

ECOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

LABORATORY OF RENEWABLE ENERGY SCIENCES AND ENGINEERING

SUPERVISOR : PR. HAUSSENER SOPHIA

ASSISTANT : MATTHIEU JONIN

---

# Hydrogen fast-filling modeling, optimisation and control

---

Author : Cédric FONTAINE

MARCH 2021

**EPFL**

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Literature review</b>	<b>2</b>
2.1	CFD studies . . . . .	2
2.2	Analytical models . . . . .	3
2.3	Optimisation and control oriented studies . . . . .	4
<b>3</b>	<b>Objectives</b>	<b>5</b>
<b>4</b>	<b>Physical modeling</b>	<b>6</b>
4.1	Physics . . . . .	6
4.1.1	Joule-Thomson expansion . . . . .	6
4.2	System modeling . . . . .	7
4.2.1	Conservation equations . . . . .	7
4.2.2	State equations . . . . .	9
4.2.3	Heat transfer model . . . . .	9
4.2.4	Dynamics with heat transfer . . . . .	11
4.3	Hydrogen isentropic expansion . . . . .	12
4.4	Model validation . . . . .	12
<b>5</b>	<b>Optimal control numerical formulation</b>	<b>14</b>
5.1	Optimal control theoretical background . . . . .	14
5.2	Multiple shooting approach . . . . .	15
5.2.1	Implementation of the NIST database . . . . .	16
5.3	First implementation using the Matlab <sup>®</sup> solver <code>fmincon</code> . . . . .	18
5.3.1	A first optimisation procedure . . . . .	18
5.3.2	Computation speed issue . . . . .	18
5.4	Second implementation with Casadi . . . . .	19
5.4.1	RK4 sensitivity analysis . . . . .	19
<b>6</b>	<b>Optimal control problem formulation and results</b>	<b>21</b>
6.1	Free control input . . . . .	21
6.1.1	Free control input I . . . . .	21
6.1.2	Result : an optimal discrete-time control law . . . . .	22
6.1.3	Free control input II . . . . .	24
6.2	Optimal PID controller . . . . .	26
6.2.1	OCP statement . . . . .	26
6.2.2	Results for a single optimal PID controller . . . . .	28
6.2.3	Adding a second controller . . . . .	29
6.2.4	Simulations . . . . .	30
6.3	Adaptive optimal PID controller (APID) . . . . .	30
6.3.1	Parametric uncertainties . . . . .	31
6.3.2	Optimal adaptive PID controller . . . . .	32
6.3.3	Adaptive I-PID controller training . . . . .	34
6.3.4	Adaptive I-PID controller simulations . . . . .	35
6.4	Discussion and further improvements . . . . .	38
<b>7</b>	<b>Conclusion</b>	<b>39</b>
	<b>References</b>	<b>41</b>

## 1 Introduction

Hydrogen energy is widely considered a green alternative to fossil fuels and could provide a sustainable source of energy while also contributing to overcome environmental issues, such as green house gas emissions. The main benefit of hydrogen energy consists in its green production since it can be synthesised using renewable energy only, such as solar energy. Furthermore, hydrogen combustion generates water vapor as its only reaction product, making fossil fuels look even more out-of-date.

Although being equipped with such advantages, one sizable pitfall subsists, best described with one word : storage. As a matter of fact, hydrogen precisely is the single lightest gas on earth. This is an issue since almost any application where the use of hydrogen energy could bring a valuable contribution righteously requires compact storage devices, such as automobile or aircraft propulsion, household powering or plant energy supply. Until now, two storage modes have been predominantly employed : liquified hydrogen or pressurised gaseous hydrogen storage. On the one hand, liquid hydrogen involves maintaining the storage temperature under its critical point, lying at  $-252.87$  [°C], implying burdensome constraints. On the other hand, high pressure storage enables convenient ambient temperature storage, with the downfall of necessitating highly resisting pressure vessels. Nevertheless, high pressure storage is generally preferred for applications where space is restricted.

The context of the present work settles within the *Solar Hydrogen for Cooking in the Global South* project from the Laboratory of Renewable Energy Sciences and Engineering (LRESE) which aims at providing a sustainable, health-preserving energy source for household cooking. On-site produced hydrogen is then stored in gaseous form within Composite Overwrapped Pressure Vessels (COPV) which are used to fill household-intended tanks. However, hydrogen heats up while relaxing from the high pressure seasonal storage tank, which will in turn affect the physical properties of the pressure vessels in regards to their limited temperature resistance. This is a major issue for it would cause irrevocable damages to the filling station and possibly lead to hazardous system operation. Indeed, several filling stations have already been subjected to serious harms and even explosions due to the hydrogen extremely high flammability. Nonetheless, this issue can be mitigated by performing an online control of the filling process.

Therefore, works aims at demonstrating the possibility of using a simple, well-suited for the task controller for optimising while restricting the filling of a COPV. To that end, a versatile modelisation of the system will be carried out, after which a wide-purposed optimisation problem will be stated. Several control schemes will then be proposed in order to address the different challenges which naturally arise from such a modelisation.

## 2 Literature review

A wide range of research has been actively conducted during the past decade, which probably comes from the rising interest about renewable and sustainable energy in the likes of hydrogen. A huge portion of the research was concerned about characterising the temperature rise which occurs during hydrogen tank fast filling. The preferred method for studying these systems consists in performing Computational Fluid Dynamics (CFD) analyses since it generally enables direct comparisons with experimental data, due to the precise geometry modeling. Hence, the major CFD-oriented papers will be presented in the first instance. Then, some distinct and worthwhile features of the numerous papers investigating analytical models will be discussed. This will establish a modeling basis for this work, introducing some of the assumptions performed thereafter. Ultimately, articles treating the problem from the standpoint of constrained optimisation will be specified, although there are not many of them.

### 2.1 CFD studies

First of all, Heitsch et al. [1] have investigated the effect of hydrogen fast-filling for a 74 [L] tank to 350 [bar], which was also physically available in the aim of comparing simulation and experimental data. Although starting the filling with a 100 [bar] pressure allows a relatively fast filling, yielding almost no temperature stratification within the tank, they showed that performing a complete tank filling from 10 [bar] might be problematic since distinct stratification might occur, resulting in lower heat transfers. Next, Galassi et al. [2] have inspected the impact of several types of filling on the temperature rise in a 70 [bar] Type IV tank. They first concluded that assuming an adiabatic process can result in almost 70°C difference for the gas temperature when compared to a simulation with full heat transfer incorporated.

Thus, they definitely discarded this assumption. In a second test, they also emphasised that the best option regarding the pressure rise rate might be to consider a relatively linear one since huge pressure gradients leads to abrupt temperature rises too. Finally, they mention that implementing a pre-cooling process would allow a smaller final gas pressure, which also diminishes the final gas temperature for the same final mass. Then, Galassi et al. [3] have performed CFD tank lifecycle testing, also contrasting their results with experiments. They assert that the primary factors influencing the temperature rise are the mass flow rate and the initial tank pressure, ascertaining that filling an almost empty tank gives rise to the highest temperatures, which can be considered as a worst case scenario.

Still on the CFD side, Melideo et al. [4] released a noteworthy paper on fast hydrogen filling from very low to very high pressures (from 20 [bar] up to 770 [bar]), examining the effect of pre-cooling with CFD tools as well as experimental data. In the end, they showed that reducing the inlet gas temperature from 15 [°C] to 0 [°C] would achieve a final temperature 10 [°C] lower. Also, pre-cooling the tank inlet down to -40 [°C] results in an almost 40 [°C] cooler temperature at the final pressure, hence demonstrating the huge benefits which come along with pre-cooling. Furthermore, this has an additional impact on the tank state of charge (SoC) since a higher final temperature yields a lower gas density for a given pressure. Several researchers from the same team continued their work on pre-cooling in Melideo and Baraldi [5] and examined the different strategies that would allow the least pre-cooling energy consumption while ensuring the secure operation of the tank filling and a maximised SoC. Indeed, they showed that it is not necessary to pre-cool hydrogen for the whole filling time and that partial-time pre-cooling can allow a greater combined performance.

Finally, Liu et al. [6] investigated the temperature rise for the two main types of tank, namely the Type III and Type IV tanks<sup>1</sup>, respectively made from linings of aluminium and plastic, both overwrapped with CFRP (Carbon Fiber Reinforced Plastic). They showed that Type IV tanks yield a 20 [°C] higher final gas temperature compared to Type III tanks, as a result of their much lower

---

<sup>1</sup>See COPV constructor site at <https://www.infinitecomposites.com/composite-pressure-vessel-resources>

thermal conductivity. Despite benefiting from sizable advantages, seasonal high pressure hydrogen storage involves many high-pressure resisting tanks, which can become costly. Hence, installing a compressor in line with low pressure seasonal storage tanks may solve this problem, which might lead to higher gas temperatures, as shown in Li et al. [7].

## 2.2 Analytical models

Along with these numerical-focused researches, several paper have exclusively elaborated theoretical and analytical models in the aim of better understanding the physical phenomenons which arise during a fast hydrogen filling process. Among those, Wang et al. [8] implemented a thermodynamic 0D model taking into account a varying heat transfer coefficient, depending on the hydrogen mass flow rate. They also compared simulations performed with constant mass-flow rate and heat transfer, mass-flow rate dependant heat transfer and SoC dependant heat transfer. The results agreed very well to experimental data, derived from a 29 [L] Type IV tank filling. Yet, SoC-varying heat transfers appear as being the closest model to the real-life situation, while a constant heat transfer coefficient shows a clear temperature over-estimation.

Further investigating heat transfer correlations, Woodfield et al. [9] derived a heat transfer correlation based on an averaged Nusselt number over the tank wall, yielding a global heat transfer model. The correlation accuracy was improved by contrasting their results with experimental data, allowing them to determine extra coefficients by fitting their correlation onto these.

Although the correlations found for the in-tank forced convection heat transfer yield accurate results, Guo et al. [10] used a constant  $h_{in} = 500 [W/m^2K]$  heat transfer coefficient and retrieved results very close to both their own experimental data and those from the JARI<sup>2</sup> test facility. This may actually be caused by their heat transfer coefficient value being based on a renowned article, Woodfield et al. [12]. Its authors exclusively studied what constant values can be assumed for the internal heat transfer coefficient.

Also, two of their findings are particularly interesting for the modeling aspect of the tanks. Firstly, considering a constant supply tank temperature yields a clear temperature overestimation. This is due to the supplied enthalpy being much greater since the supply tank temperature should decrease while it is emptying. Secondly, considering a purely cylindrical tank underestimate the overall tank heat capacity because both tank extremities are covered with massive metal domes<sup>3</sup>. This in turn leads to a gas and tank temperature overprediction. Likewise, they also demonstrated that the natural convection heat transfer coefficient had almost no effect on the temperature rise of Type III tanks, due to the very low thermal conductivity of CFRP. In Khan et al. [13], the authors used heat transfer coefficients derived from the previously presented paper and observed that their model predictions yielded a better accuracy for short filling durations (60 [s]), while longer ones (120 [s]) let appear discrepancies of about 15 degrees when comparing both.

However, heat transfers across the tank linings can be formulated in various ways, such as the 2-temperature, 4-temperature or 1D semi-infinite wall models which have been formulated in Ruffio et al. [14]. Both the 1D and 4-temperature models reportedly yield accurate heat transfer transient values. Another significant aspect of fast-filling modeling consists in choosing an appropriate equations of state. Hence, they also compare analytical results obtained with respectively the ideal gas law, the Van Der Waals law as well as the Redlich-Kwong law, leading to the conclusion that the latter yields better results when comparing with an accurate thermodynamics database.

<sup>2</sup>Indeed, accurate and thorough experiments regarding hydrogen tank burst and fast-filling were performed at the Japan Automotive Research Institute (JARI) in Ooi et al. [11], enabling very informative analytical model validation tests.

<sup>3</sup>Indeed, a CFRP overwrapped tank cannot be entirely covered with composite material since piping connections must be placed on the tank ends (or caps). Those have to be made of metal, thus requiring a higher material thickness at these places. Since metal (more precisely aluminium the case of Type III tanks) has a fairly high specific heat capacity, this amounts to a overall higher tank heat capacity.

In the aim of improving the model accuracy, Woodfield and Monde [15] also, modeled the supply tank thermodynamics, connecting both tanks with an isenthalpic expansion due to the Joule-Thomson effect. By feeding the inlet pressure history from experimental measurements in their model, they were able to retrieve relatively precise output values such as the storage tank mass and temperature evolutions, despite their model being quite basic. In the same vein, Deymi-Dashtebayaz et al. [16] realized a complete modeling of the filling process, this time also including flow equations for the purposes of taking the hydrogen turbulence into account, as well as the choking effect caused by the high in-pipe velocities. This led to equally conclusive results when using the National Institute of Standards and Technology (NIST) as a reference. Nevertheless, one of the most analytically comprehensive study regarding hydrogen fast-filling systems can be found in Olmos and Manousiouthakis [17]. Throughout their work, a generic cubic equation of state is used in pair with ideal gas representations for specific heat capacities and a lumped parameter model for a multi-tank system. They draw the interesting conclusion that if several tanks have to be filled sequentially for a large supply vessel, the next tank will benefit from the fill-up memory effect, which can be attributed to the supply vessel temperature decrease each time a new tank is filled-up.

