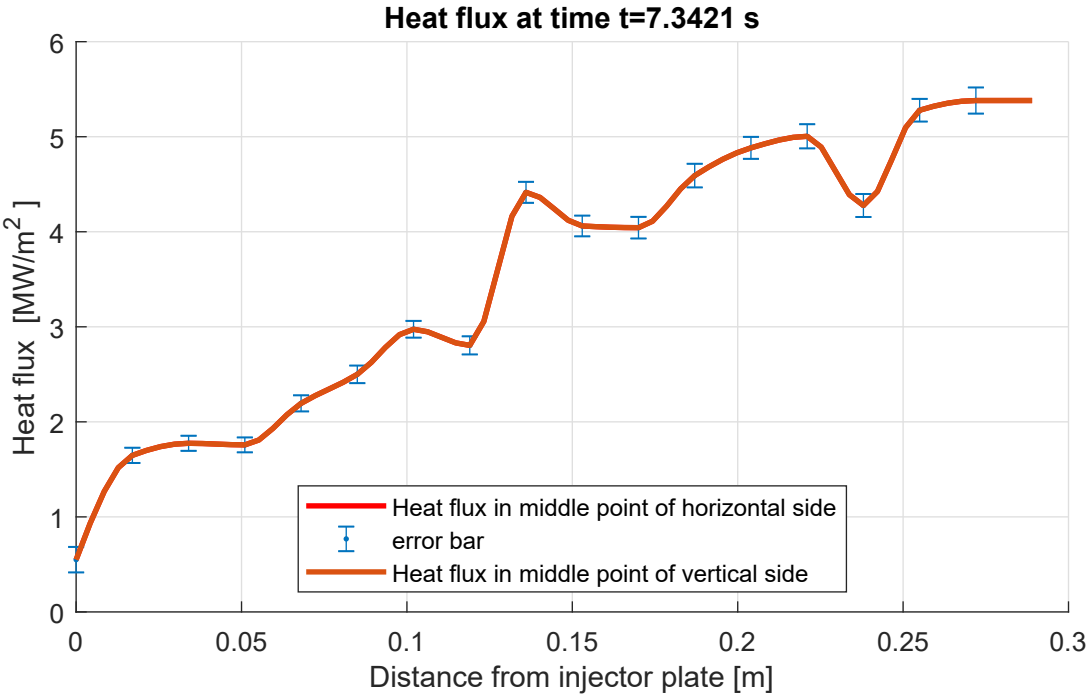


Figure 6.9: Heat flux profile and error bars at evaluation time.

A verification of the validity of the error bars estimation has been done. It has been introduced a known accuracy error in a set of transient measured temperatures, supposing that the original measurements were not affected by accuracy errors. All the 17 thermocouples measurements have been changed introducing the accuracy error of Figure 6.10. Figures 6.11, 6.12, 6.13 show the results of this test. Error bars in these Figures contain estimation of accuracy errors and also errors from not matching temperatures. The test shows that the error bars are able to predict the errors coming from thermocouples accuracy. One can note that error bar are conservative at time in which accuracy error is constant and equal to 1^o. This happens beacuse the heat flux equation deals with temperature differences. If the error in accuracy is linear with time (in this example from time start burning and 0.5 s after time start burning) errors are bigger, but still included in error bars.

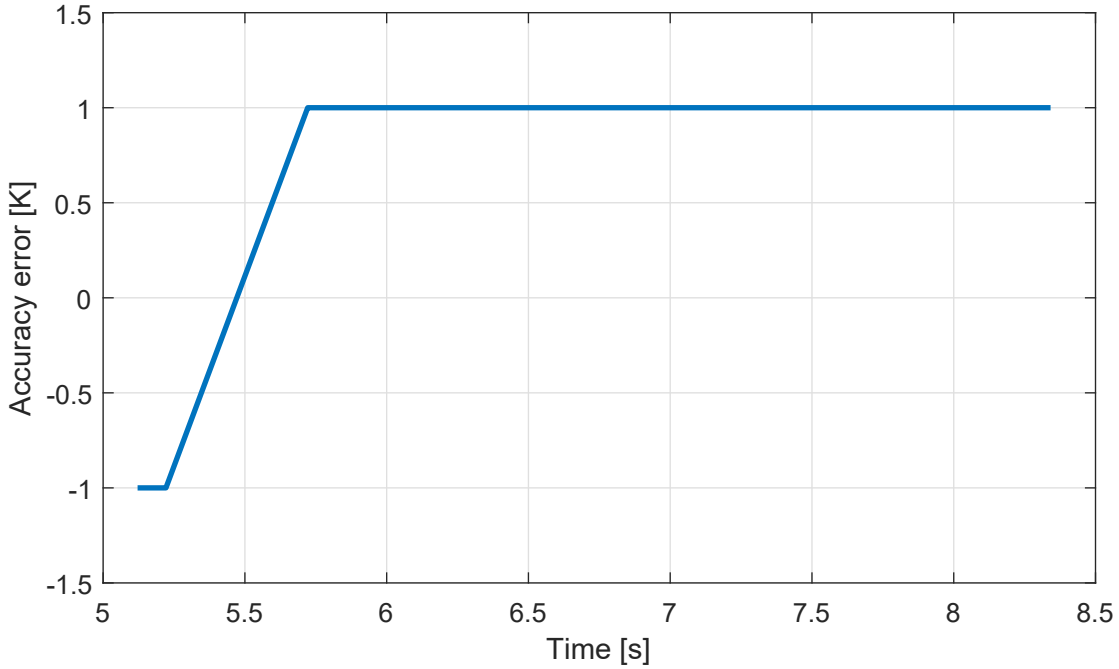


Figure 6.10: Accuracy errors simulated (the same for each thermocouple).

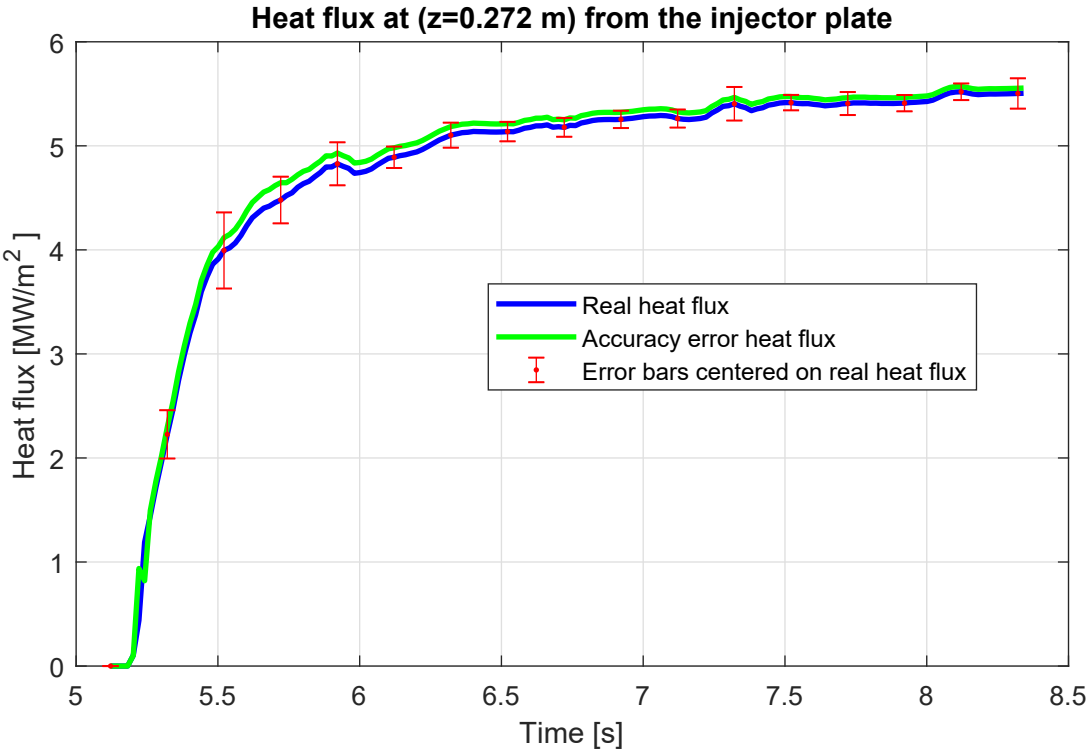


Figure 6.11: Heat flux (real and obtained adding accuracy errors) at $z = 0.272$ m over time.

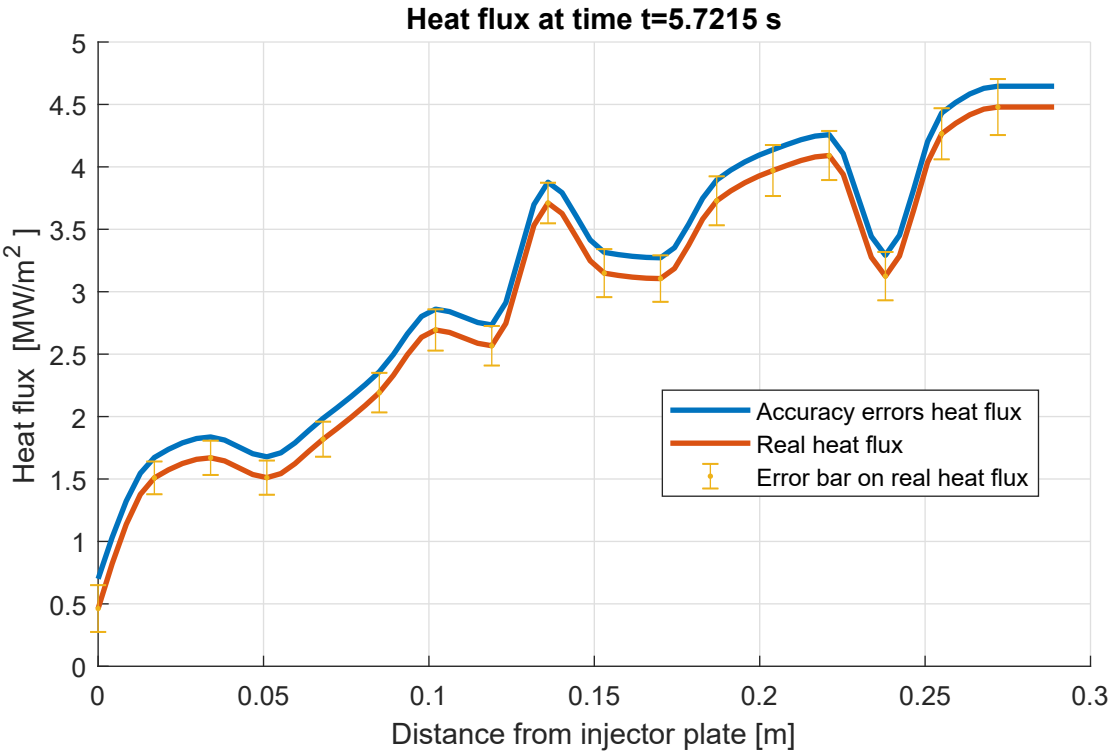


Figure 6.12: Heat flux (real and obtained adding accuracy errors) 0.5 seconds after burning over z.