### 2.3 Optimisation and control oriented studies

However important these numerical and analytical considerations might be, these researches lack a whole aspect of the hydrogen fast-filling systems : how can one control these systems and what might be the optimal control method. Hence, the first study focusing its attention on controlling hydrogen fast-filling systems was Pregassame et al. [18]. Indeed, they investigated what were the main risks constraining such a process, and mainly identified three of them : over-pressure, over-heating and over-filling of the tank. As a matter of fact, the main issue in fast-filling systems is inherent to their rapid gas expansion, which causes steep temperature rises and require a higher pressure to obtain the same mass at a higher temperature. Hence, this causes over-pressure. Over-filling can conversely occur when the ambient temperature is low and the traditional pressure overshoot leads to a mass overshoot. A more specific approach is undertaken in Xian Wu et al. [19], where a time-delayed method is used to prevent sharp temperature rises. They show that establishing two 15 [s] filling pauses respectively after 30 [s] and 85 [s] of filling would yield a 62 % reduction in total filling time while conforming to the temperature limits.

Now, several studies also sought at determining optimal filling laws for such tanks using optimisation techniques. One of them, Zhenga et al. [20], investigated how one could optimise the filling time for a 3-stage filling station with respectively high, medium and low pressure supply tanks. In the same vein, Bai et al. [21] implemented an optimisation algorithm in the aim of specifying the optimal switching pressure coefficients for a cascade filling system. At the end of the day, Olmos and Manousiouthakis [22] formulated an optimal control problem for minimum-time control and were able to solve it analytically for a global solution by using appropriate assumptions. Since they considered an adiabatic filling process, they could investigate optimal control law profiles enabling reaching the desired mass while satisfying temperature constraints without the usual pressure overshoot strategy. As a matter of fact, preventing heat from leaving the tank would not allow pressure to overshoot and then decrease back, benefiting from heat transfers.

In light of the abundant literature on the subject, one can already tell that practical control issues do not appear to have been tackled. However, the physical system at hand in this work will be bestowed with tighter constraints than most of those studied so far, as will be further explained later. Therefore, the focus will be brought on the determination of an adequate control method for an equally safe and rapid filling.

### 3 Objectives

The main goal of this work consists in designing an optimal controller for a hydrogen filling system. As mentioned before, gaseous hydrogen will heat up while expanding from a high pressure tank into a lower one. This temperature rise could eventually entail harmful effects for the tank materials, thus causing safety issues. Hence, the system has to be controlled in a way that prevents it from breaking some hard constraints, articulated as follows :

- The tank wall temperature should not exceed  $65 [^{\circ}\text{C}]$  or  $338 [K]$
- The tank operating pressure is  $300 [\text{bar}]$
- The tank maximum pressure is  $330 [\text{bar}]$
- The tank final mass must be  $267 [g]$ , i.e. the mass necessary for an operating pressure of  $300 [\text{bar}]$  at ambient temperature ( $298 [K]$ )

The system at hand is schematised in Figure 3.1.

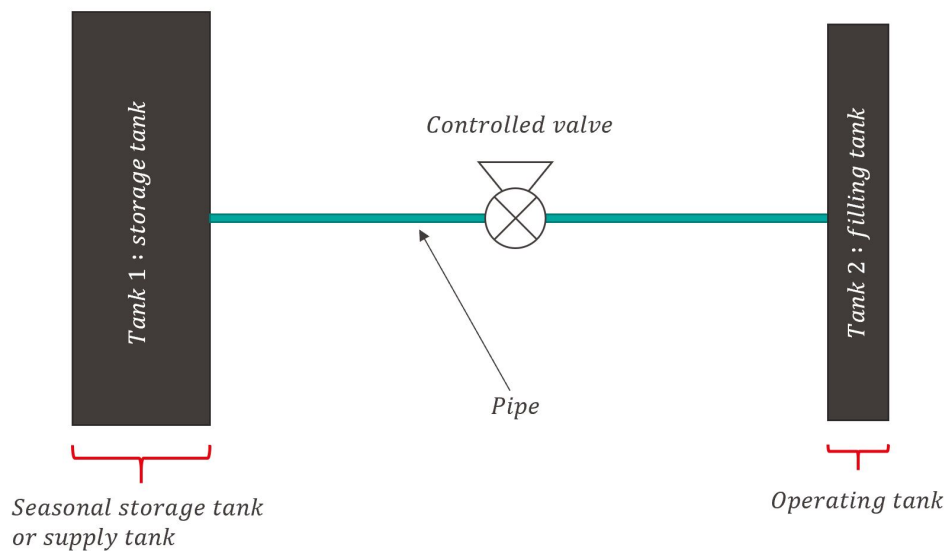


Figure 3.1: System case-study for this work

The seasonal storage or supply tank namely tank 1, is a  $500 [L]$ ,  $500 [\text{bar}]$  COPV which is directly connected to an operating tank, namely tank 2, which internal volume is  $13 [L]$ . In between both tanks stands a flow-limiting valve, which is assumed to be controlled. Since the physical system characteristics were not entirely defined at the beginning of this work, the valve stepper motor dynamics and the actual controller implementation are not part of this project. The aim consists in designing an efficient control law which shall be fully parametrised, i.e. one must be able to easily modify physical parameters and launch an optimisation problem to get an appropriate control law. A little piece of software will thus be written and explained in the annexes of this work.

## 4 Physical modeling

### 4.1 Physics

The initial physical system that we need to model consists in a gas expansion through a pipe, between two tanks at different pressures. In the following, we will use the term *tank 1* to refer to the seasonal supply tank and *tank 2* for the tank to be filled, respectively on the left and on the right of Figure 4.1.

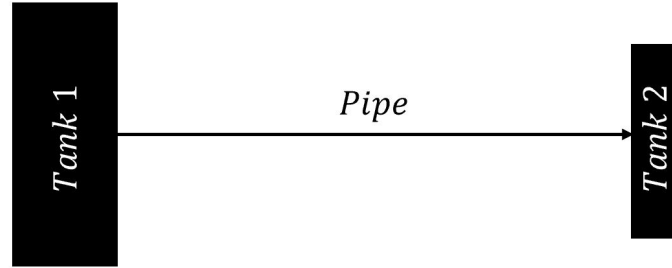


Figure 4.1: Physical system considered

#### 4.1.1 Joule-Thomson expansion

The thermodynamic process which takes place within the pipe can be modeled as a *Joule - Thomson expansion*. This assumes that the gaseous  $H_2$  follows an isenthalpic, adiabatic expansion between both tanks, as presented in W.G.Kortekaas et al. [23]. This expansion is characterized by a *J-T coefficient*, as shown in Equation 4.1.

$$\mu_{J-T} = \left\{ \frac{\partial T}{\partial P} \right\}_H = \frac{V(T\alpha - 1)}{C_p} \left[ \frac{K}{Pa} \right], \quad \alpha = \frac{1}{V} \left\{ \frac{\partial V}{\partial T} \right\}_P \quad (4.1)$$

We know that an ideal gas subjected to such an expansion would yield  $dh = V(1 - T\alpha)dP + C_p dT = V(1 - T\frac{1}{T})dP + C_p dT = 0$ . Hence, the work produced would be zero and there would be no change in the fluid internal energy, yielding a constant temperature. However, as reported in Pratt [24], Joule and Lord Kelvin already experimentally showed in 1854 that hydrogen warms under expansion from the temperature of  $-80.5$  [°C], making the ideal gas assumption incorrect. Hence, a real gas model must be used. The *Joule - Thomson coefficient* provides an insight on the variation of temperature when pressure changes. When flowing from a higher to a lower pressure region (e.g. from *tank 1* to *tank 2*) through an insulated pipe, the gaseous hydrogen will experience an isenthalpic irreversible expansion, yielding  $dH = dU + d(PV) = 0$ . The  $PV$  term represents the inter-atomic work (which is zero for an ideal gas) done by the gas while expanding, yielding an increase or a decrease in the internal energy  $U$ .

The *Joule - Thomson coefficient* has what is called an inversion point, for every gas. This means that  $\mu_{J-T}$  switches from a positive, yielding a drop in temperature when pressure decreases, to a negative value, where the inverse effect happens. However, it is well known that this inversion occurs at very low temperatures for hydrogen, at about  $200$  [K], as shown on Figure 4.2. Hence, for typical hydrogen filling systems working around ambient conditions,  $\mu_{J-T} < 0$ , which leads to an increase in temperature while it expands.

<sup>4</sup>[https://en.wikipedia.org/wiki/Joule%E2%80%93Thomson\\_effect](https://en.wikipedia.org/wiki/Joule%E2%80%93Thomson_effect)



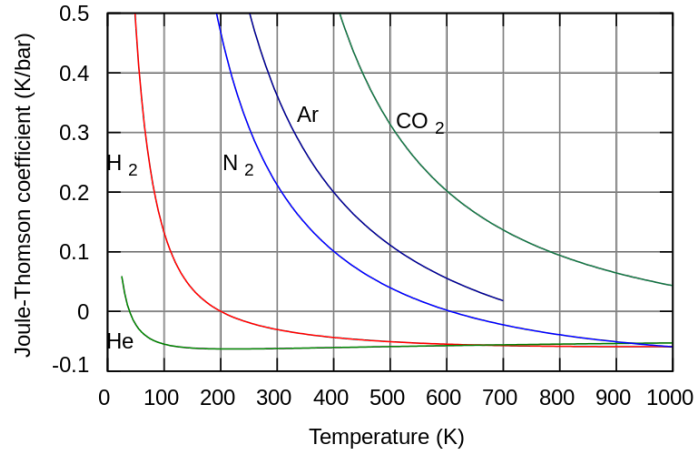


Figure 4.2: Temperature dependence of the *Joule-Thomson coefficient*<sup>4</sup>

## 4.2 System modeling

### 4.2.1 Conservation equations

The foundations of this system model consist in the *mass conservation law* and the *energy conservation law for open systems*. Tank 1 and tank 2 can be considered as being two energy storage systems with respectively one output and one input. Thus, the coupled dynamics must be solved for simultaneously for both tank 1 and tank 2, whose dynamics equations are very similar. First, let's consider the energy balance.

$$\frac{\partial}{\partial t}(m_{\text{inside}}) = \dot{m}_{\text{IN/OUT}} \quad (4.2)$$

The objective consists in filling tank 2 with gaseous hydrogen stored in tank 1. Therefore, it can be assumed that the control input  $u(t)$  [kg/s] is the mass-flow rate flowing between both tanks. From these simple considerations, the mass balance writes down as follows :

$$\frac{\partial}{\partial t}(m_1) = -u \quad (4.3) \quad \frac{\partial}{\partial t}(m_2) = u \quad (4.4)$$

The minus sign in Equation 4.3 indicates that hydrogen is flowing out of tank 1. Next, two energy balances are performed according to the 1<sup>st</sup> law of thermodynamics. In its most general form, it is expressed as in Equation 4.5,  $N_i$  and  $N_o$  respectively being the number of inputs and outputs :

$$\frac{\partial}{\partial t}(E) = \sum_{i=1}^{N_i} \dot{m}_i \left( \frac{1}{2} v_i^2 + h_i + g z_i \right) - \sum_{o=1}^{N_o} \dot{m}_o \left( \frac{1}{2} v_o^2 + h_{OUT} + g z_o \right) + \dot{Q} - \dot{W} \quad (4.5)$$

The term on the left-hand side of Equation 4.5 represents the total variation of energy within a certain control volume CV, and can be expressed as  $\frac{\partial}{\partial t}(E) = \frac{\partial}{\partial t}\{m(u + \frac{v^2}{2} + g z)\}$ . Since changes in potential energy can be considered as negligible and the kinetic energy of the gas inside the tank is assumed to be very low compared to that of the gas entering into it, this simplifies to  $\frac{\partial}{\partial t}(E) = \frac{\partial}{\partial t}\{m u\}$ . The gas specific internal energy  $u$  [J/kg] is an intensive property of the gas, representing the microscopic movements among molecules of a substance, contrasting with the gas kinetic and potential energies, both being macroscopic forms of energy.

In both models for tank 1 and 2, changes in potential energy  $gz$  will be neglected, considering that there is no work provided by or to the system and that only heat transfers occur in tank 2. Indeed, the storage tank can be assumed to be considerably larger than tank 2 and at ambient temperature, leading to negligible heat transfers due to small temperature changes. An expression for the heat transfer happening in tank 2  $\dot{Q}_2$  will be provided in Sec. 4.2.3. The energy balances for our system are stated in Equation 4.6 and 4.7.

$$\frac{\partial}{\partial t}(m_1 u_1) = -\dot{m}_1 \left( \frac{1}{2} v_1^2 + h_1 \right) \quad (4.6) \quad \frac{\partial}{\partial t}(m_2 u_2) = \dot{m}_2 \left( \frac{1}{2} v_1^2 + h_2 \right) + \dot{Q}_2 \quad (4.7)$$

Since both  $m_1$  and  $m_2$  have a time dependence, we can re-express the energy balance as follows :

$$\begin{cases} \frac{\partial}{\partial t}(u_1) = \frac{1}{m_1} \left( -\dot{m}_1 \left( \frac{1}{2} v_1^2 + h_1 \right) - \dot{m}_1 u_1 \right) \\ \frac{\partial}{\partial t}(u_2) = \frac{1}{m_2} \left( \dot{m}_2 \left( \frac{1}{2} v_2^2 + h_2 \right) + \dot{Q} - \dot{m}_2 u_2 \right) \end{cases} \quad (4.8)$$

We now introduce the states, presented in Tab. 4.1, and our dynamics equations in Equation 4.9.

Table 4.1: System states and input

State	Physical meaning	Units
$x_1$	Tank 2 mass, $m_2$	$[kg]$
$x_2$	Tank 2 specific internal energy, $u_2$	$\left[\frac{J}{kg}\right]$
$x_3$	Tank 1 mass, $m_1$	$[kg]$
$x_4$	Tank 1 specific internal energy, $u_1$	$\left[\frac{J}{kg}\right]$
$u$	Control input, $\dot{m}_1 = \dot{m}_2$	$\left[\frac{kg}{s}\right]$

$$\begin{cases} \dot{x}_1 = u \\ \dot{x}_2 = \frac{1}{x_1} \left( u \left( \frac{1}{2} v_2^2 + h_2 \right) - u x_2 + \dot{Q} \right) \\ \dot{x}_3 = -u \\ \dot{x}_4 = \frac{1}{x_3} \left( -u \left( \frac{1}{2} v_1^2 + h_1 \right) - u x_4 \right) \end{cases} \quad (4.9)$$

As an effect of the *Joule-Thomson* expansion, the flow is isenthalpic. This yields  $h_2 = h_1$ , or  $dh = 0$ . Also, conservation of mass implies that the velocity of the gas entering *tank 2* is the same as the velocity of the gas flowing out of *tank 1*, reducing to  $v_2 = v_1$ . Hence, for a fixed pipe diameter, Equation 4.10 gives the fluid velocity when entering tank 2.

$$v_1 = v_2 = \frac{u}{\rho_{pipe} A_{pipe}} \left[ \frac{m}{s} \right] \quad (4.10)$$

The main benefit which comes with the modeling of tank 1 dynamics consists in explicitly specifying the amount of specific enthalpy entering tank 2. As a matter of fact, Woodfield et al. [12] showed that taking this into account would avoid over-estimating tank 2 temperature, since specific enthalpy should decrease with tank 1 temperature, which in turn should decrease while it is emptying.

### 4.2.2 State equations

At high pressure, hydrogen cannot be modeled as an ideal gas, since inter-molecular forces are not negligible. Hence, two options can be thought about. Either an appropriate state equation (EoS) is defined, such as the the Van Der Waals EoS or the Redlich Kwong EoS if a greater accuracy is desired. This would allow to have access to a closed-form solution for the temperature, as was shown in Olmos and Manousiouthakis [17] with a generic cubic EoS. The second option consists in using thermodynamic property database, which internally models the gas states with the most advanced and recent EoS available. Therefore, such databases will bring access to more accurate values and avoid having to implement highly non-linear EoS, which would amount to fairly elevated computational cost in a gradient-based optimization framework.

As referenced in Section 2.2, many researchers have focused on deriving analytical solutions for the hydrogen fast-filling problem. In Chen et al. [25], concern is brought onto finding a compact while accurate real-gas equation for hydrogen. The primary difficulty is indeed that the behaviour of hydrogen a very high pressures is highly different that what an ideal gas model would predict, thus compelling the researchers to formulate this EoS in other ways. Therefore, they propose a simplified real-gas EoS by fitting its coefficients on NIST data, providing only 1.3 % to 3.8 % error in comparison to the same data. Obviously, such errors can be prevented by directly using the NIST database. Moreover, Ruffio et al. [14] evaluated the respective performance of an ideal gas model as well as both the Van-Der-Waals and the Redlich-Kwong EoS, compared with a straightforward use of the NIST database. They concluded that both the Redlich-Kwong EoS and the NIST database yielded close results, further motivating the preference for the second option.

Although access to the latest version is not provided, the REFPROP V9.1<sup>5</sup> database from Lemmon et al. [26] can be freely used and enables the access of fluid and gas properties through the `refpropm` MATLAB<sup>®</sup> command. Among the advantages of such an implementation, the following can be highlighted :

- All desired properties can be easily accessed
- No assumption is needed regarding the independance of certain properties with respect to other. Only the density and the specific internal energy, which are the states of our dynamics, are needed to access other properties, such as the pressure, temperature, specific enthalpy or even specific heat ratio.
- Performance of such database will be better than a hand-made model. Indeed, all relevant EoS are already implemented within this database, and each property is computed according to the most precise one, until convergence.

However, the use of an external software causes several complications regarding its implementation within an optimization routine, as will be discussed in Sec. 5.3. Indeed, systematic evaluation of the database should be performed by gridding along discretized properties for each quantity of interest, yielding huge `.mat` tables.

### 4.2.3 Heat transfer model

Albeit quite conservative, the thermodynamic dynamic equations developed so far only take the increase in specific internal energy of tank 2 into account. Since the tanks considered here are likely to be made out of an aluminium liner and a carbon-fiber reinforced composite (CFRP) overwrap (Type III tanks), heat will flow out of these, especially from the second one due to sharp in-tank temperature rises. As stated in Section 3, the primary objective of this work consists in ensuring that the temperature of the CFRP overwrap stays below 338 [K] during the whole filling process. For now, only the gas temperature can be computed, which works fine since limiting this temperature would

<sup>5</sup>This database can be downloaded from <https://trc.nist.gov/refprop/LINKING/Linking.htm> GithubWebsite.

probably restrict the one of the tank lining. However, although the aluminium liner has a low thermal resistance, it happens that CFRP has a really low thermal conductivity, entailing a considerable heat accumulation.

As indicated in Section 2.2, many heat transfer models have been studied throughout the years, and each of those comes with several benefits and drawbacks. Nevertheless, Ruffio et al. [14] demonstrated that 4-temperature models yield a better accuracy for predicting the gas temperature as well as both wall materials temperatures. Then, several options exist regarding the internal forced heat transfer coefficients. One of them might consist in using a mixed forced and natural correlation, developed by fitting coefficients onto experimental data, presented in Woodfield et al. [9]. This yields the Nusselt correlation of Equation 4.11, applicable for  $3 \cdot 10^5 Ra_H < Ra_H < 4.2 \cdot 10^{11}$  and  $270 < Re_d < 1.2 \cdot 10^5$ .

$$Nu = 0.56 Re_d^{0.67} + 0.104 Ra_H^{0.352} \quad (4.11)$$

Another possibility would consist in using the empirical relationship derived in Wang et al. [8], which yields Equation 4.12 for a 19 [L] tank, *SOC* being the tank state of charge.

$$Nu = \frac{D_{in} \alpha_{in}}{\lambda_{gas}} = 1.6 \left( \frac{Re Pr}{SOC} \left[ \frac{d^2}{L_{in} D_{in}} \right] \right)^{0.632} \quad (4.12)$$

However, constant heat transfer coefficients will be used since it yields sufficient accuracy, as demonstrated in Woodfield et al. [12]. This will also enable the study of parametric variations of these parameters in Section 6.3.1.

**4-Temperature model** A 4-temperature heat transfer model shall be developed in order to investigate this effect, as shown on Figure 4.3. In order to do that, two new states shall be introduced as the aluminium liner and CFRP overwrap temperatures.

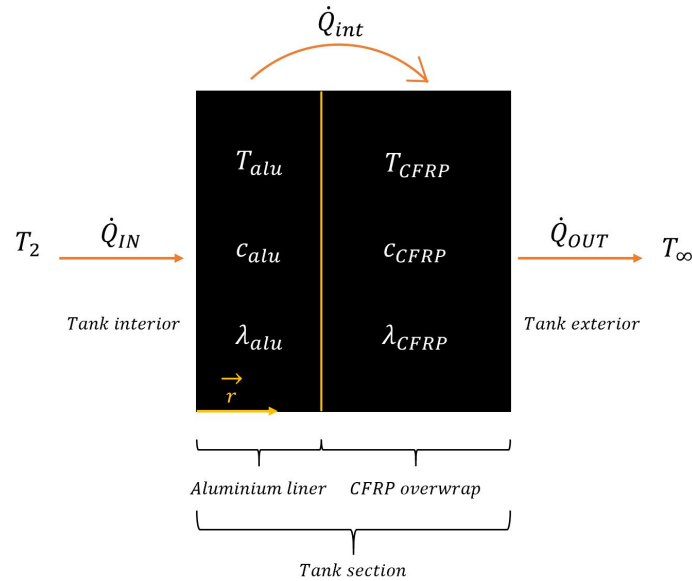


Figure 4.3: 4-temperature heat transfer model across the tank lining

By taking the lining thermal properties into account, one can use the heat flux balance within both material layers to introduce the time variation of the temperatures of interest, according to Equation 4.13.

$$\begin{cases} (m_{alu} c_{alu}) \frac{\partial}{\partial t}(T_{alu}) = UA_{IN}(T_2 - T_{alu}) - UA_{int}(T_{alu} - T_{CFRP}) \\ (m_{CFRP} c_{CFRP}) \frac{\partial}{\partial t}(T_{CFRP}) = UA_{int}(T_{alu} - T_{CFRP}) - UA_{OUT}(T_{CFRP} - T_{\infty}) \end{cases} \quad (4.13)$$

The parameters  $m$  and  $c$  respectively refer to the total mass [kg] and specific heat capacity [J/kgK], and the overall heat transfer coefficients [W/K] are defined in Equation 4.14, 4.15 and 4.16. All quantities of interest are displayed in Tab. 4.2.

$$UA_{IN} = \left( \frac{1}{2\pi R_1 L h_{IN}} + \frac{\ln(R_2/R_1)}{2\pi L \lambda_{alu}} \right)^{-1} \quad (4.14)$$

$$UA_{OUT} = \left( \frac{\ln(R_3/R_2)}{2\pi L \lambda_{CFRP}} + \frac{1}{2\pi R_3 L h_{OUT}} \right)^{-1} \quad (4.15)$$

$$UA_{int} = \left( \frac{\ln(R_2/R_1)}{2\pi L \lambda_{alu}} + \frac{\ln(R_3/R_2)}{2\pi L \lambda_{CFRP}} \right)^{-1} \quad (4.16)$$

Table 4.2: Overview of the different physical quantities at stake. Values either come from the literature (Deymi-Dashtebayaz et al. [16], Khan et al. [13]) or from webpages<sup>6</sup>

Physical quantity	Symbol	Value	Units
Specific heat capacity of aluminium	$c_{alu}$	896.06	[J/kgK]
Specific heat capacity of CFRP	$c_{CFRP}$	798.84	[J/kgK]
Aluminium mass	$m_{alu}$	6.31	[kg]
CFRP mass	$m_{CFRP}$	7.88	[kg]
Aluminium thermal conductivity	$\lambda_{alu}$	180	[W/mK]
CFRP thermal conductivity	$\lambda_{CFRP}$	0.55	[W/mK]
Inner forced convection coefficient	$h_{IN}$	250	[W <sup>2</sup> /mK]
Outer natural convection coefficient	$h_{OUT}$	4.5	[W <sup>2</sup> /mK]
Inner diameter	$R$	0.079	[m]
Aluminium thickness	$t_{alu}$	$5 \cdot 10^{-3}$	[m]
CFRP thickness	$t_{CFRP}$	$10 \cdot 10^{-3}$	[m]

#### 4.2.4 Dynamics with heat transfer

Now that equations modeling heat losses within the tank have been formulated, the augmented system dynamics can be stated. First, the  $\dot{Q}$  term in Equation 4.9 will take the form of Equation 4.17 :

$$\dot{Q} = UA_{IN}(T_2 - T_{alu}) \quad (4.17)$$

Then, defining two new states as  $x_5 = T_{alu}$  and  $x_6 = T_{CFRP}$ , the system represented by Equation 4.9 can easily be extended by appending a conveniently modified version of Equation 4.13, yielding Equation 4.18

<sup>6</sup>See <https://dielectricmfg.com/knowledge-base/aluminum/> or [https://www.substech.com/dokuwiki/doku.php?id=carbon\\_fiber\\_reinforced\\_polymer\\_composites](https://www.substech.com/dokuwiki/doku.php?id=carbon_fiber_reinforced_polymer_composites)

$$\begin{cases} \dot{x}_1 = u \\ \dot{x}_2 = \frac{1}{x_1} \left( u \left( \frac{1}{2} v_1^2 + h_1 \right) - u x_2 + \dot{Q} \right) \\ \dot{x}_3 = -u \\ \dot{x}_4 = \frac{1}{x_3} \left( -u \left( \frac{1}{2} v_1^2 + h_1 \right) - u x_4 \right) \\ \dot{x}_5 = \left( U A_{IN} (T_2 - x_5) - U A_{int} (x_5 - x_6) \right) \frac{1}{m_{alu} c_{alu}} \\ \dot{x}_6 = \left( U A_{int} (x_5 - x_6) - U A_{OUT} (x_6 - T_\infty) \right) \frac{1}{m_{CFRP} c_{CFRP}} \end{cases} \quad (4.18)$$

### 4.3 Hydrogen isentropic expansion

In the aim of eventually controlling the mass flow rate  $\dot{m} = u$ , it must first be ensured that it does not exceed the physical limitations of the present system. As a matter of fact, these are represented by the natural expansion which will occur within the pipe in case it is unobstructed. The aforementioned Joule-Thomson effect arising here prevents the use of an ideal gas model for the expansion. However, it can be assumed that heat transfer through the pipe wall is negligible compared to that in tank 2 since the gas only resides for a short amount of time within the pipes, as explained in Bai et al. [21]. Hence, an isentropic, i.e. both adiabatic and reversible, expansion model can be considered here, although keeping the non-ideality of the gas. This also relies on the extensive use of the NIST database, allowing one to easily compute the specific heat ratio of hydrogen. It is however to be noticed that diatomic gases have a relatively constant specific heat ratio, as explained in Cantwell [27], which could justify the use of an ideal gas expansion model, as used in Farzaneh-Gord et al. [28].