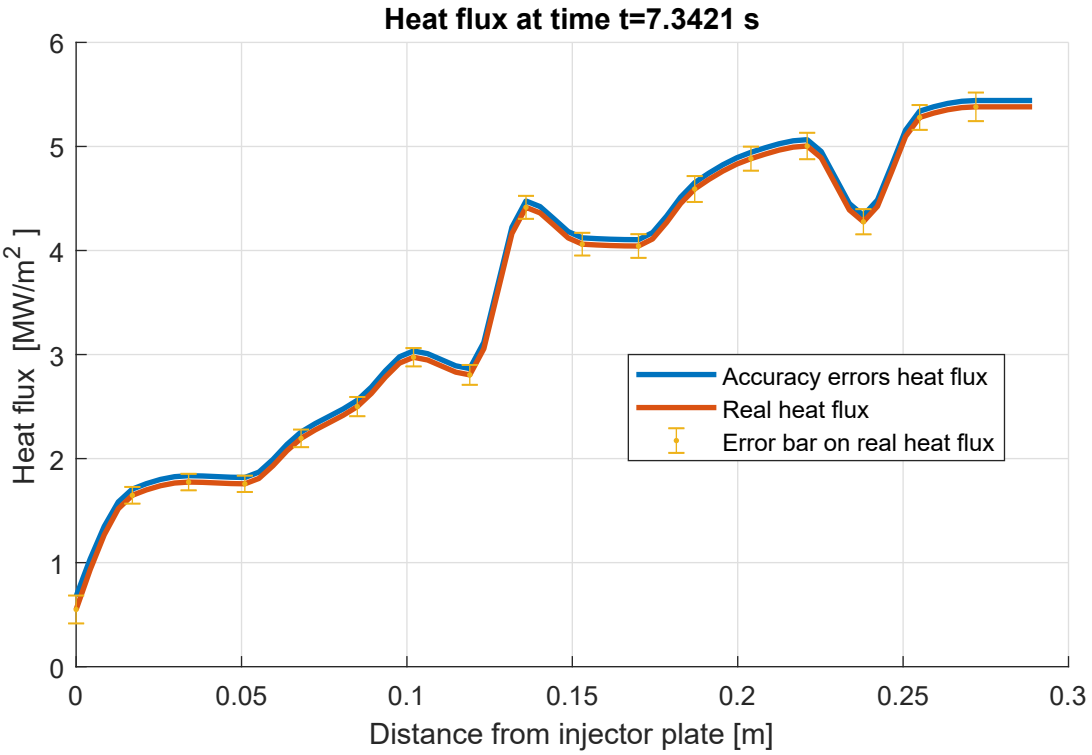


Figure 6.13: Heat flux (real and obtained adding accuracy errors) at evaluation time over z.

6.1.3 Thermocouples locations errors

As mentioned in Oschwald et al. [14] thermocouples locations are not known exactly and this is a source of errors. In fact, during the cycle the function $S(\mathbf{P}) = \mathbf{T_c}\,(\mathbf{P}) - \mathbf{T_m}$ is minimized. T_m is the vector of thermocouples measurements and **$\mathbf{T_c}$ is the vector of computational themperatures in the nominal position of thermocouples**. If the nominal position is not the real position, a wrong matching in temperatures is performed. The final result is a systematic error in temperatures since the computational temperatures in the real position are not matched with the measured one. It is simple to estimate this source of errors. One has to supppose a new thermocouples location and see how much is the difference in temperatures between computational temperatures in the nominal location and in the new one.
The error in temperatures due to error in thermocouples location is so defined as:

$$\Delta\mathbf{T}^k_{\mathbf{location}} = \mathbf{T}^k_{\mathbf{c}}(y_0) - \mathbf{T}^k_{\mathbf{c}}(y_0 + \Delta y) \tag{6.5}$$

where y_0 is the nominal y location of the thermocouples and Δy is the uncertainty of the distance from the wall. These errors are **systematic**. They spread during the time experiment and so the correct matrix to convert these temperature errors in heat flux errors is the sensitivity matrix computed along the whole experimental time. Errors on heat flux paramters are estimated solving the following linear system for each time step.

$$\Delta\mathbf{T}^k_{location} = \underline{\mathbf{J}}^k_S \Delta\mathbf{P}^k_{location} \tag{6.6}$$

Note that, not only the temperature is matched wrongly, but also the position in which sensitivity matrix has been calculated is wrong. Sensitivity matrix depends on the position of the domain in which it is calculated. Figure 6.14 shows the comparison between 2 diagonal elements of sensitivity matrix calculated at 1 mm from hot gas wall and sensitivity calculated at 1.5 mm from hot gas wall (both calculated using constant $5MW/m^2$ heat fluxes). Since errors are bigger when sensitivity matrix is smaller, it is correct to compute a new sensitivity matrix in the new position (but this has a high computational cost) or use a corrective factor on errors on heat fluxes. This analysis showed that a corrective factor of 1.2 on heat flux errors (due to positioning error of 0.5 mm) is needed to take into account that the two sensitivity matrix differ.

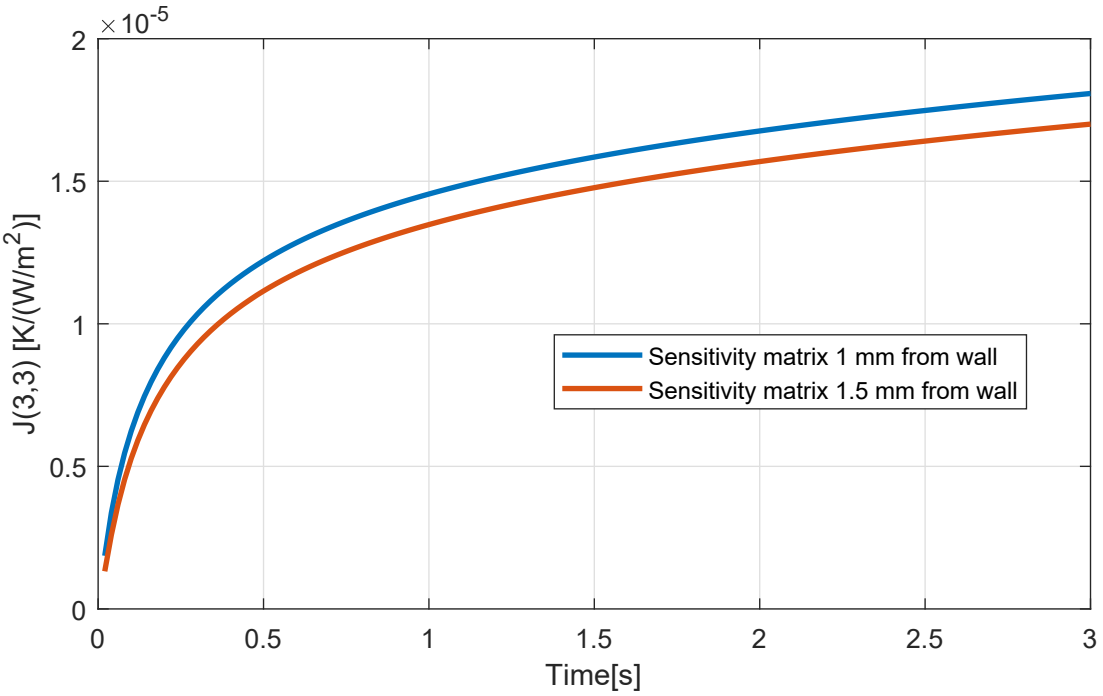


Figure 6.14: Sensitivity matrix over time (1 mm and 1.5 mm from hot gas wall).

Eq. (6.5) doesn't establish what values of Δy should be used. In general the values could be positive or negative. Two examples to estimate errors from wrong positioning have been followed:

1. a constant $\Delta y = 0.5\ mm$ for each thermocouples.
2. $\Delta y = 0.5\ mm$ for a single thermocouples.

. Results are shown in Figures 6.15, 6.16, 6.17. In particular:

- Figure 6.15 shows the Δy for each thermocouples in example 1 and 2.
- Figure 6.16 shows the error on temperatures for the two examples.
- Figure 6.17 shows the relative estimated error on heat fluxes.

Figure 6.17 is particulary important since it shows that if all the thermocouple have the same positioning errors, errors on heat flux are smaller. In fact, if only one thermocouple is affected by positioning error, the relative error on heat flux is bigger.
Because of this, error bars due to errors on thermocouples positions are computed following example 2, for each thermocouple.
Figure 6.18 shows the error on temperature due for a single thermocouple over time. It can be noted that the error continues to increase in time until it reaches an asymptotic

value. Furthemore values of errors are very big, if compared to errors from thermocouples accuracy.
Figure 6.20 shows error bars due to errors in thermocouples locations over z, supposing an error in thermocouples locations of 0.5 mm for each thermocouple. One can note that these error bars are very big if compared to accuracy error bars. It is possible to note that oscillation over z could be caused by thermocouples location errors. As a proof of this, the position of some thermocouples has been changed and the inverse problem has been solved with this new values of thermocouples locations. In particular thermocouples locations modified are:

- thermocouple at z=0.1190 m $\rightarrow$ y=y_0+0.5 mm.
- thermocouple at z=0.1360 m $\rightarrow$ y=y_0-0.5 mm.
- thermocouple at z=0.2380 m $\rightarrow$ y=y_0+0.75 mm.

Results of this test are presented in Figures 6.21 and 6.22. One can note that oscillations are still present at initial time, but then they are smoothed, in accordance with the hypotesis that they could be caused by the effect of the igniter.

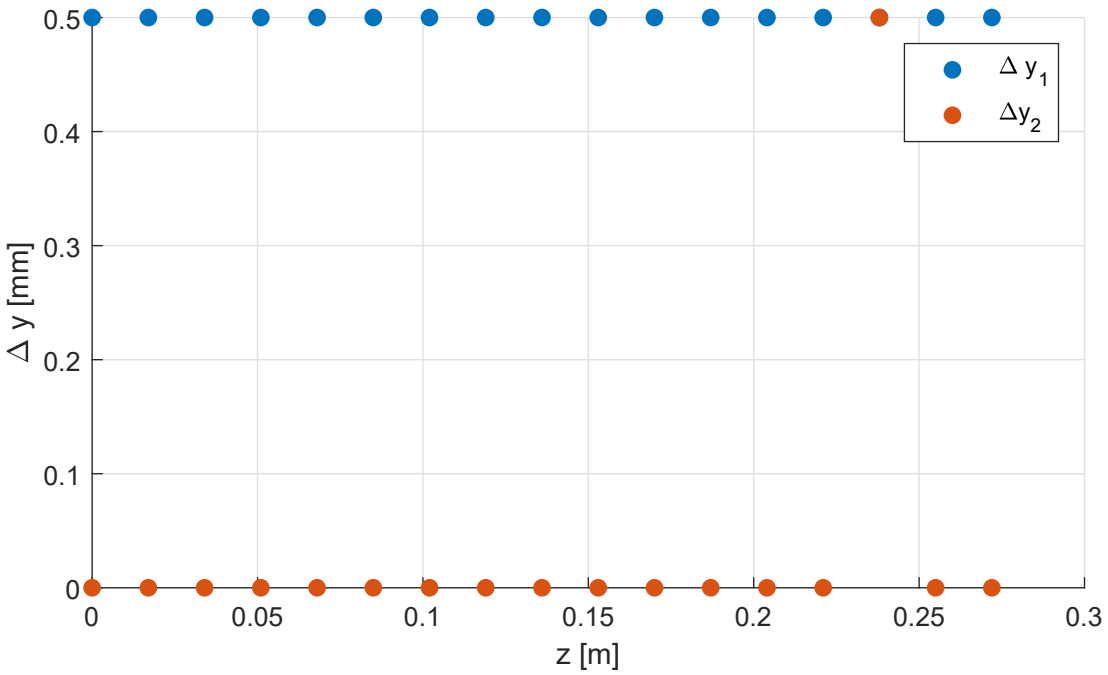


Figure 6.15: Supposed error in thermocouples locations.

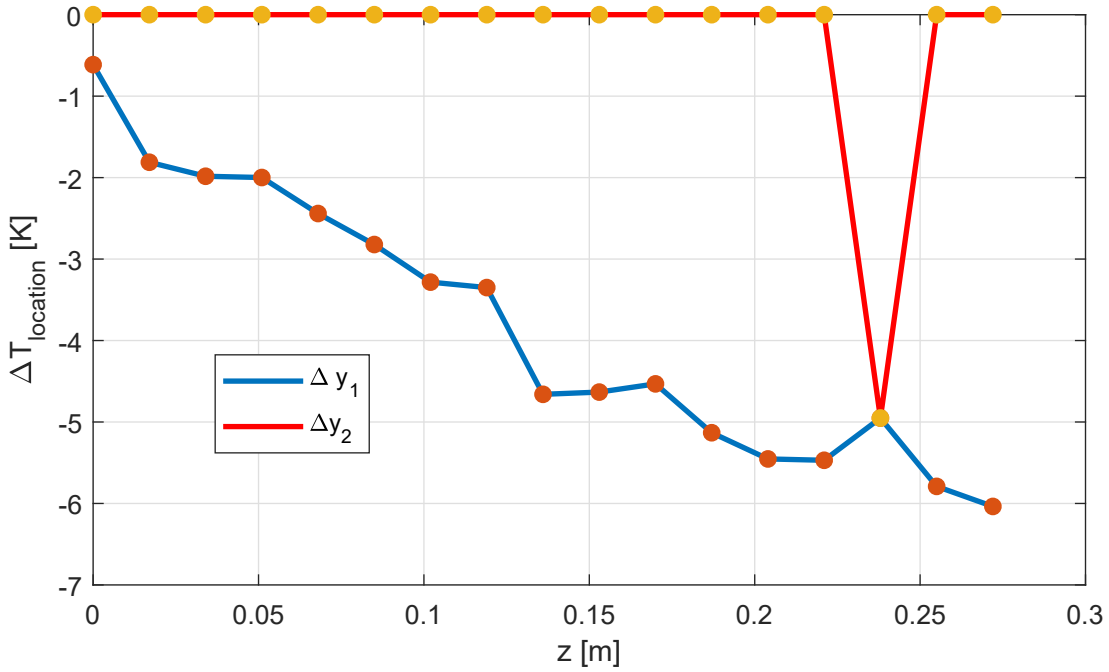


Figure 6.16: Temperatures errors due to wrong information about thermocouples locations at time t=7.1615 s.

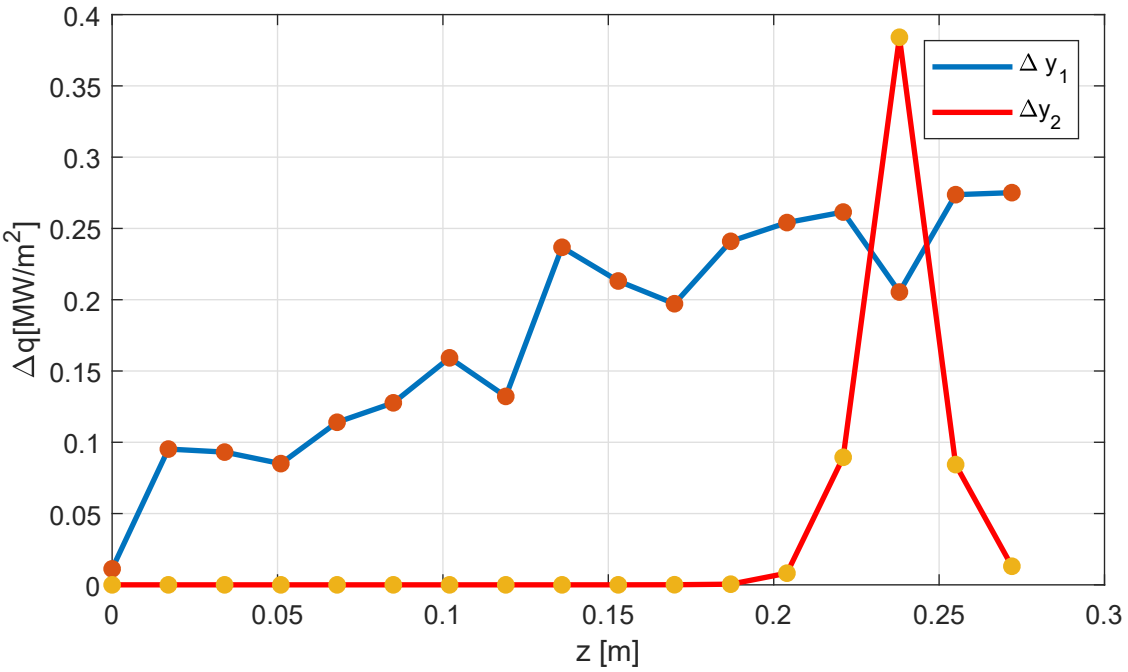


Figure 6.17: Errors on heat flux due to wrong information about thermocouples locations time $t = 7.1615$ s (Errors are given as absolute values).

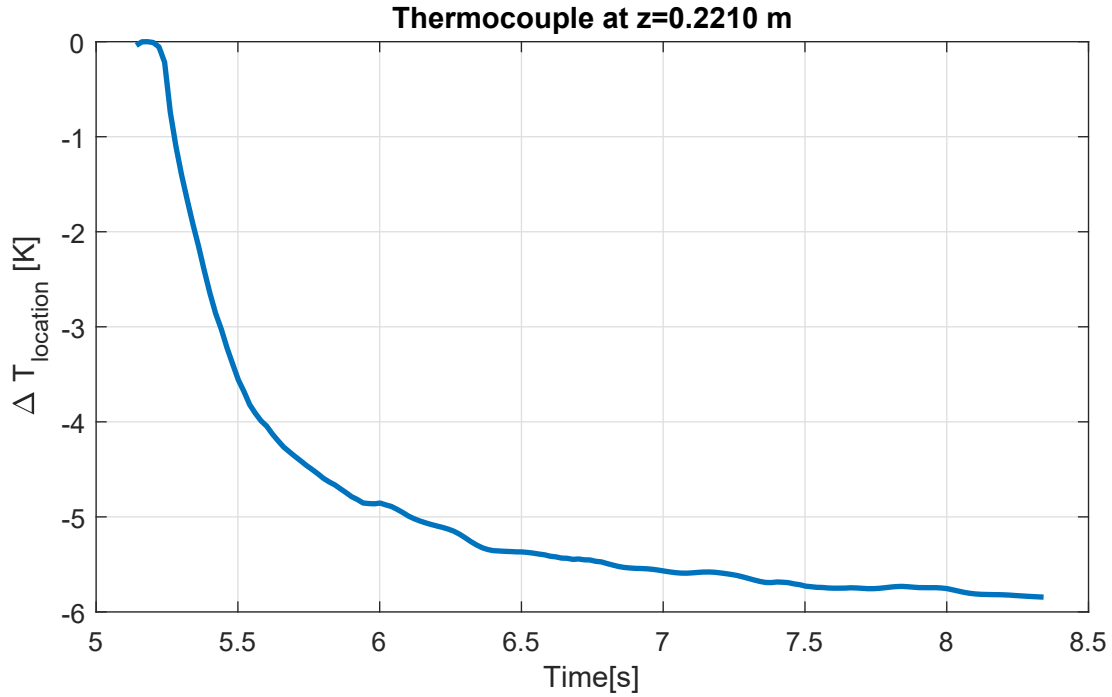


Figure 6.18: Temperatures errors due to wrong information about thermocouples locations over time at $z = 0.2210$ m.

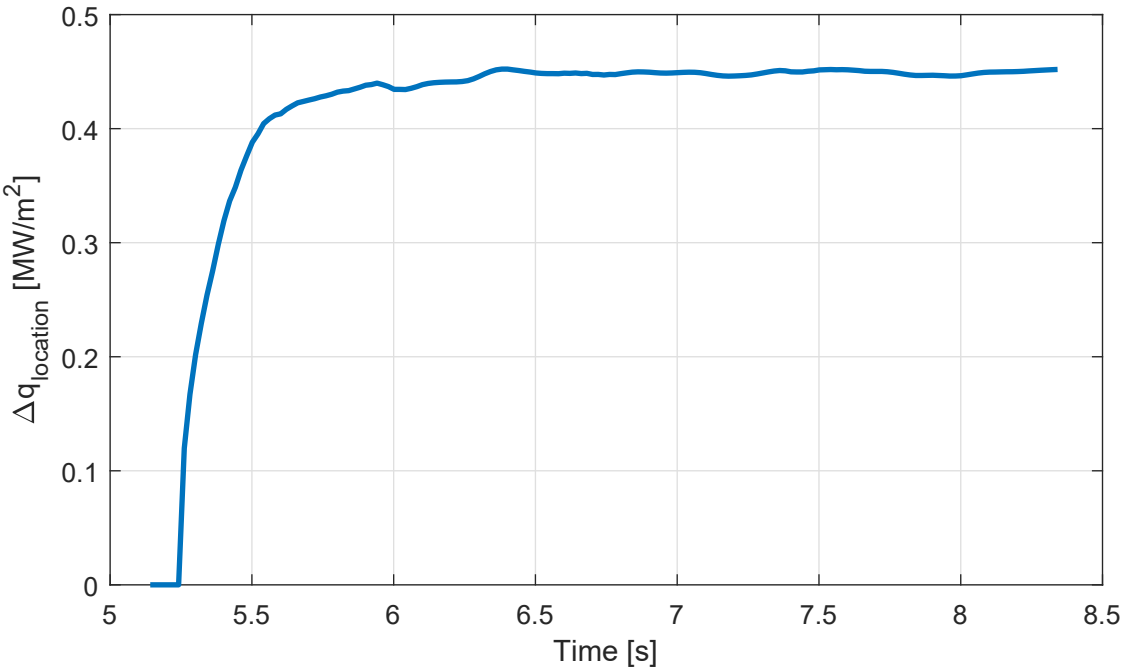


Figure 6.19: Errors on heat flux due to wrong information about thermocouples locations over time at $z = 0.2210$ m.