The mass flow rate bounding equation used throughout this work states as follows :

$$\dot{m} = C_d \sqrt{\gamma P_1 \rho_1} A_{pipe} \left( \frac{2}{\gamma + 1} \right)^{\frac{\gamma+1}{2(\gamma-1)}} \quad (4.19)$$

$C_d$  corresponds to the orifice discharge coefficient and  $\gamma$  is the specific heat ratio, computed according to the NIST tables. Hence, Equation 4.19 can be used to limit the mass flow rate provided by the controller in order to yield a more realistic simulation.

### 4.4 Model validation

Now that the dynamics have been specified comes the time for testing and validating. To perform that, it seems appropriate to compare the model at hand by using both experimental and analytical data found in the literature. Hence, data from Khan et al. [13] will be used since they contain exactly that in the aim of comparing two simulations. Both will consist in filling a 50 [L] tank with 350 [bar] of gaseous hydrogen, starting from the ambient temperature of 293 [K], but with different durations of 60 [s] and 120 [s]. According to the same study, the internal heat transfer coefficient be approximated to a constant 500 [W/m<sup>2</sup>K] during the whole filling, and decreased to a constant 250 [W/m<sup>2</sup>K] during the rest of the simulation while the outer heat transfer coefficient can be kept to 4.5 [W/m<sup>2</sup>K] since its influence on the gas and wall temperatures is limited. The results are presented on Figure 4.4.

On the one hand, it can be observed that the temperature peaks for both simulation durations are quite similar if compared with analytical data<sup>7</sup>, respectively amounting at 80 [°C] and 78 [°C] for

<sup>7</sup>Analytical data correspond to the "Gas Temp" of the Figure borrowed from Khan et al. [13], whereas the "JARI" (Japan Automotive Research Institute) data from Ooi et al. [11] correspond to experimental data.

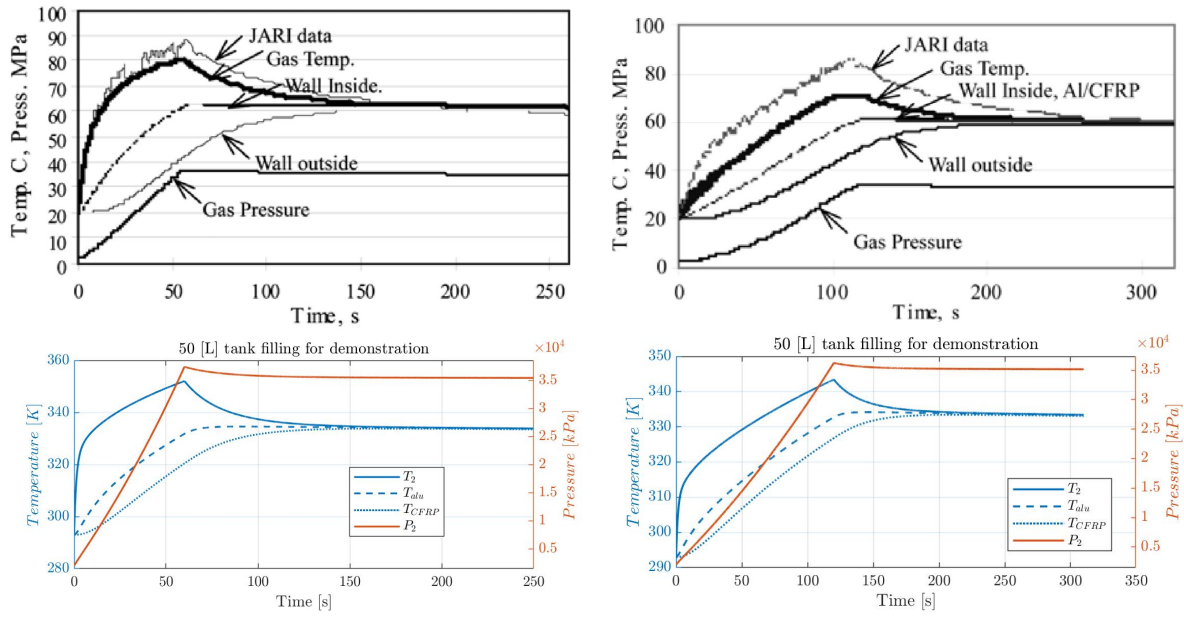


Figure 4.4: Simulation for 350 [bar] filling, comparing both 60 [s] and 120 [s] filling time

the 60 [s] filling time and at 70 [°C] and also 70 [°C] for the 120 [s] filling time. The steady-state stabilizing temperature also compares very well, reaching about 60 [°C] and 60 [°C] for the first case, and equally 60 [°C] and 60 [°C] for the second one. On the other hand however, although the comparison also holds fairly well for the 60 [s] filling duration, it leads to a quite frank underestimation of the overall temperature if we use the JARI experimental measurements.

This discrepancy could very well be caused by an overestimation of the heat transfer coefficient for the case where the filling rate has been reduced by a factor of 2. Therefore, this hints that heat transfers through the tank wall will play an essential role in the gas and wall temperature predictions. Also, a clear connection between filling speed and rate of heat transfer can be foresaw, letting sense that the heat transfer coefficient could very well be one of the most uncertain parameter of this model. As it will appear later, the overall heat transfer coefficients presented in Equations 4.14, 4.15 and 4.16 can be replaced by values found in the literature in order to have a more precise estimation of those. Also, an efficient manner of handling this uncertainty will be explained and demonstrated in Section 6.3.

## 5 Optimal control numerical formulation

Now that a formulation for our dynamics equations has been derived (see Equation 4.18), this analytical problem will be converted into a numerical one. In the aim of finding an optimal control law, the numerical discretisation methods widely used in *Optimal control theory* will be employed. First, this discretisation method will be formulated and then applied to the system at hand and a numerical solving scheme will be suggested. The procedure to implement the NIST database will then be explained and a time-step sensitivity analysis will be performed.

### 5.1 Optimal control theoretical background

A general case of optimal control problem is stated in Equation 5.1.

$$\begin{aligned} & \underset{u}{\text{minimize}} && \int_{t_0}^{t_1} \mathcal{J}(t, x(t), u(t)) dt \\ & \text{subject to} && \dot{x} = f(t, x(t), u(t)) \\ & && x_0 = x(t_0) \end{aligned} \tag{5.1}$$

The aim generally consists in seeking for the optimal input law  $u(t)$  such that it follows the system dynamics  $f(t, x(t), u(t))$  while minimising a certain control objective  $\mathcal{J}$ . Of course additional constraints and/or multiple objectives can be specified if desired, and this can be handled with different techniques. In general, two classes of methods can be used for solving such problems, as presented in Diehl and Gros [29].

**Indirect methods** These types of methods take advantage of the *Pontryagin Maximum Principle* to derive the necessary conditions to solve our optimal control problem. Those can be expressed from the Hamiltonian 5.2, which is then used to derive the three conditions of Equation 5.3, 5.4 and 5.5, respectively the *Optimality Condition*, the *Adjoint Condition* and the *Transversality Condition*. Indirect numerical methods integrate the adjoint equation with an initial value and then solve the optimality equation.

$$H(t, x, u, \lambda) = f(t, x, u) + \lambda g(t, x, u) \tag{5.2}$$

$$\frac{\partial H}{\partial u} = 0 \text{ at } u^* \implies \frac{\partial f}{\partial u} + \lambda \frac{\partial g}{\partial u} \tag{5.3}$$

$$\frac{\partial H}{\partial x} = -\lambda' \implies \lambda' = h(t, x, \lambda, u) - \left( \frac{\partial f}{\partial x} + \lambda \frac{\partial g}{\partial x} \right) \tag{5.4}$$

$$\lambda(0) = 0 \tag{5.5}$$

**Direct methods** Another approach to this problem consists in performing direct optimisation on an approximate version of the original OCP. To do that, a finite discretisation of the system dynamics has to be performed from the infinite dimensional decision variables space, such as  $x(t)$  or  $u(t)$ , which are continuous in time according to the formulation of Equation 5.1. Those iterative methods obviously lead to sub-optimal solutions, but are essential when the problem at hand is too complicated to be solved analytically. In this work, it happens to be the case, primarily because of the use of the NIST database, which yields analytically intractable dynamics.



## 5.2 Multiple shooting approach

In this work, a *multiple shooting approach* (MSA) from Bock and Plitt [30] will be carried out to solve the OCP. First, a relatively coarse gridding of the time horizon has to be performed into  $N$  time-intervals. Then, both the control input  $u(t)$  and state  $x(t, u)$  have to be discretised accordingly. This yields to  $N$  ODE starting at different initial conditions which must be solved in parallel in order for the respective final state of the  $i^{th}$  ODE  $x(t_{N_{sub}}^i)$  to match the variable initial condition of the  $(i+1)^{th}$  ODE, which can be expressed as  $x(t_{N_{sub}}^{i+1}) = m_{i+1}$  and is also stated by Equation 5.6. Hence, the main idea consists in defining as many initial conditions  $m_i$  as the number of ODEs, and then to sub-integrate from each of those initial conditions, making sure that the final sub-integration step matches the next initial condition  $m_{i+1}$ . Also, the objective has to be discretised according to the time grid defined before. The notation  $\mathcal{M}$  refers to a state and input vector gathered together according Equation 5.7. The reader can refer to An et al. [31] or Gander [32] for a deeper understanding of the method.

$$\begin{aligned} \dot{x}_i &= f(t, x_i(t), u_i) \quad t \in [t_i, t_{i+1}] \\ x_i(0) &= m_i \quad i \in [1 \dots N] \end{aligned} \quad (5.6)$$

$$\mathcal{M} = [x_{1,1} \dots x_{1,N} \ x_{2,1} \dots x_{2,N} \dots x_{3,1} \dots x_{3,N} \ x_{4,1} \dots x_{4,N} \ x_{5,1} \dots x_{5,N} \ x_{6,1} \dots x_{6,N} \ u_1 \dots u_N]^T \quad (5.7)$$

Therefore, a discretised version of the infinite-dimensional OCP of Equation 5.1 can be defined in Equation 5.8, with respectively discrete costs and initial conditions  $\mathcal{J}_i$  and  $m_i$ .

$$\begin{aligned} &\underset{m, u}{\text{minimize}} && \sum_{i=1}^N \mathcal{J}_i(m_i, u_i) \\ &\text{subject to} && m_1 - x_1 = 0 \\ & && m_{i+1} - x_i(t, m_i, u_i) = 0, \quad i = 1, \dots, N \\ & && g(m_i, u_i) \leq 0 \quad i = 1, \dots, N \\ & && x_N(t, m_N, u_N) = X_T \\ & && x_i \in \mathcal{X} \quad i = 1, \dots, N \\ & && u_i \in \mathcal{U} \quad i = 1, \dots, N \end{aligned} \quad (5.8)$$

Throughout this work, the discrete dynamics at hand,  $\dot{x}_i = f_i(t, m_i, u_i)$  are non-linear (see Equation 4.18) and shall thus be integrated in accordance with the time grid  $i \in [1, \dots, N]$  in order to enforce the continuity of the system dynamics. This will be performed with a Runge-Kutta 4 (RK4) integrator, which is recalled in Equation 5.9 and Algorithm 1 summarizes the logical step involved in the optimisation setup procedure. The primary advantage in using an RK4 integrator instead of a simpler Euler forward scheme is the truncation error, which is substantially smaller, respectively growing in  $\mathcal{O}(h^5)$  instead of  $\mathcal{O}(h^2)$  since RK4 will fix the first 4 derivatives, letting the error arise from the 5<sup>th</sup> one.

$$\begin{aligned}
k_1 &= f(x(k), u(k)) \\
k_2 &= f(x(k) + h \frac{k_1}{2}, u(k)) \\
k_3 &= f(x(t) + h \frac{k_2}{2}, u(k)) \\
k_4 &= f(x(t) + hk_3, u(k)) \\
f(k+1) &= f(k) + \frac{h}{6}(k_1 + \frac{k_2}{2} + \frac{k_3}{2} + k_4)
\end{aligned} \tag{5.9}$$

Also, each of the  $k_i$  explicitly depends on the dynamics parameters, which must thus be provided, although they are implicit in Equation 5.9.

---

**Algorithm 1:** Multiple shooting for OCP

---

**Result:** Non-linear equality and inequality constraints

initialize :  $\mathcal{M} = [m_1 \dots m_N] = \mathcal{M}_0$  ;

**repeat**

    Evaluate Tables;

$x_i^{N_{sub}} = RK4(x_i^1, u_i)$  ;

    Formulate dynamics constraints :  $m_{i+1} - x_i^{N_{sub}} = 0$ ;

    Formulate path constraints :  $g(m_i, u_i) \leq 0$ ;

**until** *Until*  $i = N - 1$ ;

Terminal dynamics constraints :  $m_N - x_{N-1}^{N_{sub}} = 0$ ;

Terminal path constraints :  $g(m_N, u_N) \leq 0$ ;

---

However, the time-step  $h$  of the RK4 integrator should stay quite small for a better accuracy, which is an issue considering that we need one variable for each state and for every time-interval. Assuming our problem can be fully simulated in 500 [s] and a time-step of 0.1 [s], a vector  $\mathcal{M} \in \mathbb{R}^{N_{\mathcal{M}}}$  with  $N_{\mathcal{M}} = 30000$  variables would be required (even 35000 with the discretized control input). Clearly, this can be considered as a large-scale optimisation problem, involving more advanced optimisation techniques such as the hierarchical problem decomposition of Chang et al. [33], consisting in dividing the large-scale problem into several short horizon problems, linked together with a parameter optimisation problem enforcing the continuity of the smaller OCPs.

Nevertheless, another technique consisting in performing sub-integration in each discrete time-interval shall be used in this work, enabling a very similar formulation as the one presented in Algorithm 1. As a matter of fact, nothing prevents one of creating a small simulation in each of these intervals in the aim of choosing a much bigger overall time-step, thus drastically reducing the effective number of variables. This was already explained in Equation 5.6. Considering  $N_{sub}$  sub-intervals,  $N_{sub}$  sub-integrations can be performed, originating from the artificial sub-interval initial condition  $m_i$ . The control input will be kept constant for each of the  $N_{sub}$  steps, while every other physical quantity will be updated accordingly for consistency, as illustrated in Figure 5.2. One drawback to this will be the control input accuracy, since it is subjected to a quite coarse discretisation. Please refer to Munzir et al. [34] for more detailed explanations.

### 5.2.1 Implementation of the NIST database

Up to now, it has been assumed that access to the system temperatures  $T_1$  and  $T_2$ , specific enthalpy  $h_1$  and pressures  $P_1$  and  $P_2$  was provided. However, according to the aforementioned explanations (see Section 4.2.2), no state equations were developed nor used throughout this work, the use of the NIST database being rather preferred. This raises substantial issues considering that several of those

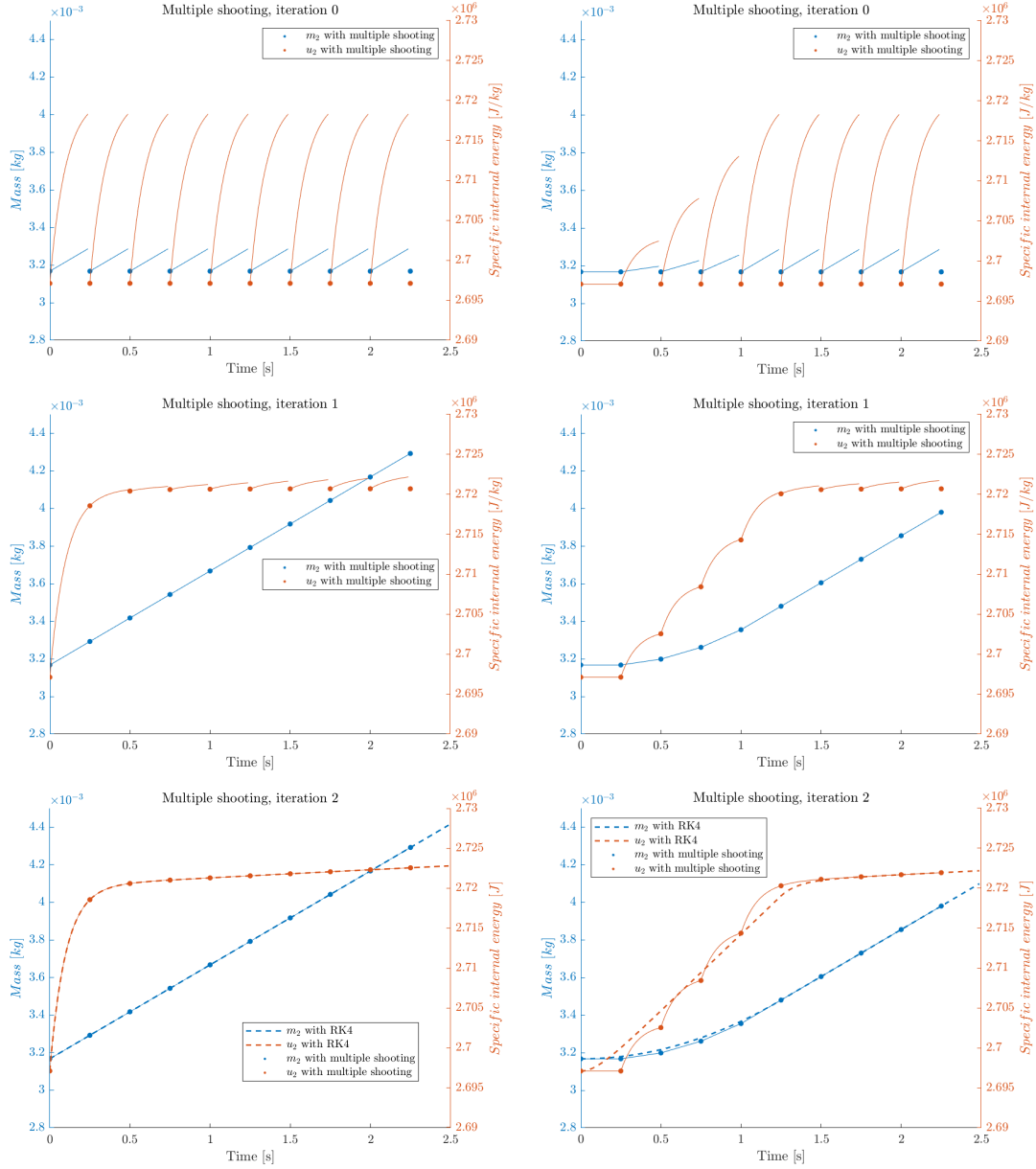


Figure 5.1: Multiple shooting demonstration for a constant mass-flow rate  $\dot{m} = 0.0005$  [kg/s] on the left and for a linearly varying mass-flow rate  $\dot{m} \in \{0.0005, 0.001\}$  [kg/s] on the right. For each column, 3 multiple shooting iterations are presented. On the last row, a simple Runge-Kutta 4 integration is superimposed in the aim of showing the influence of the multiple shooting time-step. As can be seen, even though the multiple shooting approximation shows its limits for the varying mass-flow rate case on the right, it eventually does not change the final mass or specific internal energy value.

quantities explicitly appear in the dynamics, formulated in Equation 4.18. The following paragraphs explain how this issue was mitigated.

In order for the NIST database to be usable within the optimisation routine, several sampling have been performed across the database, leading to discrete tables of properties. Two of our state variables being the tanks hydrogen mass  $m_1$  and  $m_2$  (which can easily be converted to a density according to the tanks volumes) and their respective specific internal energy  $u_1$  and  $u_2$ , two grid vectors  $D \in \mathbb{R}^{N_D}$  (for density) and  $U \in \mathbb{R}^{N_U}$  were defined and employed to evaluate the database. Therefore, three bidimensional tables could be defined and saved into .mat format as  $T(D, U)$ ,  $P(D, U)$  and  $H(D, U)$

for the temperature, the pressure and the specific enthalpy respectively as a function of the density to specific internal energy grid defined before.

### 5.3 First implementation using the Matlab<sup>®</sup> solver `fmincon`

#### 5.3.1 A first optimisation procedure

As a first iteration to solve the OCP problem at hand, the non-linear optimisation solver `fmincon` from the Matlab<sup>®</sup> optimisation toolbox was considered. The dynamics were handled with a MSA and a standard RK4 integrator was used throughout the sub-integration process. During this first step, a simple way to check the correctness of the formulation consists in optimising the system with a constant input  $u = \dot{m}_{const}$  and no objective,  $\mathcal{J} = 0$  in order to compare these results with a bare RK4 integration. Indeed, both should be equivalent up to some minor numerical errors since the MSA should be able to perfectly optimise the trajectory so that it is continuous in very few iterations, since dynamics continuity would be the only constraint here. This can be observed on Figure 5.2.

#### 5.3.2 Computation speed issue

Although this numerical formulation of the OCP at hand works very well, the computational time needed to perform any simulation with a reasonable number of variables stays very high. Indeed, `fmincon` uses finite differences in order to compute the gradient of the objective as well as the dynamics continuity constraints, as those are non-linear and also rely on tables. To overcome this issue, one can provide `fmincon` user-defined gradients of the problem objective and constraints, leading to the solver only performing efficient matrix multiplications and evaluations instead of lengthy, problem-dimension scaled finite difference operations. The usage of user-defined tables to replicate the NIST database (cf. Section 5.3) still requires to manually compute tables of gradients of the original tables. For example, a 50 [s] simulation performed with finite differences lasts about 329 [s] while it is roughly divided by two when using user-supplied gradients, amounting to about 166 [s]. Also, these computations have been performed when the model only had 4 states instead of 6, as well as no PID controllers, with all the constraints that it would involve.

Hence, tables containing the gradients of the three main tables were pre-computed using finite-differences with respect to the state variables  $m$  and  $u$ . This was performed using central finite differences for a better accuracy, as these yield a truncation error in the order of  $\mathcal{O}(\Delta x)^2$  :

$$\begin{aligned}\frac{\partial f}{\partial x} &= \frac{f(x+h, y) - f(x-h, y)}{2h} \\ \frac{\partial f}{\partial y} &= \frac{f(x, y+k) - f(x, y-k)}{2h}\end{aligned}\tag{5.10}$$

Next, the bilinear interpolation method displayed in Equation 5.11 was deployed to evaluate these tables, for a query point  $(x^*, y^*)$ .

$$\begin{aligned}f(x^*, y^*) &= \Delta f_x \frac{dx}{\Delta x} + \Delta f_y \frac{dy}{\Delta y} + \Delta f_{xy} \frac{dx}{\Delta x} \frac{dy}{\Delta y} + f(x, y) \\ \text{with} \\ \Delta f_x &= f(x_2, y_1) - f(x_1, y_1) \\ \Delta f_y &= f(x_1, y_2) - f(x_1, y_1) \\ \Delta f_{xy} &= f(x_1, y_1) + f(x_2, y_2) - f(x_2, y_1) - f(x_1, y_2) \\ \Delta x &= x_2 - x_1 \\ \Delta y &= y_2 - y_1\end{aligned}\tag{5.11}$$

This, however, raises another issue regarding the fact that the user must update the scripts computing the gradients each time the OCP formulation is modified. This leads to painfully long chain-rule based derivative calculations since the problem is recursively integrated at each time-step. Hence, after several weeks of persistent gradient computations, this option was discarded in favour of an optimal control oriented library, as will be explained in Section 5.4.

## 5.4 Second implementation with Casadi

In the second and main part of this work, the open-source *algorithmic* or *automatic* differentiation (AD) library Casadi from Andersson et al. [35] was used in order to drastically improve the computational speed for solving the OCP. Automatic differentiation consists in explicitly constructing expressions only with single atomic operations, building computational graphs enabling the calculation of derivatives by simple dot products. Hence, no gradient needs to be computed since AD automatically generates fast derivative computation with a cost proportional to simply evaluating the function to derive. Moreover, it is able to automatically detect sparsity patterns within the jacobian and hessian matrices in order to further save computation time.