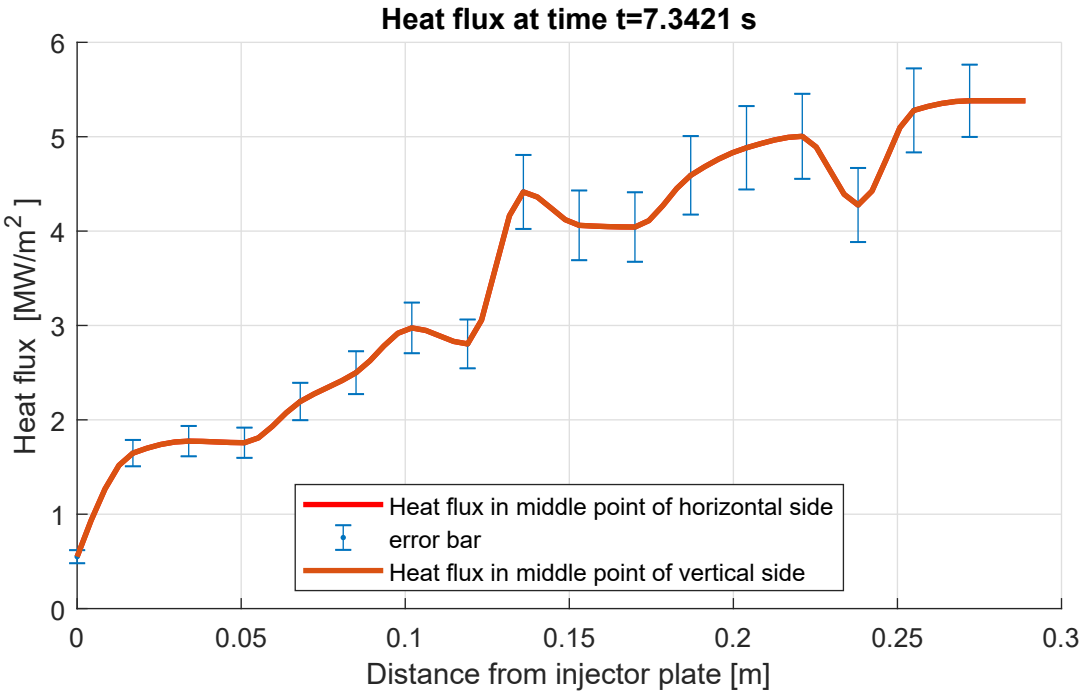


Figure 6.20: Heat flux and error bar (only thermocouples locations over z at evaluation time).

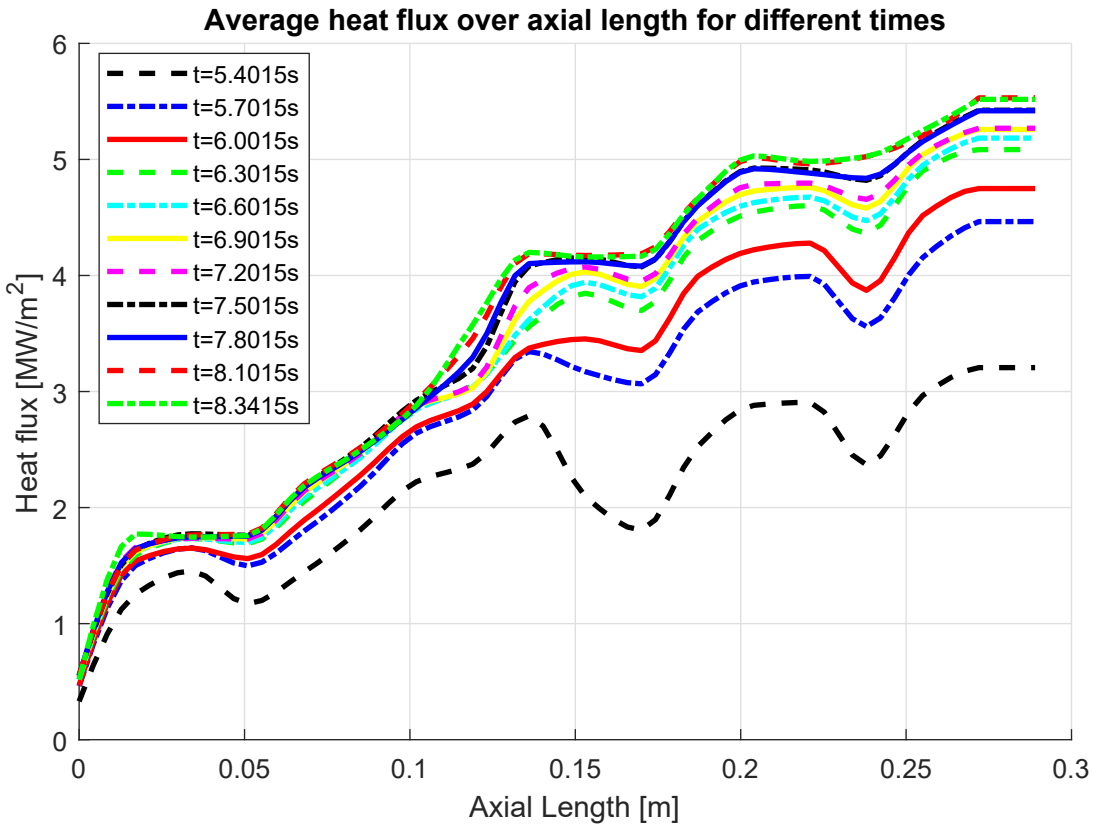


Figure 6.21: Heat flux over z at different times, with modified thermocouples locations.

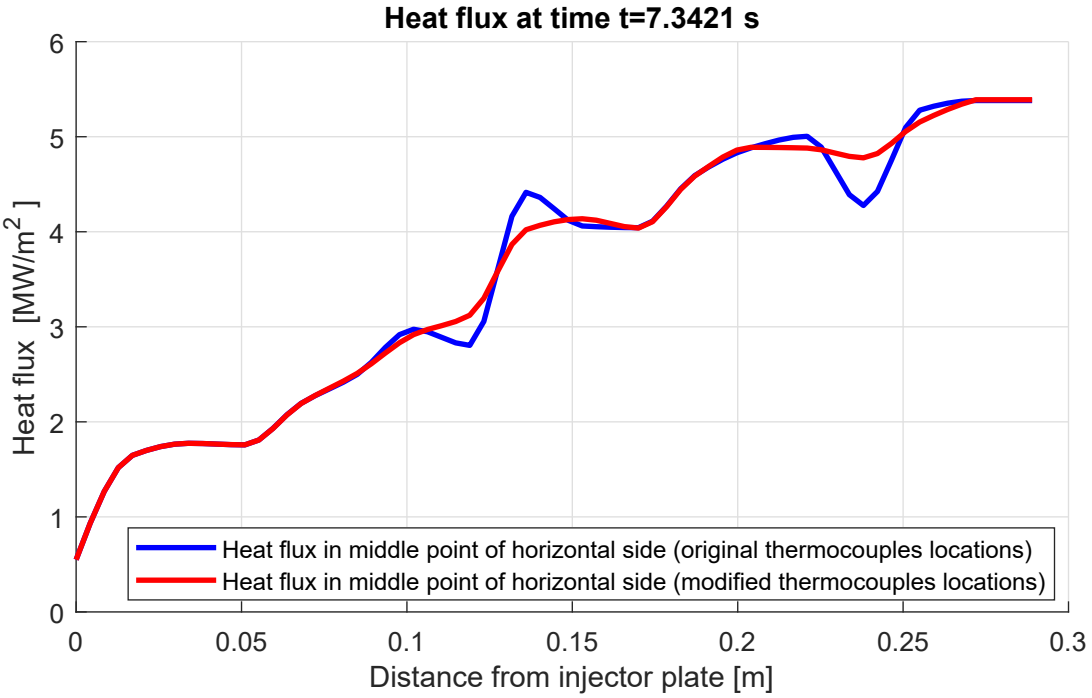


Figure 6.22: Heat flux over z at evaluation time, with modified thermocouples locations.

6.1.4 Material properties errors

Also, the fact that the material properties are not known exactly is a source of error. In particular, the material property which affects the calculation is the thermal diffusivity α. It is possible to estimate errors from not knowing material properties, in the following way:

- Solve the inverse problem using nominal values for material properties.
- Change the values of material properties and solve again the direct proble for each time step, using the converged heat flux, but the new one material properties.
- Evaluate temperature differences in thermocouples location (between temperature computed using nominal material properties and temperatures computed using modified material properties.
- These errors in temperatures are systematic and so temperatures errors can be converted to heat flux errors using the sensitivity matrix.

To evaluate errors from not knowing material properties, one has to solve for each time step the following linear system:

$$\Delta \mathbf{T}_{\alpha}^{k} = \underline{\mathbf{J}}_{S}^{k} \Delta \mathbf{P}_{\alpha}^{k} \tag{6.7}$$

where:

$$\Delta \mathbf{T}_{\alpha}^{k} = \mathbf{T}_{\mathbf{c}}^{k}(\alpha = \alpha_0) - \mathbf{T}_{\mathbf{c}}^{k}(\alpha = \alpha_0 + \Delta\alpha) \tag{6.8}$$

where α_0 is the nominal value of thermal diffusivity and $\Delta\alpha$ is the error on thermal diffusivity. In particular, material nominal values for material properties used in the solution of the inverse problem are:

- $\lambda_0 = 385\ W/mK$.
- $\rho_0 = 8940\ Kg/m^3$.
- $c_{p\,0} = 393\ J/KgK$.
- $\alpha_0 = \frac{\lambda_0}{\rho_0 c_{p\,0}} = 1.0957 * 10^{-4}\ m^2/s$.

Also in this case, sensitivity matrix could have changed (beacause of the change of material properties) and so a corrective factor of on heat fluxes errors is needed. Results from this analysis (supposing an error on α of 10 %) are presented in Figures 6.23, 6.24, 6.25,6.26. In particular:

- Figure 6.23 shows the errors on temperatures due material properties errors in thermocouple at z=0.2210 m . One can note that the error in temperature increases over time.
- Figure 6.24 shows the error on temperatures due to error in material properties for each thermocouples at time 8.1215 s. One can note that temperature errors in a thermocouple are proportional to the respective parameter heat flux.
- Figure 6.25 shows the errors on heat fluxes due to error in material properties at position z=0.2210 m. Error in heat fluxes reaches an asymptotic value after 1.5 s. The order of degree the error from material properties is comparable with error due to thermocouples locations.
- Figure 6.26 shows heat flux and error bar at evaluation time over z. Also in this case is possible to see that the order of magnitude of error is comparable with error from wrong positioning of thermocouples.

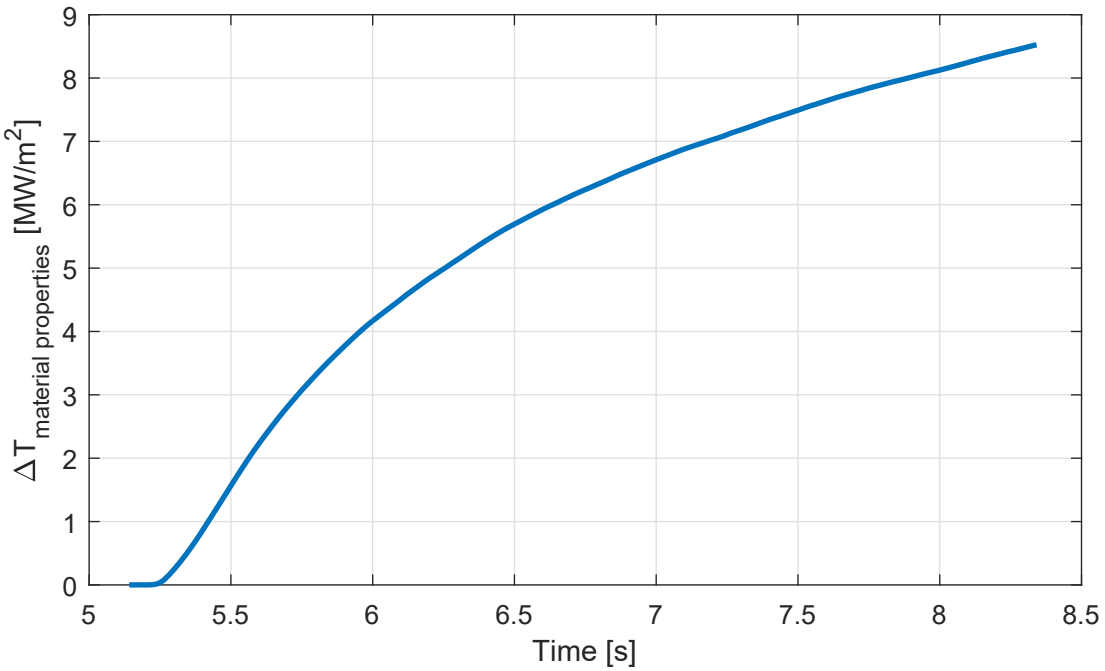


Figure 6.23: Error on temperature due to material properties over time on thermocouples at $z = 0.2210$ m ($\Delta\alpha = 10\%$).