Casadi also provides access to a non-linear programming (NLP) primal-dual interior-point solver, namely IPOPT, which will be used throughout this work.

### 5.4.1 RK4 sensitivity analysis

A critical parameter in any simulation requiring the integration of a system of first-order differential equations is the time-step  $h$ . The precision of the simulation strongly depends on its value, which means that it is very important to study the sensitivity of a given problem to variations in this parameter. Moreover, since the optimisation problem proposed in Section 6 will aim in minimising the time-step, bounds for its value will have to be provided, hence making it even more crucial to determine which of these are reasonable regarding the simulation accuracy.

In the aim of performing this analysis, the simulation setup proposed in Table 5.1 will be used. This means that this analysis will be performed for two time horizons, each of those being discretised with 8 different time-steps. The results obtained will be compared using one directly observable state, namely the mass in tank 2, and one thermodynamic state computed with the NIST database, namely the temperature inside tank 2. The results can be visualized on Figure 5.8.

Table 5.1: Time-step sensitivity simulation setup

Quantity	Value	Units
Total simulation time	$\{1; 100\}$	[s]
Time-step $h$	$\{10^{-3}; 10^{-2}; 1.3 \cdot 10^{-2}; 2 \cdot 10^{-2}; 4 \cdot 10^{-2}; 5 \cdot 10^{-2}; 6.3 \cdot 10^{-2}; 10^{-1}\}$	[s]
Linearly varying $u = \dot{m}$	$\{\dot{m} \in [5 \cdot 10^{-4}; 10^{-3}], \dot{m} \in [5 \cdot 10^{-4}; 10^{-2}]\}$	$\left[\frac{kg}{s}\right]$

On the left column of Figure 5.8, it can be observed that time-steps until  $h = 0.063$  [s] lead to acceptable simulation precision, unlike  $h = 0.1$  [s], for which the temperature trajectory becomes oscillatory. The critical time-step will hence be defined as  $h = 0.1$  [s], which means that the simulation time-step used throughout the optimisation procedure will be bounded by this value. The right column of Figure 5.8 shows the same simulation, but for 100 [s] and lets observe that the relative error decreases to very low levels. This is quite obvious since those are relative error computed with respect to the simulation performed with  $h = 10^{-3}$  [s], which is considered as the reference simulation from a precision perspective. Therefore, simulations with any of the time-steps presented in Table 5.1 can be considered as sufficiently accurate for the present system. However the main constraint remains the temperature, which appears to become unstable from  $h = 0.1$  [s] and higher, thus bounding the time-step at this values, as mentioned before.

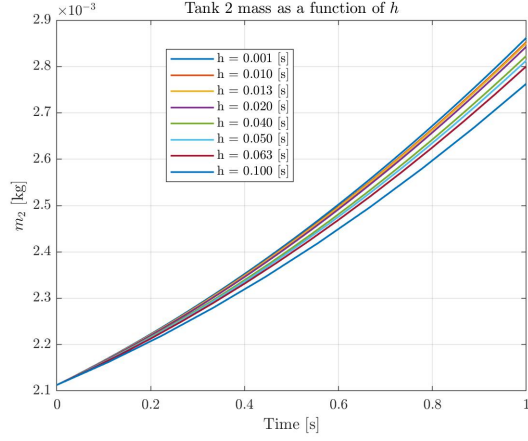
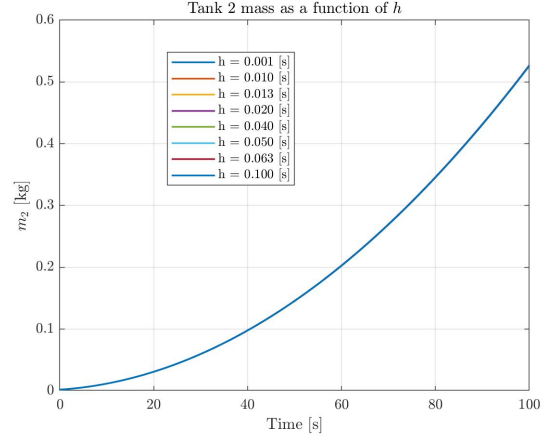
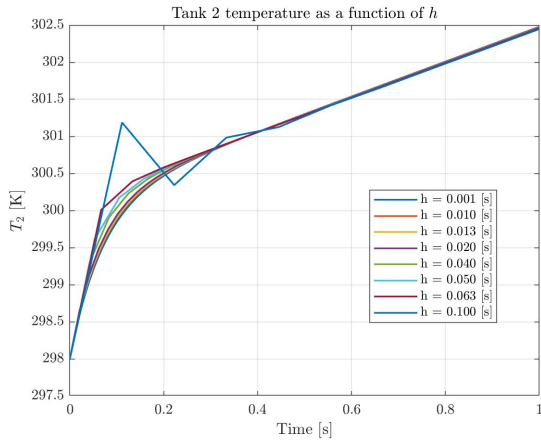
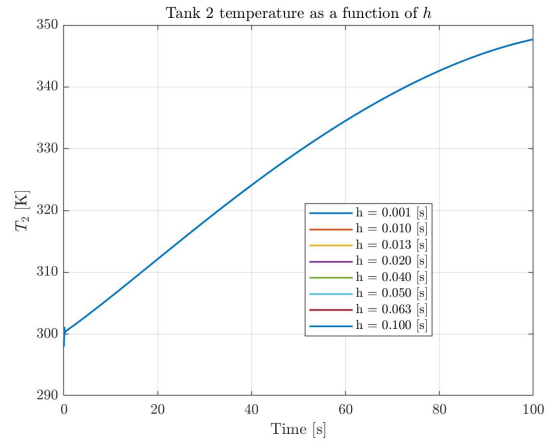
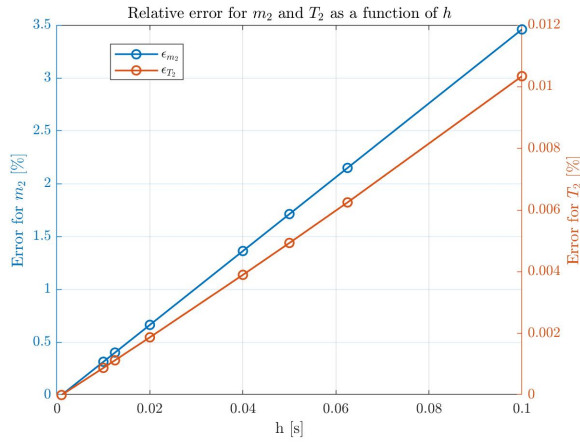
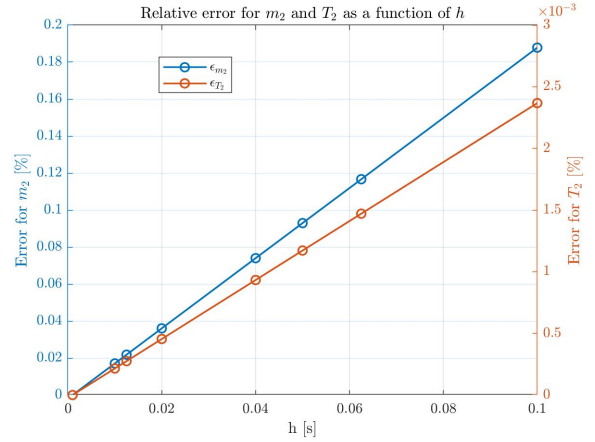
Figure 5.2: *Tank 2 mass*Figure 5.5: *Tank 2 mass*Figure 5.3: *Tank 2 temperature*Figure 5.6: *Tank 2 temperature*Figure 5.4: *Relative error*Figure 5.7: *Relative error*

Figure 5.8: Comparative analysis between different time-steps, time-horizons and mass-flow rates

## 6 Optimal control problem formulation and results

### 6.1 Free control input

In the aim of optimising the filling of our system, a free control law can be defined in order for the optimiser to be as effective as possible. It will allow the control input  $u(t)$  to have the shape which optimises at most the system at hand regarding its constraints. Therefore, the problem will first be stated and all its parameters will be justified. Next, two different simulation will be performed in order to observe the influence of the control horizon, the integration step size and the system temperature evolution. Different objectives will also be tested in combination with various constraints, allowing one to select an efficient OCP formulation. Next, a parametrised control law will be optimised under the form of a feedback law, allowing a straightforward implementation on a simple micro-controller, contrarily to the previous one.

#### 6.1.1 Free control input I

In general, free<sup>8</sup> control inputs are well-suited for online optimisation-based controllers, such as Model Predictive Control (MPC) controllers. Roughly speaking, these methods perform online optimisation and aim at designing an optimal control law satisfying several constraints along the system nominal trajectory, at each time step. Only the first control input is then retained and the optimisation starts again, with different sensor measurements. A similar approach arises in Optimal Control, with the difference that the control input will be optimised on the full refueling horizon. Being able to enforce hard physical constraints sounds appealing for controlling the system at hand, which will be performed in this section.

First of all, all physical constraints which should be satisfied by the system along the trajectory of its states must be formulated. These are presented in Tab. 6.1 and can be mathematically stated as simple lower and upper inequality, along with the global state constraints  $\mathcal{M} \in \mathcal{X}$ , where  $\mathcal{X}$  represents the allowable state domain. The OCP problem at hand is presented in Equation 6.2.

Table 6.1: Physical constraints on the system states

Quantity	Value	Units	Reason
Max. gas temperature	338	[K]	To avoid tank blow-up, the tank wall temperature should be limited to 85 [°C]. Hence, a limit of 65 [°C] will be used to take the system uncertainties into account.
Max. gas pressure	300	[bar]	Maximum sustainable tank pressure. However, pressures up to 330 [bar] will be allowed in order to reduce the filling time, although being minimized too.
Min. control input value	0	[kg/s]	The input cannot be negative since it is the inlet mass-flow rate and backflow is not allowed to occur with the use of anti-reverse valves.
Max. control input value	$\dot{m}_{max}(\rho_1, u_1)$	[kg/s]	The input cannot be outreach the maximum mass-flow rate allowed by the system physics, which is stated in Equation 4.19.
Final tank 2 mass	0.267	[kg]	Final mass needed in order to have a final state of ( $P_2 = 300$ [bar]; $T_2 = 298$ [K]).

Then, the objective can be stated according to the performance desired. In the present case, a simple time-step (and thus total time) minimisation will be performed, allowing filling time optimisation. The optimiser vector and the full optimisation problem are respectively formulated in Eqs. 6.1 and 6.2. The following terminology will be used :

- Index  $i$  represents the  $N$  time-intervals of the problem

<sup>8</sup>The denomination "free" denotes the freedom which the optimisation solver has for optimising the control input  $u$ . In this case, the number of degrees of freedom is equal to the number of time-intervals  $N$ , and is thus maximal, supporting the "free" designation.

- Index  $k$  represents the  $N_{sub}$  time-intervals sub-integration
- The same states denomination as was introduced in Equation 4.18 will be used

$$\mathcal{M} = \{x_{1,1} \dots x_{1,N} \ x_{2,1} \dots x_{6,N} \ u_1 \dots u_N \ h\}^T \quad (6.1)$$

$$\begin{aligned}
& \underset{h}{\text{minimize}} && \mathcal{J}(\mathcal{M}) = h \\
& \text{subject to :} && m_{i+1} = f(x_{N_k}(m_i, u_i), p_k), \quad i = 1, \dots, N-1 && (\text{Dyn. continuity}) \\
& && T_{2,i} \leq 338, \quad i = 1, \dots, N && (\text{Tank 2 temp.}) \\
& && m_{2,N} \geq r, && (\text{Mass reference}) \\
& && u_i \geq 0, \quad i = 1, \dots, N-1 && (\text{Input positiveness}) \\
& && u_i \leq \dot{m}_{\max}(\rho_{1,i}, u_{1,i}), \quad i = 1, \dots, N-1 && (\text{Input saturation}) \\
& && \mathcal{M} \in \mathcal{X} && (\text{Variables bounds})
\end{aligned} \quad (6.2)$$

---

**Algorithm 2:** Free control input optimisation algorithm

---

**Result:** Optimal input  $u^*(t)$ , optimal trajectory  $X^*(t)$  and optimal time step  $h^*$

Initialize :  $M = [m_1 \dots m_N]$  ;

State bounds :  $x \in \mathcal{X}$  ;

**repeat**

$x_{sub} = X_i$ ;

$u_{sub} = u_i$ ;

**repeat**

$x_k = x_{sub}$ ;

        Evaluate Tables;

        Evaluate system parameters  $p_k$ ;

$x_{sub} = RK4(x_k, u_{sub}, p_k)$  ;

**until** *Until*  $k = N_{sub}$ ;

    Dynamics constraints :  $X_{i+1} = x_{sub}$ ;

    Temp. constraints :  $T_2(\frac{x_{sub}^{(1)}}{V_2}, x_{sub}^{(2)}) \leq T_{\max}$ ;

    Pressure constraints :  $P_2(\frac{x_{sub}^{(1)}}{V_2}, x_{sub}^{(2)}) \leq T_{\max}$ ;

    Input positiveness :  $u_{sub} \geq 0$  Input saturation :  $u_{sub} \leq \dot{m}_{\max}$

**until** *Until*  $i = N - 1$ ;

Terminal path constraints :  $T_2(\frac{x_N^{(1)}}{V_2}, x_N^{(2)}) \leq T_{\max}$ ;

Reference constraint :  $x_N^{(1)} \geq r$ ;

Optimize :  $\underset{h}{\text{minimize}} \mathcal{J} = h$

---

### 6.1.2 Result : an optimal discrete-time control law

In this section, the results for all possible system states will be presented for completeness, even though those for tank 1 states will not be provided in the incoming sections, simply because it does not contain much added value for the results analysis.