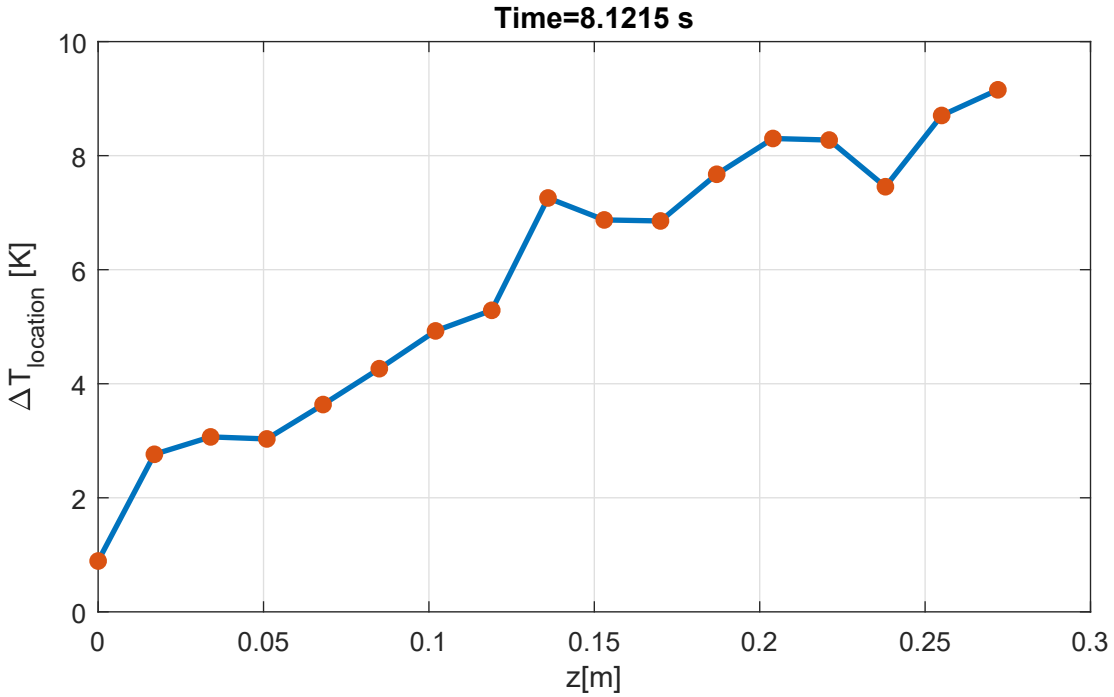


Figure 6.24: Error on temperatures due to material properties over z on each thermocouple ($\Delta\alpha = 10\%$).

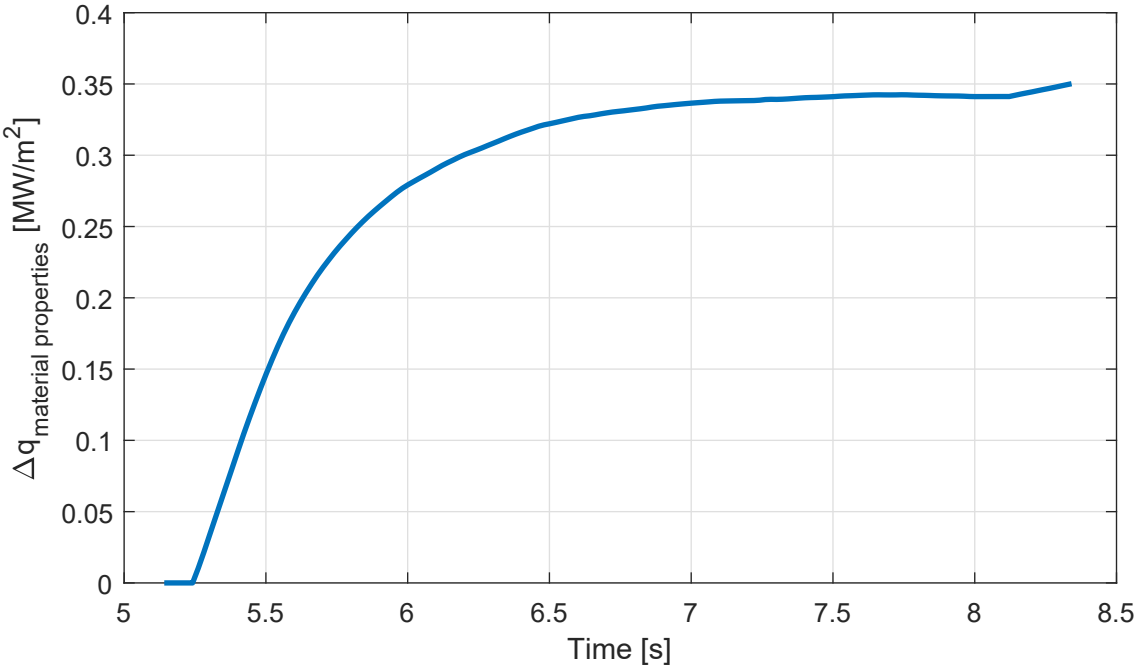


Figure 6.25: Error on heat flux due to material properties over time at $z = 0.2210$ m ($\Delta\alpha = 10\%$).

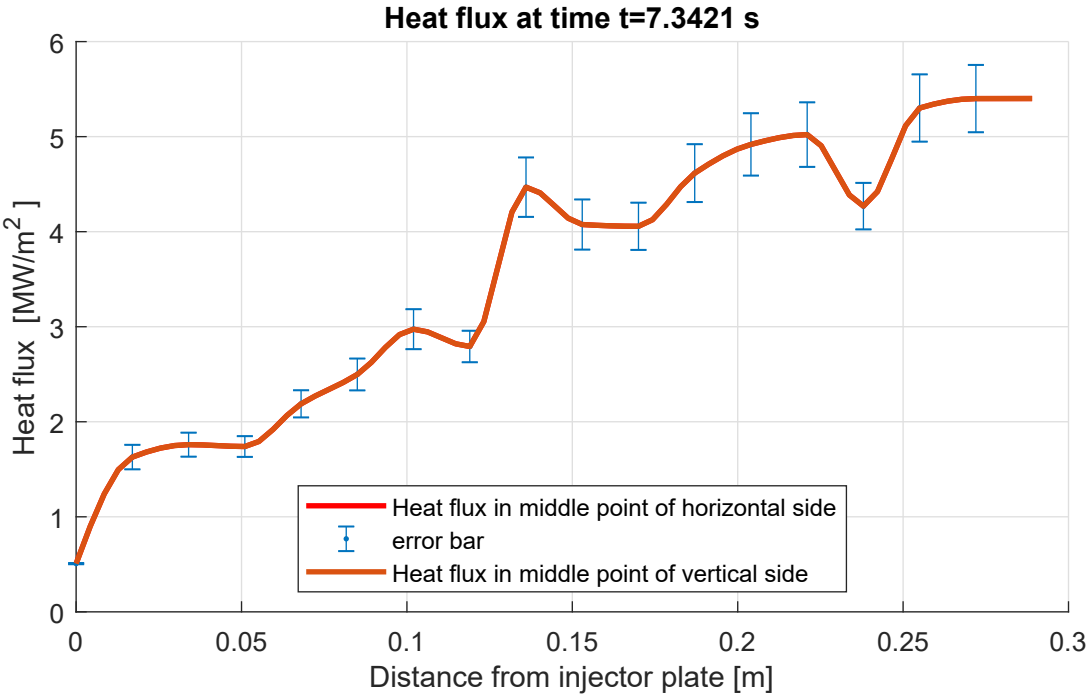


Figure 6.26: Heat flux and error bar due to material properties over z at evaluation time ($\Delta\alpha = 10\%$).

A validation of material properties error bars has been performed. Thermal diffusivity α has been modified as $\alpha = \alpha + 0.1\alpha$ and the inverse problem has been solved using this new thermal diffusivity. Results are depicted in Figures 6.27 and 6.28. The results show a good accordance with errors estimations and also show that error in material properties are certainly not responsable of oscillations over z.

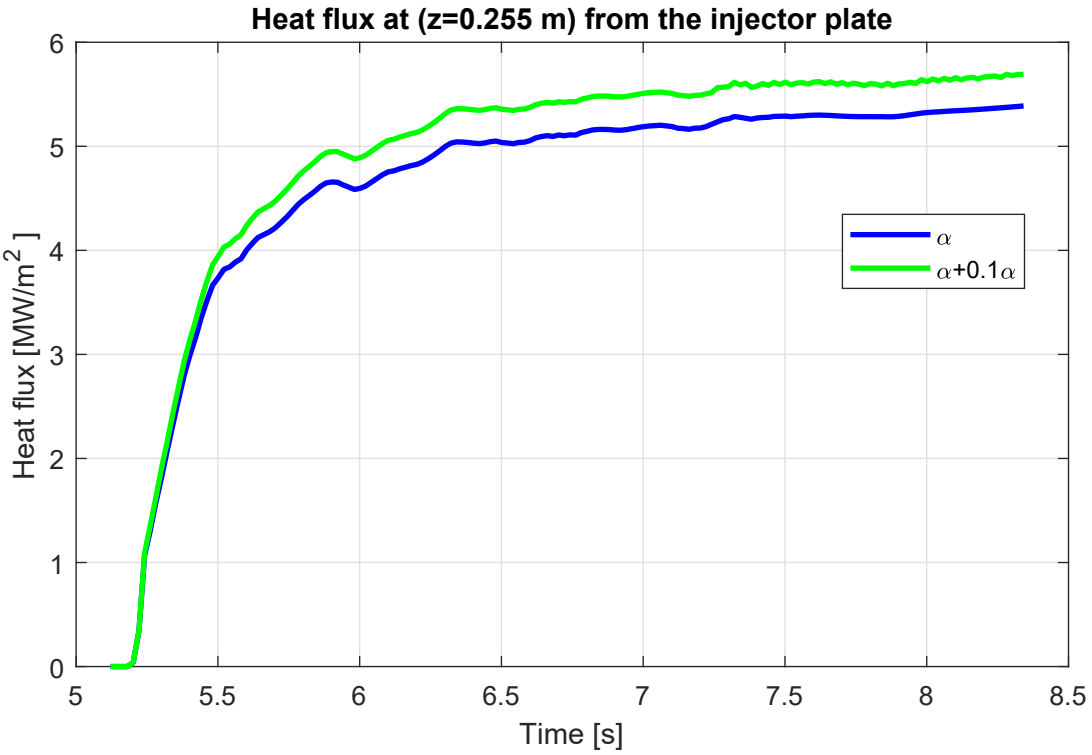


Figure 6.27: Heat flux over time at $z = 0.255$ m with modified and nominal material properties.

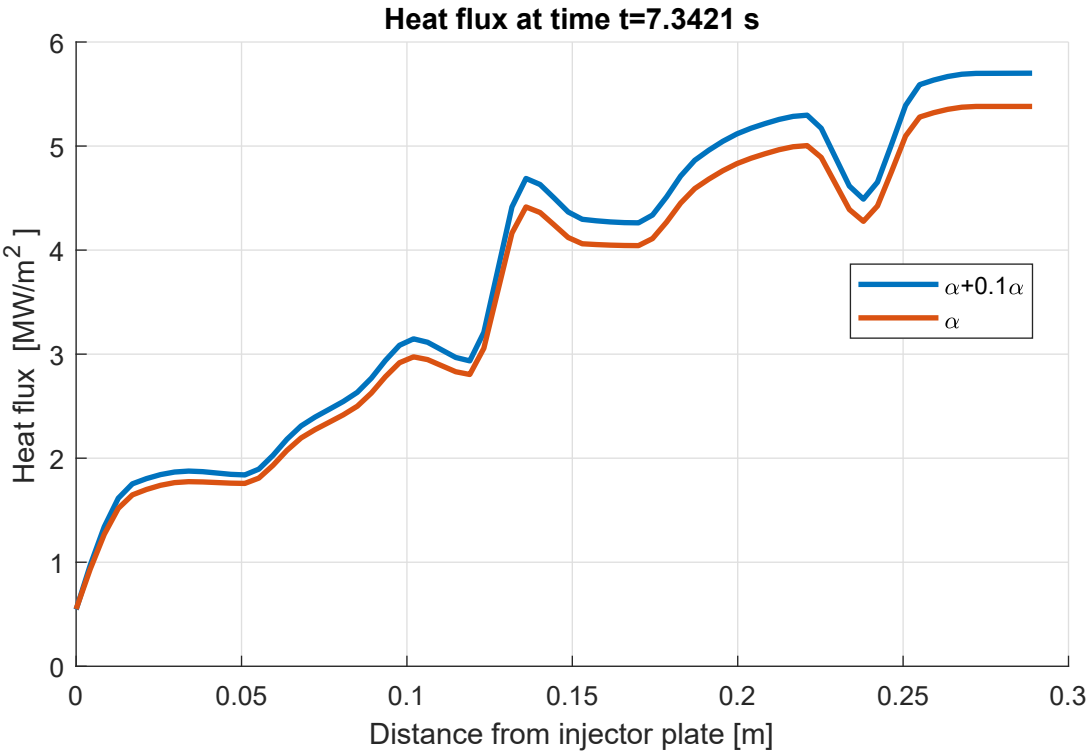


Figure 6.28: Heat flux over z at evaluation time with modified and nominal material properties.

6.1.5 Total amount of errors

The total amount of error estimation is computed as follow:

$$\Delta q_{total} = \sqrt{\Delta q^2_{precision} + \Delta q^2_{accuracy} + \Delta q^2_{locations} + \Delta q^2_{material\ properties}} \tag{6.9}$$

This relationship is valid for each parameter point and for each time. Results for total error estimations are depicted in Figures 6.29 and 6.30. One can note from Figure 6.30 that errors increase as heat flux increases.
It also possible to note from Figure 6.29 that error on heat flux is almost constant in time apart from first time instants.

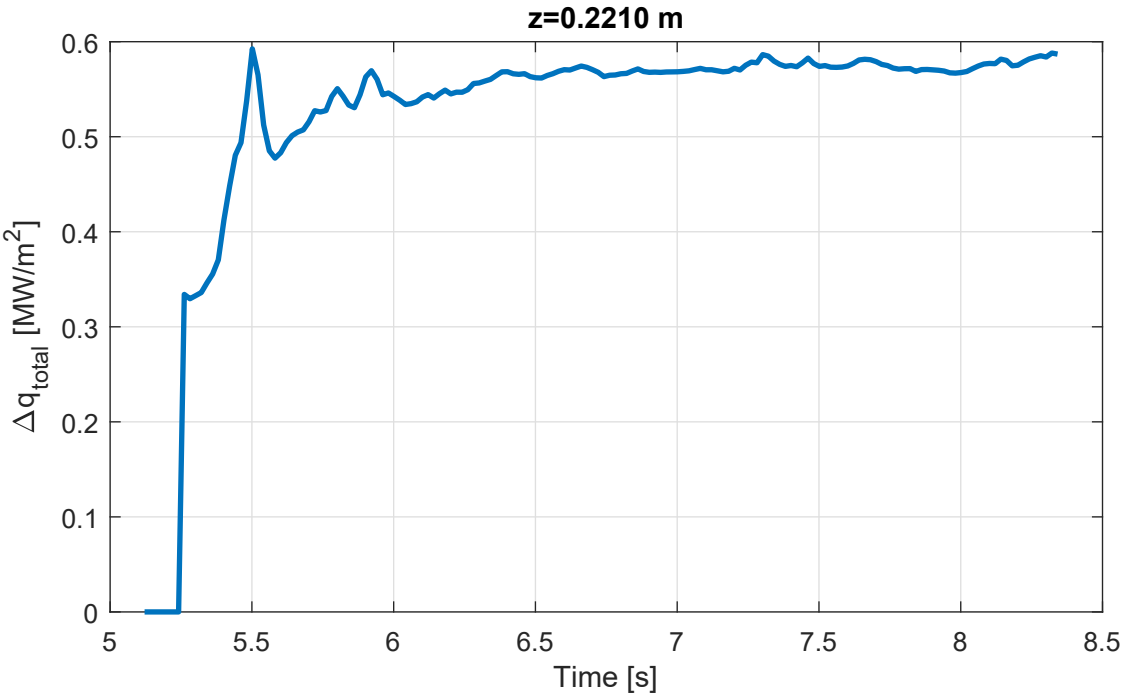


Figure 6.29: Total error estimated at $z = 0.2210$ m vs time.

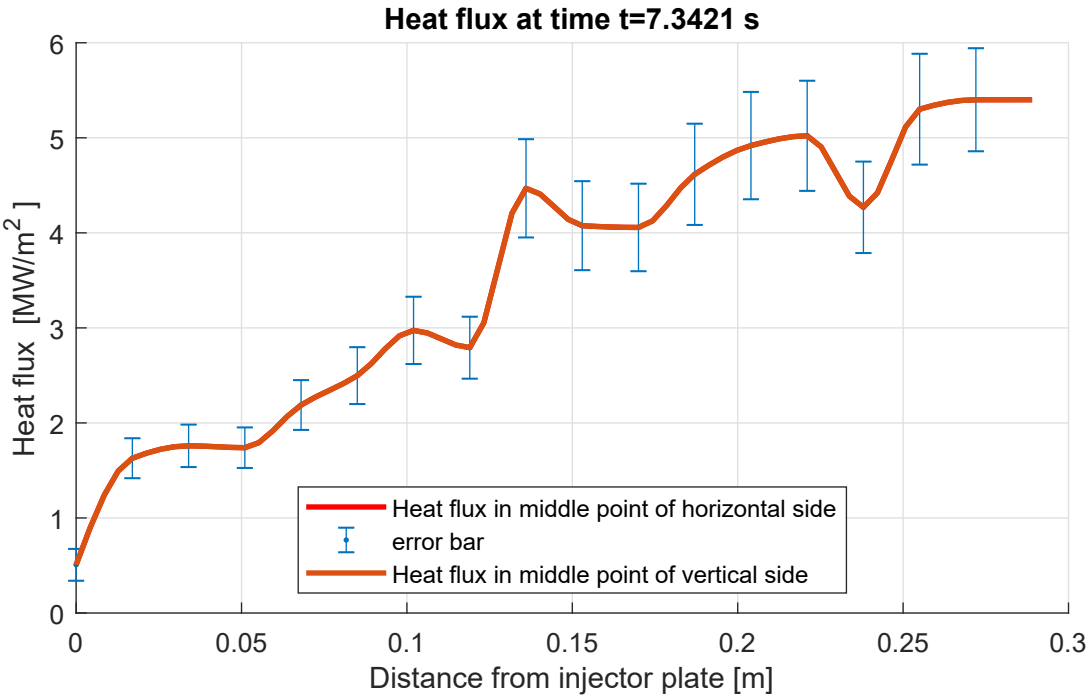


Figure 6.30: Heat flux and total error bars over z at evaluation time with modified and nominal material properties.

Table 6.1 shows the absolute and relatives (normalized with total error) values of errors on esimated heat flux for some locations at time t=7.3421 s. The total value of error

increases with z coordinate (this is mainly because the total amount of error increases as the absolute value of the heat flux increases). An other interesting consideration is that errors in z locations near the faceplate are mainly affected by precision and accuracy errors while on the other side, errors in z location near to the nozzle are mainly affected by thermocouples locations and material properties errors.

z [m]	0	0.0510	0.1020	0.1530	0.2040	0.2550
Δq_{rand} $[MW]$	0.0638	0.0576	0.0362	0.0662	0.0697	0.1049
$\Delta q_{rand}/\Delta q_{tot}$ [%]	38.4048	26.9368	10.2974	14.2992	12.4683	18.1601
Δq_{acc} $[MW]$	0.1423	0.0800	0.0801	0.0801	0.0801	0.0790
$\Delta q_{acc}/\Delta q_{tot}$ [%]	85.7065	37.4088	22.7925	17.2926	14.3199	13.6857
Δq_{loc} $[MW]$	0.0698	0.1601	0.2676	0.3684	0.4414	0.4439
$\Delta q_{loc}/\Delta q_{tot}$ [%]	42.0226	74.8341	76.1350	79.5373	78.9093	76.8754
$\Delta q_{mat\ prop}$ $[MW]$	0.0054	0.1090	0.2102	0.2641	0.3300	0.3549
$\Delta q_{mat\ prop}/\Delta q_{tot}$ [%]	3.2477	50.9270	59.8210	57.0269	59.0058	61.4632
Δq_{tot} $[MW]$	0.1660	0.2140	0.3514	0.4632	0.5594	0.5774

Table 6.1: Errors for different sources and total on heat flux at some z locations ($t = 7.3421$ s).

6.2 Errors on hot gas wall temperatures

In Section 6.1.5 the total amount of errors on heat flux for each time t_i and for each parameter location has been estimated. Recalling that computational temperatures in the domain and in particular hot gas wall temperatures are basically a function of the applied heat flux parameters :

$$\mathbf{T_{hot\ gas}} = F(\mathbf{P}) \tag{6.10}$$

and recalling that the sensitivity matrix:

$$\underline{\mathbf{J}}_S = \left[\frac{\partial \mathbf{T}}{\partial \mathbf{P}}\right] \tag{6.11}$$

it is possible to estimate errors on hot gas wall temperatures coming from errors on heat fluxes linearizing the problem in the following way:

$$\Delta \mathbf{T}^k_{hot\ gas} = \underline{\mathbf{J}}^k_{S\ hot\ gas} \Delta \mathbf{P}^k_{total} \tag{6.12}$$

where:

- $\Delta \mathbf{T}^k_{hot\ gas}$ is the vector of errors on hot gas wall temperatures (in particular in parameters locations) at time segment k.

- $\underline{\mathbf{J}}^k_{S\ hot\ gas}$ is the sensitivity matrix calculated in the 17 parameter points[2] at time segment k.
- $\Delta\mathbf{P}^k_{total}$ is the vector of the total amount of errors on heat flux parameters at time segment k.