First, it can be observed on Figure 6.1 (d) that the temperature evolution exactly follows the imposed temperature limit, i.e. 338 [K]. This is due to the single-objective optimisation, which only takes filling time into account. This yields an optimal filling time of 39 [s] for a reference mass of 267 [g], as desired. However, it can be noticed that both the control input and the temperature in Figure 6.1 (c) and



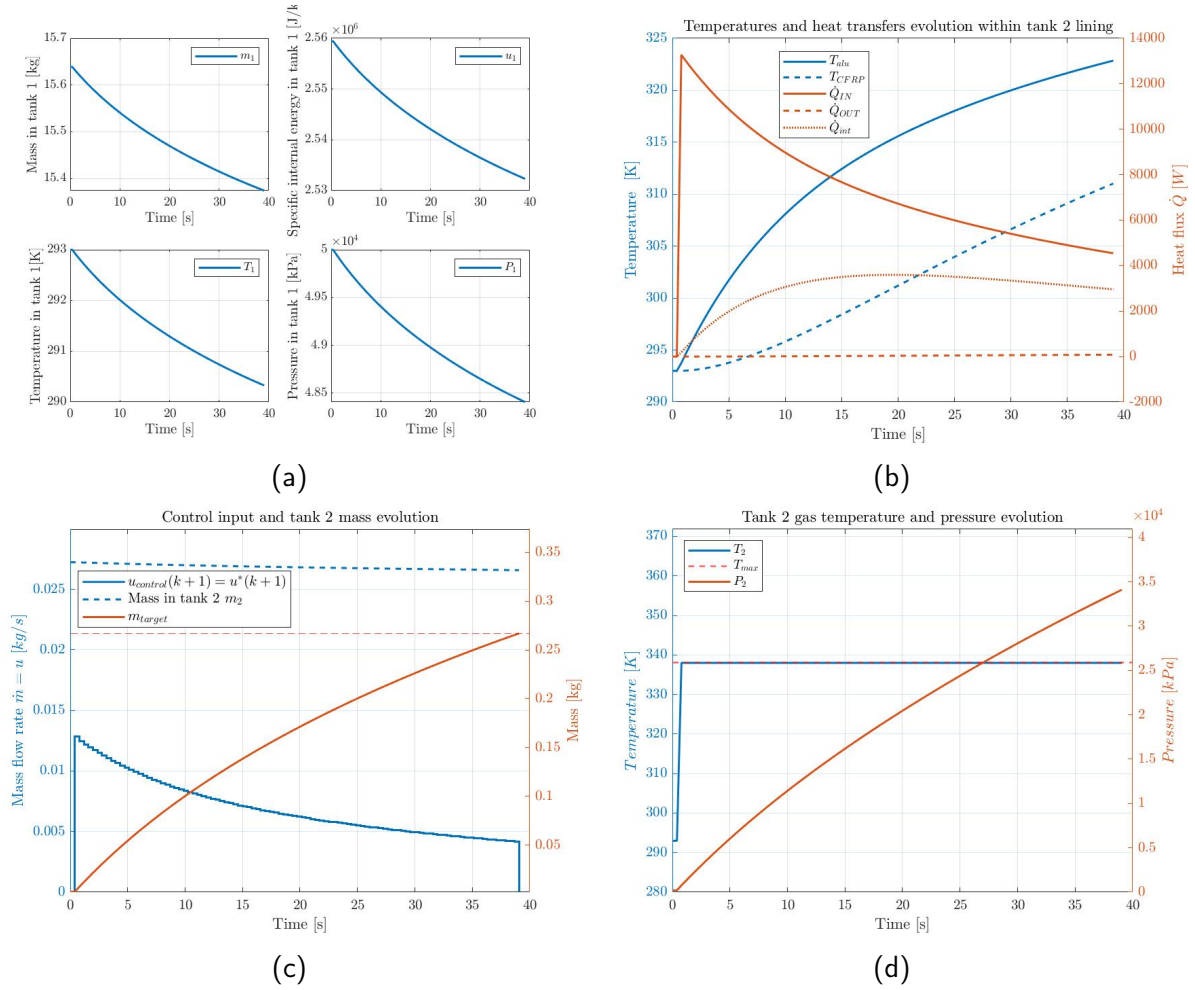


Figure 6.1: Results for the time-optimised free control input simulation

(d) follow an unrealistic, quasi-instantaneous rise at the second time-step. From the solver point of view, this basically means that the best possible trajectory for fast-filling consists in increasing the gas temperature as high as possible in the aim of benefiting as much as possible from the heat transfers which occur through the tank lining. And indeed, those happen to raise with the temperature, as can be seen on Figure 6.1 (b), on the continuous red curve. They then diminish since the wall temperature starts increasing, with a little delay, as the blue dotted line shows. Figure 6.1 (a) lets observe the inverse behaviour happening in tank 1, due to its emptying process. This yields a decrease in temperature in the supply tank, which in turn diminishes the total enthalpy provided to tank 2, as discussed in Woodfield et al. [12]. Additionally, the dotted blue curve on Figure 6.1 (c) represents the choking condition, restricting the mass flow rate.

By looking more closely at the control input curve on Figure 6.1 (c), an explicit similarity with a simple proportional feedback law becomes obvious. This makes sense that some sort of proportional controller could also be an option, as will be demonstrated in Section 6.2. Although the optimised control law found here looks both appealing and promising, it may be of some interest to investigate whether a less abrupt controller shape could be identified by transforming our single-objective OCP into a multi-objective OCP. Also, the mass trajectory on Figure 6.1 (c) suggests that a mass overshoot could very well occur if the valve is not closed perfectly on time. The issue with that is precisely that some delay caused by all the gas contained within the pipe will actually make the mass overshoot, which is not desired since backflow cannot occur in our system. Lastly, the pressure outreaches both the lower and upper pressure limits, respectively set at 300 [bar] and 330 [bar].

### 6.1.3 Free control input II

For all of the aforementioned reasons, the optimised control input found in Section 6.1.2 cannot be used as it is. Therefore, several changes can be brought to both the objective and the constraints set in order to enforce additional conditions allowing a well-posed control law.

First, an anti-overshoot measure is added by specifying that the mass trajectory should be contained within a shallow tube of size  $\epsilon$  at the end of its path. This can be mathematically formulated by stating that the mass at time-step  $N - n_\epsilon$  must be less than our target as well as more than a lower value for our tube. The  $\epsilon$  value will obviously be minimised in order for the mass to be as close to the reference as possible. Next, a simple manner of dealing with the pressure constraints consists in bounding the pressure with a soft constraint, formulated as a hard constraints complemented by a slack variable, which will also be minimised. Now, the hard constraint can be set to 300 [bar] and the slack bounded in the domain 0, 30 [bar] in order to fulfill the {330 [bar]} hard constraint. The state and input vector and the OCP are formulated in Equations 6.4 and 6.3.

$$\mathcal{M} = \{x_{1,1} \dots x_{1,N} \ x_{2,1} \dots x_{6,N} \ u_1 \dots u_N \ h\}^T \quad (6.3)$$

$$\underset{h, \epsilon, \eta, m_2}{\text{minimize}} \quad \mathcal{J}(\mathcal{M}) = Q_h h^2 + Q_\epsilon \epsilon^2 + Q_\eta \eta^2 + (m_2 - r)^T Q_m (m_2 - r)$$

$$\begin{aligned} \text{subject to : } & m_{i+1} = f(x_{N_k}(m_i, u_i), p_{N_k}), \quad i = 1, \dots, N-1 && (\text{Dyn. continuity}) \\ & T_{2,i} \leq 338, \quad i = 1, \dots, N && (\text{Tank 2 temp.}) \\ & P_{2,i} \leq 300 + \eta, \quad i = 1, \dots, N && (\text{Tank 2 pressure}) \\ & 0 \leq \eta \leq 30, && (\text{Pressure slack}) \\ & r - \epsilon \leq m_{2,N-n_\epsilon:N} \leq r, && (\text{Mass reference}) \\ & 0 \leq \epsilon \leq 0.01r, && (\text{Mass slack}) \\ & u_i \geq 0, \quad i = 1, \dots, N-1 && (\text{Input positiveness}) \\ & u_i \leq \dot{m}_{\max}(\rho_{1,i}, u_{1,i}), \quad i = 1, \dots, N-1 && (\text{Input saturation}) \\ & \mathcal{M} \in \mathcal{X} && (\text{Optimisers bounds}) \end{aligned} \quad (6.4)$$

The cost coefficients (or matrices)  $Q_h$ ,  $Q_\epsilon$ ,  $Q_\eta$  and  $Q_m$  correspond to weights for the different terms in the objective term. This is important because although the objective in Equation 6.2 only contained one term, in the present case multiple term are interacting, thus creating a multi-objective optimisation problem. Such problems can yield dominated solutions since one term in the objective might have much more influence on the solution than other terms. In the aim of finding a non-dominated solution for the problem at hand, several techniques exist, the most basic of which consisting in summing weighted objectives with user-supplied weights for each single objective. This allows one to find a solution for the optimiser that belongs to what is called the *Pareto-optimal set*, yielding an objective itself belonging to a *Pareto-optimal front*. These are respectively a non-dominated set of the feasible optimiser space and its mapping onto the objective space.

Although being quite simple to implement in practice, this method involves a manual tuning of the cost coefficients, depending on the user purpose. Nevertheless this approach has been used here, partly because a prior knowledge of the respective importance of the different objective terms was quite simple to obtain. Indeed, it mainly depends on the respective physical units these terms have. Since a lot of tuning was necessary to find an appropriate solution, only the basic methodology will be exposed here.

For example, the time cost coefficient can be set by using the inverse squared of the lower bound defined for the time-interval. In this case, this lets the time-step normalized to 1 if it is close the its lower bound, and about two order of magnitude greater if it sits in the other end of the spectrum, i.e. close to the upper bound. For the reference slack cost coefficient,  $Q_\epsilon$ , the inverse squared of the lower bound has also been considered, simply because it normalizes it to 1 and elevates it to about two orders of magnitude greater in case it gets closer to its upper bound. The pressure slack cost coefficient was a little bit trickier to tune, and can be set to its upper bound (or a little bit less, depending on the need) because it varies in the range  $\{0, 3000\}$  [kPa], and limiting too much the final pressure value would make the problem either infeasible or very sub-optimal. Indeed, the total filling time would simply be increased. On the other hand, the mass cost coefficient was tuned quite straightforwardly at  $1/r^2$  since the maximum and final value of  $m_2$  precisely is the mass reference  $r$ .

Then, the results are presented on Figure 6.2.