Figure 6.31 shows the comparison between a diagonal element of the sensitivity matrix computed in thermocouples locations and sensitivity matrix computed in parameters locations. One can note that the sensitivity matrix in paramter points (on the hot gas wall) is bigger than the one computed in thermocouples locations. This indicates that the temperatures on the wall are more influenced by heat fluxes than the temperatures in thermocouples locations. (i.e for a fixed ΔP, ΔT is bigger on the wall than in thermocouples locations). Furthermore, looking at Figure 6.14, one can note that also in this case, the sensitivity matrix computed further from the wall is smaller. So in general, sensitivity matrix reduces its absolute value as the distance from hot gas wall increases.

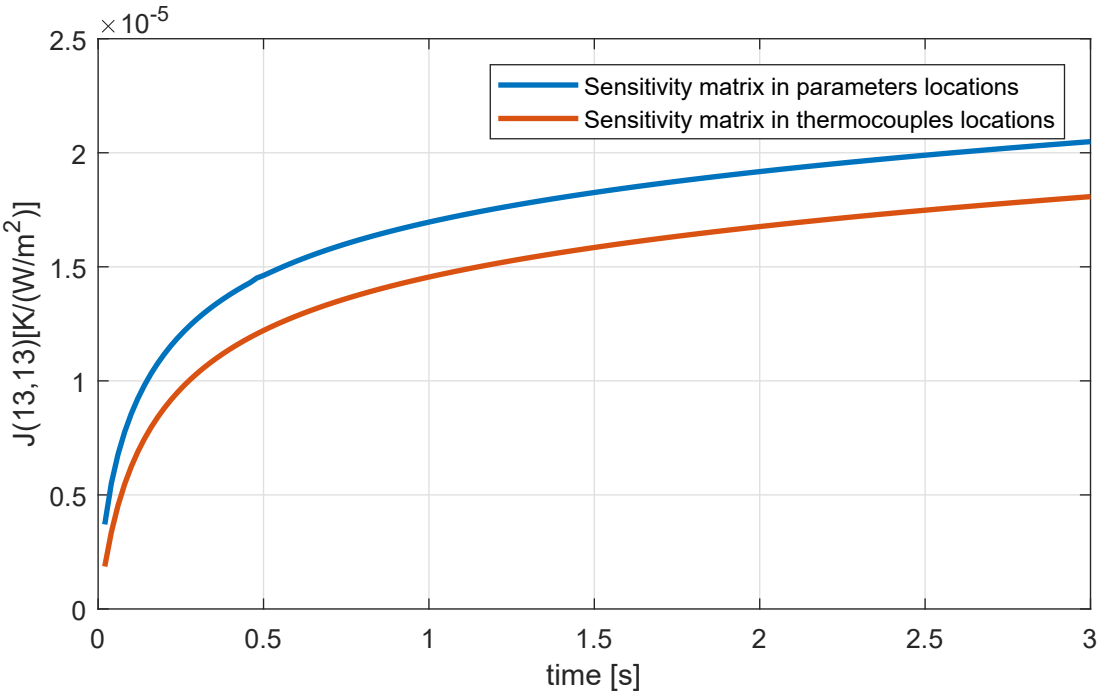
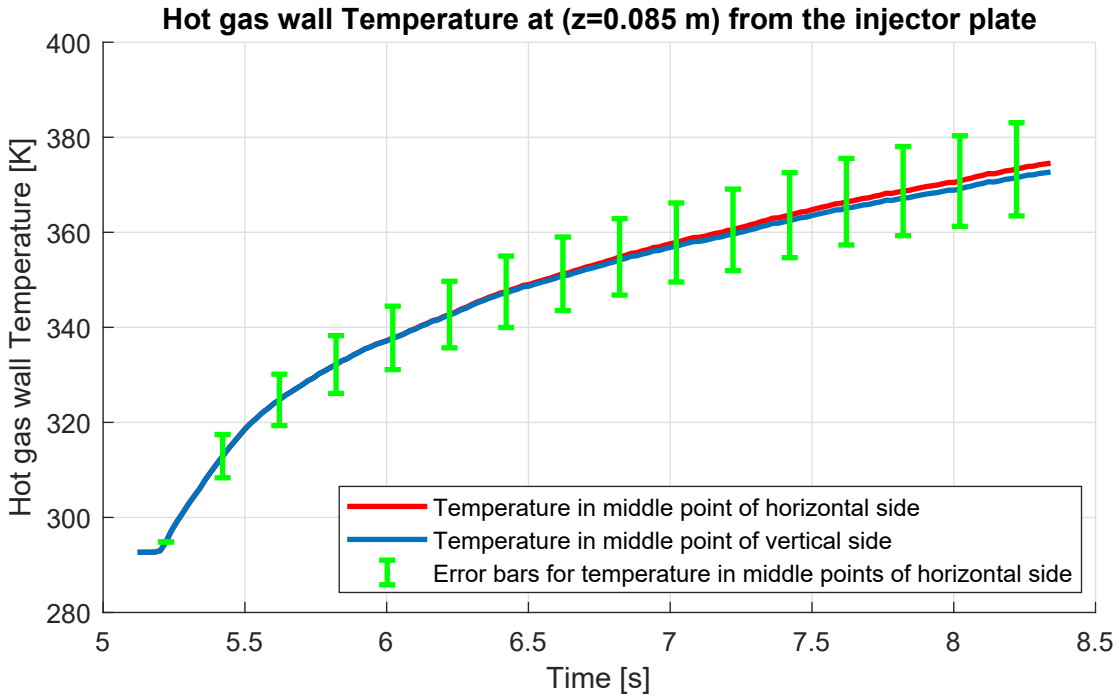


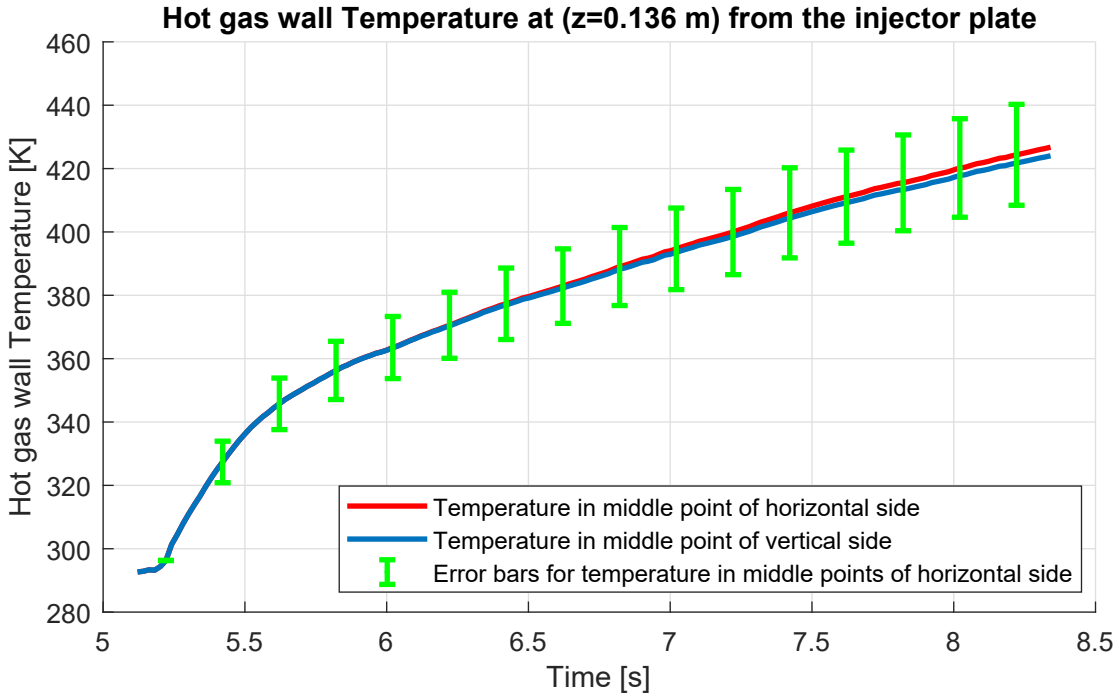
Figure 6.31: Sensitivity matrix over time (computed in thermocouples locations and on the hot gas wall).

Figure 6.32 shows the temperatures error bar in some parameter points over time. It is possible to note that errors increase with time, but then reach an almost stationary value.

[2]Note that the sensitivity matrixes used in Section 6.1 are calculated in thermocouples locations, while the sensitivity matrix defined here is calculated on 17 parameters locations on the wall.

As a matter of fact they continue to increase slightly, according to the slightly increasing of the sensitivity matrix. In fact according to 6.12 error bars on temperatures are influnced by the sensitivity matrix (which increases in time) and by the errors on heat flux, which suddenly reach a stationary values (see Figure 6.29). Figure 6.33 shows temperatures in middle point of horizontal side over z at different times. Errors increases as temperatures increase and furthermore one can note that oscillations in temperatures over z could have been introduced by errors on heat fluxes.


Hot gas wall Temperature at (z=0.085 m) from the injector plate
Hot gas wall Temperature [K]
Time [s]
Temperature in middle point of horizontal side
Temperature in middle point of vertical side
Error bars for temperature in middle points of horizontal side


Hot gas wall Temperature at (z=0.136 m) from the injector plate
Hot gas wall Temperature [K]
Time [s]
Temperature in middle point of horizontal side
Temperature in middle point of vertical side
Error bars for temperature in middle points of horizontal side

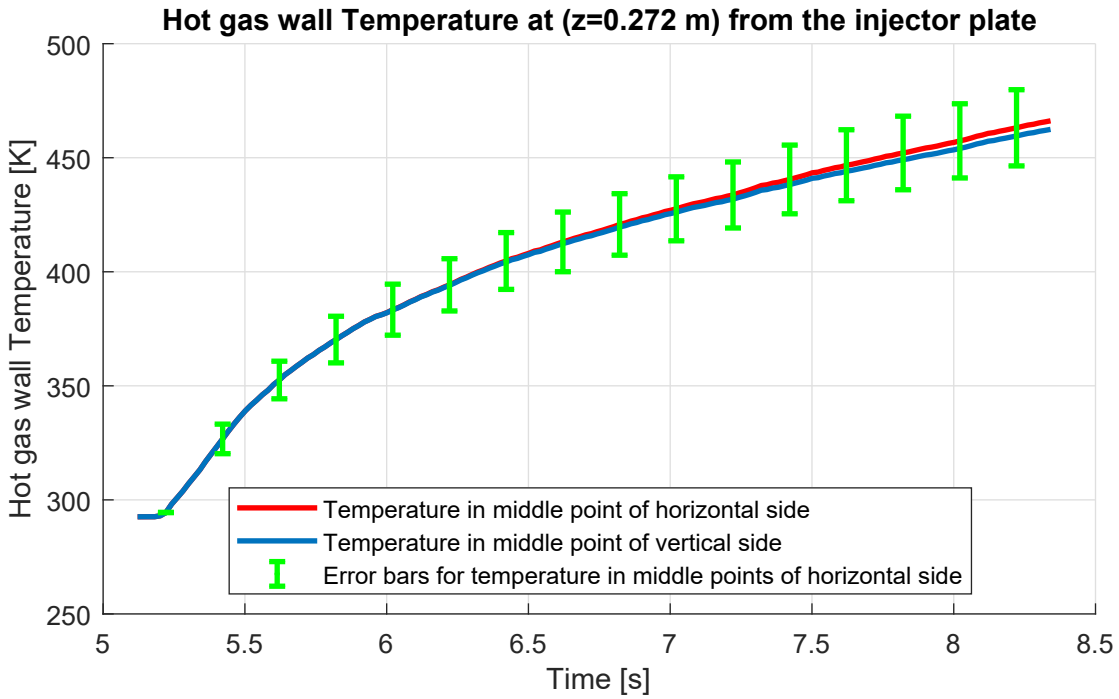


Figure 6.32: Hot gas wall temperatures and error bars over time.

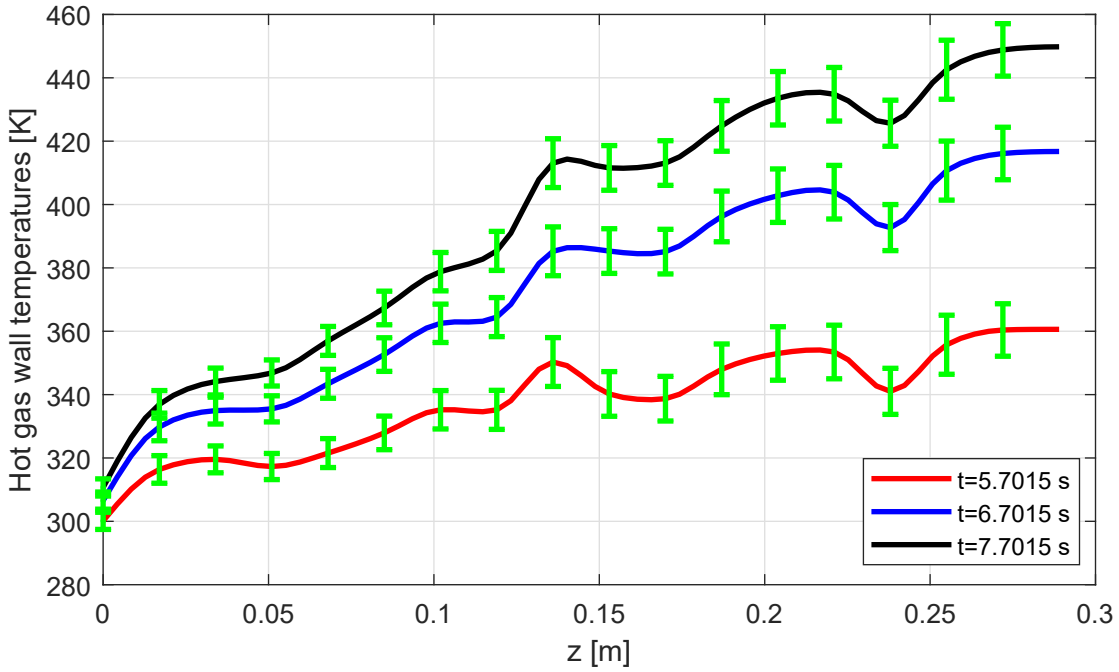


Figure 6.33: Hot gas wall temperatures and error bars over z.

Chapter 7

Conclusion

Preliminary modifications made on RoqFITT (new time handling and Newton Rapshon method) have shown good results from the computational time point of view. Thanks to these two modifications computational time was reduced by more than 70 % while the temperature and heat flux results remained the same.
New time segments handling allows to solve only one direct problem for each iteration instead of two. This is the reason for which new time handling is time saving.
The Newton Rapshon method is very fast thanks to the constance of the Jacobi Matrix over time (when the combustion has started all the elements of the Jacobi matrix remain constant time step after time step until the end of the experiment). This allows to calculate the Jacobi matrix for the solution of the inverse problem one time out of the optimization cycle and use it for all the optimization segments. If sensitivity matrix was not constant, Newton Rashon method would have been very time consuming since at each iteration a new Jacobi matrix should have been computed (this would have cost the solution of 18 direct problems for each iteration).
Since in general, inverse problems have not a unique solution, one can assume that the initialization of heat flux at each first iteration could influence the converged solution. One has tried to initialize heat flux using a linear interpolation along the axial coordinate for all the time segments instead of initializing it using the previous time segment converged solution. This test has shown that an influence of the heat flux initialization does not exist.
Smoothing thermocouples measurements over time using other methods instead of using 'moving average' and increasing the number of points on which the smoothing acts has allowed to reduce precision errors on measured temperatures. The smoothing process is based on the randomness requirements of the smoothing. That is through a check on skewness and kurtosis of smoothed measurements the code verifies that only random phenomena are smoothed.

Since one of the goal of the thesis was to understand the source of heat flux profile oscillations over z, a post smoothing of converged heat flux over z was performed. Then new temperatures obtained from the application of smoothed heat flux were compared to the ones obtained from the application of unsmoothed converged heat flux. Temperature differences look big at a first view, because they are much bigger than thermocouples accuracy. But then, after estimating errors on heat flux coming from wrong positioning of thermocouples it has been noted that smoothed heat fluxes over z were perfectly included in error bars from wrong positioning. Furthermore, also the temperature differences between the ones obtained from unsmoothed converged heat flux and the ones obtained from smoothed heat flux were comparable to errors in temperatures due to wrong positioning of thermocouples. This is true for oscillations away from the face-plate. Oscillations near the faceplate can not be explained by wrong positioning and so they could be physical and due to recirculation.

RoqFITT was validated using data from some computational experiments in which heat fluxes were known. In particular, since one of the hypotesis to explain oscillations over z was the effect of the igniter (which works for 0.5 s at the beginning of the combustion) the applied heat fluxes in these simulated experiments have a big oscillation over z in the first 0.5 s and it suddenly disappears after 0.5 s. Here, the input of the inverse problem are the temperatures coming from the simulated experiments (perturbed using a $\sigma = 0.3$ K). Inverse converged heat fluxes show a good agreement with the real ones. This demonstrates that oscillations are not introduced by the code itself.
Another validation has been performed setting a simplified inverse method based on the discretization of energy equation in a simplified domain. Also in this case, heat flux oscillations over z are still present, although the method used to evaluate heat flux is completely different.
The heat sink boundary condition has also been validated using different heat sink lengths. Results have shown that the physics of the problem requires the presence of an heat sink (if heat sink is not present measured temperatures don't match computational ones) and also show that heat sink length influence heat flux only near the faceplate and for heat sink length > 0.172 m the converged heat flux are almost constant with respect to a change in heat sink length.

The thesis main goal was to estimate the errors on heat flux computed by RoqFITT. Sources of errors examinated are precision and accuracy of thermocouples; wrong positioning of thermocouples; wrong material properties input. Errors on temperatures are converted in errors on heat flux thanks to a sensitivity analysis.

Errors estimations show that errors on heat flux due to thermocouples accuracy and precision are included in reasonable limits and they are not certainly the cause of heat flux oscillations over z.
Errors on heat flux coming from wrong positioning of thermocouples are much bigger than errors from precision and accuracy and they could be able to explain oscillations over z. A further proof that oscillations over z could depend on wrong information about thermocouples locations is that changing the nominal locations of some thermocouples and solving the inverse problem with these new locations oscillations disappear.
Errors from wrong material properties input are the same order of magnitude of errors from wrong thermocouples positioning, but this source of error can't be the cause of oscillations since heat fluxes computed using different material properties show only a shift but conserve their shape.
Once errors on heat flux are known it is possible, through a sensitivity analysis, to estimate errors on computed hot gas wall temperatures. Results from this test show that in the worst case (near the nozzle at the end of the experiments) errors on hot gas wall temperatures are nearly 20^oC.

It can be concluded that, although thermocouples are not a high accuracy instrument, they are very useful to estimate heat fluxes with a reasonable accuracy. But also thermocouples locations should be known with high accuracy due to high temperature gradients near the hot gas wall (a wrong positioning of 0.5 mm could lead to an error in temperatures up to 8^oC). If thermocouples locations are not known with high accuracy some fake oscillations over z could appear from the solution of the inverse problem. Also the knowledge of material properties is very important to estimate heat flux coming from the combustion with accuracy. Errors on material properties under 10 % are acceptable.

Bibliography

[1] http://aidanlyon.com/aidanlyon.com/media/publications/Lyon-normal_distributions.pdf

[2] Anderson, J.D.Jr: *Computational Fluid Dynamics , The Basics with Applications*. McGraw-Hill, 1995

[3] Celano, M. P. ; Silvestri, S. ; Pauw, J. ; Perakis, N. ; Schily, F. ; Suslov, D. ; Haidn, O J.: Heat Flux Evaluation Methods for a Single Element Heat-Sink Chamber, EUCASS 2015

[4] Celano, M. P. ; Silvestri, S. ; Schlieben, G. ; Kirchberger, C. ; Haidn, O. J.: Injector Characterization for a GOX-GCH4 Single Element Combustion Chamber. (2013)

[5] Chung, Julianne ; Saibaba, Arvind K. ; Brown, Matthew ; Westman, Erik: Efficient generalized Golub Kahan based methods for dynamic inverse problems. In: *IOP* 024005 (2018), Nr. 34

[6] https://www.thermocoupleinfo.com/type-t-thermocouple.htm

[7] Frohlich, A. ; M.Popp ; G.Schmidt ; D.Thelmann: Heat transfer characteristics of H2/O2 combustion chambers. Monterey, CA, USA : IAA 93-1826, 29th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, June 28th to 30th, 1993

[8] Goldmana, A. ; Kao, Y. C.: Numerical Solution to a Two-dimensional Conduction Problem Using Rectangular and Cylindrical Body-Fitted Coordinate Systems. In: *J. Heat Tranrfer* 103 (1981), Nr. 753-758

[9] Haidn, O. J.: Advanced Rocket Engines, Advances on Propulsion Technology for High Speed Aircraf, 2008

[10] Incropera, Frank P. ; Dewitt, David P. ; Bergman, Therodore L. ; Lavine, Adrienne S.: *Fundamentals of Heat and Mass Transfer*. 6th. John Wiley and Sons, 2007

[11] Iqbal, M.J. ; Sheikh, N.A. ; Ali, H.M. ; Khushnood1, S. ; Arif1, M.: Comparison of empirical correlations for the estimation of conjugate heat transfer in a thrust chamber, Life science journal, 2012

[12] Leonardi, M. ; Nasuti, F. ; Onofri, M.: Basic Analysis of a LOX / Methane Expander Bleed Engine, EUCASS 2017

[13] Monegato, G.: *Metodi e algoritmi per il Calcolo Numerico*. CLUT, 2008

[14] Oschwald, M. ; Suslov, D. ; Haemisch, J. ; Haidn, O. ; Celano, M. ; Kirchberger, C. ; Rackemann, N. ; Preuss, A. ; Wiedmann, D.: Measurement of Heat Transfer in Liquid Rocket Engines

[15] Ozisik, M. N. ; Orlande, H. R. B.: *Inverse Heat Transfer: Fundamentals and Applications*. Taylor and Francis, 2000

[16] Pauw, J. D.: Master degree thesis, Inverse Heat Conduction Problems applied to Rocket Combustion Chambers, Technische Universität München (TUM), 2015

[17] `http://virgo-physics.sas.upenn.edu/uglabs/lab_manual/Error_Analysis.pdf`

[18] Perakis, N.: Bachelor degree thesis,Inverse method applied to rocket engine, Technische Universitat Munchen (TUM), 2014

[19] Perakis, N.: Master degree thesis, Flamelet Modeling and Simulation of CH4/O2 Rocket Thrust Chambers, Technische Universität München (TUM), 2016

[20] Sutton, G. P. ; Biblarz, O.: *Rocket Propulsion Elements*. 7th. John Wiley and Sons, 2001

[21] Taylor, J.R.: *An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements.* University Science Books, 1999