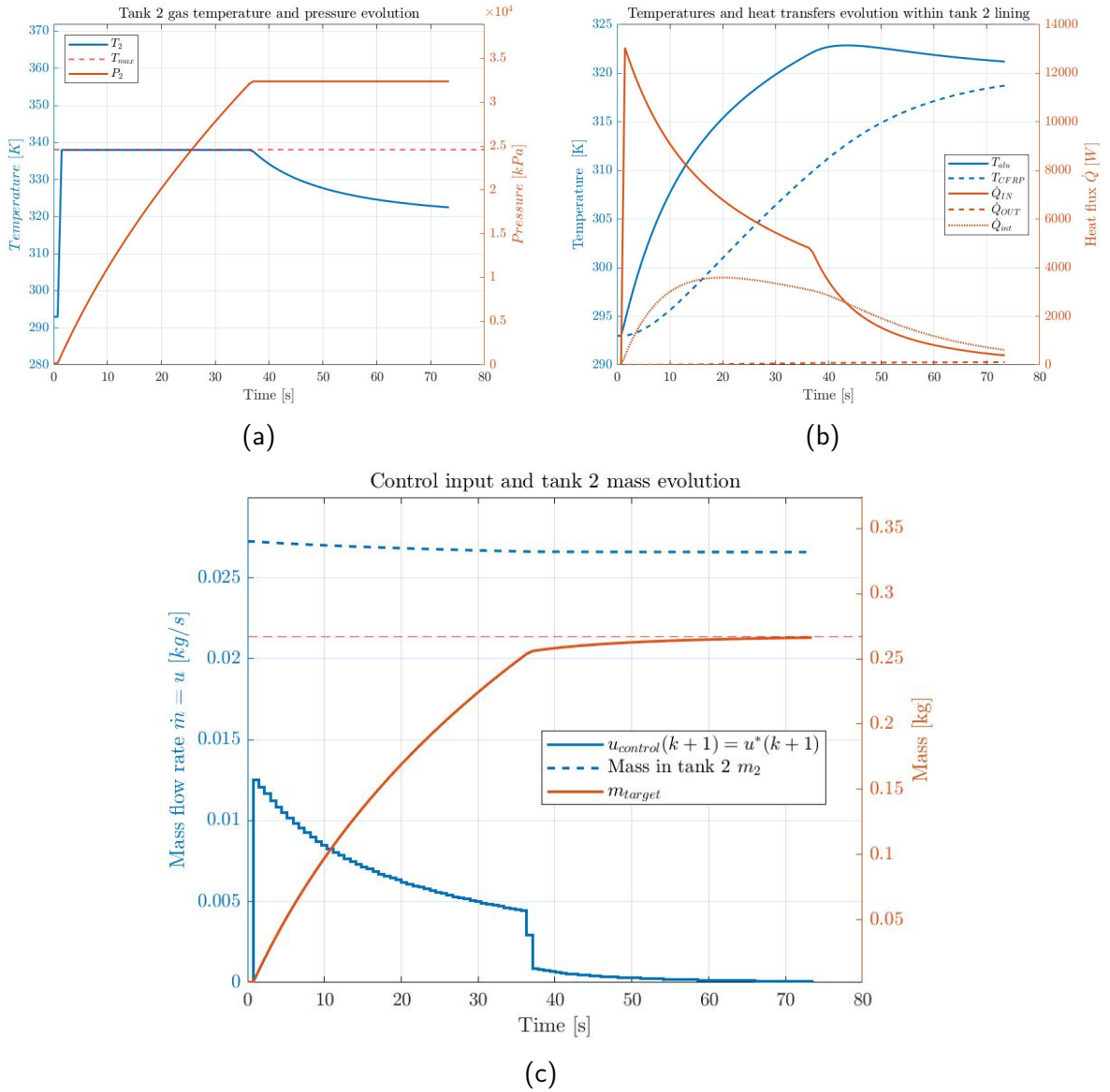


Figure 6.2: Results for the multi-objective free control input simulation

The first difference is the temperature decrease observed on Figure 6.2 (a), allowing a pressure stabilization at 324 [bar], although no drop since the mass flow rate is still non-negative until the end of the simulation. By looking at Figure 6.2 (c), it can also be seen that this induces a smoothing

effect on the mass flow rate, avoiding any potential overshoot. However, the evident tradeoff is the filling time, which increases from 40 [s] to 73 [s]. More importantly, the CFRP temperature seen on Figure 6.2 (b), which is eventually the critical value to control, has been kept under 50 [°C], letting a quite large safety margin for other model uncertainties.

Although all of this looks very promising from a performance point of view, the only way this could be implemented in a real system would consist in establishing a MPC controller. As mentioned before, this type of online optimisation routine can manage to find an optimal control input for each time step, also satisfying all the system constraints. Installing MPC would involve some sort of more powerful micro-controller in order for it to run properly, but this is not an issue. The primary problem with that simply is that online controllers could very well fail, for any sort of reason, and one cannot afford that regarding the user safety. Indeed, letting the system fail would perhaps allow the gas temperature to reach critical temperatures, weakening the epoxy matrix of the CFRP lining, and even leading the whole station explosion<sup>9</sup>.

For this reason, MPC cannot be considered as a reliable option unless there is a consistent backup feedback law. It is thus of some interest to use the methodology developed before in the aim of designing such a law. Hence, the next section will explain how this can be performed and results will be provided. Also, two further iterations will seek at showing how one can improve this law.

## 6.2 Optimal PID controller

Rather than computing an optimal control input, a PID law can be directly imposed within the optimisation problem in order for the latter to directly optimise its coefficients. The advantageous feature here is that a direct implementation of the controller of choice can be performed. Indeed, even several controllers can be implemented in one optimisation problem, as will be seen in Section 6.2.3. Hence, it will first be shown how a simple PID controller can be identified using a reworked version of the OCP presented in Section 6.1.1. Then, two controller will be stacked together in the aim of smoothing the resulting control law. Finally, extensive simulation will be performed to ensure the stability and robustness of the established control law.

### 6.2.1 OCP statement

**PID formulation** The general form for a simple parallel (or ideal) PID controller is given in Equation 6.5, which can then be discretized to obtain Equation 6.6,  $r$  being the desired setpoint and  $x$  the input of our controller. In the present case, the objective consists in filling the tank up to a certain target, which will actually be the setpoint for the controller design, while  $x$  will be a measurement coming through the feedback loop of the system. For simplicity, it will be first assumed that we continuously have access to the system mass, thus yielding  $x = x_1$  when referring to the system of Equation 4.18.

$$u(t) = K_P(r - x(t)) + K_I \int_{t=0}^T (r - x(t))dt + K_D \frac{\partial x(t)}{\partial t} \quad (6.5)$$

$$u_{PID}(k+1) = K_P(r - x(k)) + K_I \sum_{i=1}^k (r - x(i))h + K_D \frac{x(k) - x(k-1)}{h} \quad (6.6)$$

In the aim of writing an optimisation problem for PID controller design, the one of Equation 6.4 can be modified and the discretised control input can be removed from its state and input vector. Instead, the PID control law shall be enforced. Hence, three variables representing the three controller

<sup>9</sup>A car hydrogen fast-filling station exploded in 2019 in Norway, probably to a hydrogen leak : <https://www.greentechmedia.com/articles/read/automakers-hold-steady-on-fuel-cell-ev-sales-despite-norway-blast>

design parameters  $K_P$ ,  $K_I$  and  $K_D$  are introduced and used to recursively compute the next controller output or system input in each time-interval. Hence, the dynamics input is recursively replaced by Equation 6.6, with  $r$  be the final desired mass  $m_{2,N} = 267$  [g],  $x$  being the output of a sensor providing tank 2 mass,  $X_1 = m_2$ , as is shown in Figure 6.3. Also, the integrator integrating period will be chosen as a finite number  $t_I \in \{0, N\}$  since integrating the error from the start of the filling process would ultimately lead to a huge accumulated error and troubles in the optimisation problem. Indeed, the primary concern of the integrator consists in accelerating the raise to reach the reference, but only up to a certain point since it generally yields a huge overshoot. In theory, this effect can be counterbalanced by the derivative term. However, absolutely no overshoot is allowable in the present case, hence the choice of a finite, quite small integrating time has been considered.

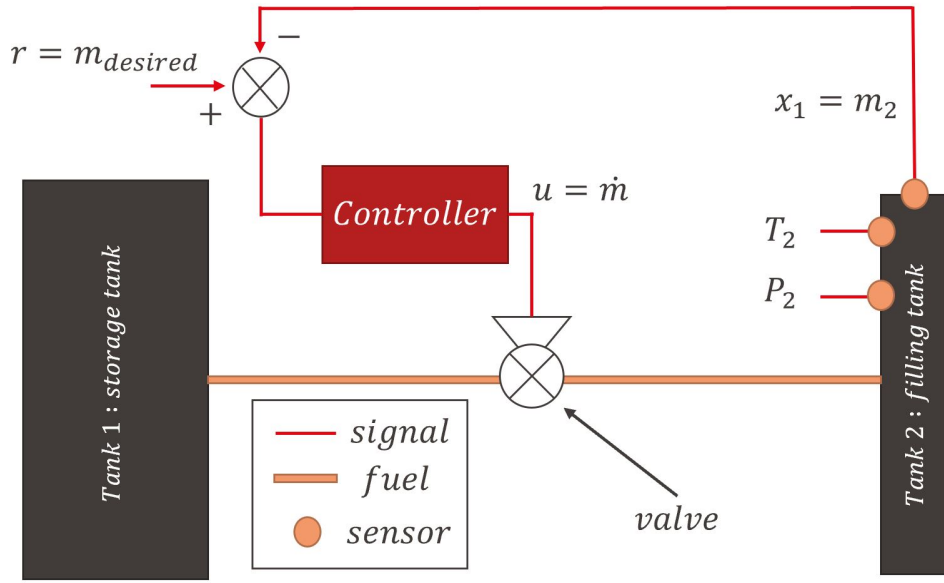


Figure 6.3: Block-diagram representation of the physical system with its PID controller

Also, since each time-interval is then integrated into  $N_{sub}$  sub-intervals, the input will be kept constant throughout each of these. The objective function can be designed by minimizing the total filling time as well as both the mass and the pressure slacks,  $\epsilon$  and  $\eta$ . All the problem equality and inequality constraints can be found in Equation 6.8, as well as the state vector in Equation 6.7.

$$\mathcal{M} = \{x_{1,1} \dots x_{1,N} \ x_{2,1} \ \dots \ x_{6,N} \ h \ K_P \ K_I \ K_D\}^T \quad (6.7)$$

$$\underset{h, \epsilon, \eta}{\text{minimize}} \quad \mathcal{J}(\mathcal{M})_{PID} = Q_h h^2 + Q_\epsilon \epsilon^2 + Q_\eta \eta^2$$

$$\begin{aligned}
\text{subject to : } m_{i+1} &= f(x_{N_k}(m_i, u_i), p_{N_k}), \quad i = 1, \dots, N-1 && (\text{Dyn. continuity}) \\
T_{2,i} &\leq 338, \quad i = 1, \dots, N && (\text{Tank 2 temp.}) \\
P_{2,i} &\leq 300 + \eta, \quad i = 1, \dots, N && (\text{Tank 2 pressure}) \\
0 &\leq \eta \leq 30, && (\text{Pressure slack}) \\
r - \epsilon &\leq m_{2,N-n_\epsilon:N} \leq r, && (\text{Mass reference}) \quad (6.8) \\
0 &\leq \epsilon \leq 0.01r, && (\text{Mass slack}) \\
u_i &\geq 0, \quad i = 1, \dots, N-1 && (\text{Input positiveness}) \\
u_i &\leq \dot{m}_{\max}(\rho_{1,i}, u_{1,i}), \quad i = 1, \dots, N-1 && (\text{Input saturation}) \\
\mathcal{M} &\in \mathcal{X} && (\text{Variables bounds})
\end{aligned}$$

---

**Algorithm 3:** OCP formulation for optimal PID controller

---

**Result:** Optimal input  $u^*(t)$ , optimal trajectory  $X^*(t)$  and optimal time step  $h^*$

Initialize :  $M = [m_1 \dots m_N]$  ;

State bounds :  $x \in \mathcal{X}$  ;

**repeat**

$x_{sub} = X_i$ ;

$u_i = K_P(r - x_1(k)) + K_I \sum_{i=1}^k (r - x_1(i))h + K_D \frac{x_1(k) - x_1(k-1)}{h}$ ;

**repeat**

$x_k = x_{sub}$ ;

        Evaluate Tables;

        Evaluate system parameters  $p_k$ ;

$x_{sub} = RK4(x_k, u_i, p_k)$  ;

**until** *Until*  $k = N_{sub}$ ;

    Dynamics constraints :  $X_{i+1} = x_{sub}$ ;

    Temp. constraints :  $T_2(\frac{x_{sub}^{(1)}}{V_2}, x_{sub}^{(2)}) \leq T_{\max}$ ;

    Pressure constraints :  $P_2(\frac{x_{sub}^{(1)}}{V_2}, x_{sub}^{(2)}) \leq T_{\max}$ ;

    Input positiveness :  $u_{sub} \geq 0$  Input saturation :  $u_{sub} \leq \dot{m}_{\max}$

**until** *Until*  $i = N - 1$ ;

Terminal path constraints :  $T_2(\frac{x_N^{(1)}}{V_2}, x_N^{(2)}) \leq T_{\max}$ ;

Reference constraint :  $r - \epsilon \leq x_N^{(1)} \leq r$ ;

Slack constraint :  $0 \leq \epsilon \leq 0.01r$ ;

Optimize :  $\underset{h}{\text{minimize}} \mathcal{J} = Q_h h^2 + Q_\epsilon \epsilon^2 + Q_\eta \eta^2$

---

### 6.2.2 Results for a single optimal PID controller

The resulting trajectories for the optimal PID controller are presented in Figure 6.4.

If these results are compared with those found with either the first or the second optimal control input found before, the first thing that can be noticed is the substantially smoother temperature evolution of Figure 6.4 (a). Also, the overall shape of the control input on Figure 6.4 (c) resembles very much the one derived with the free control input. However it looks much more realistic since it gently decreases until the reference is attained. Even though implementing such a control law seems perfectly feasible, there are still sharp peaks appearing on the beginning of both the temperature and internal heat transfer curves, seen on Figure 6.4 (a) and (b). These are caused by the PID abruptly starting to operate, and this must be mitigated. It can also be seen that the performance has slightly dropped compared to the control law developed in Section 6.2, with a 82 [s] filing time to compare

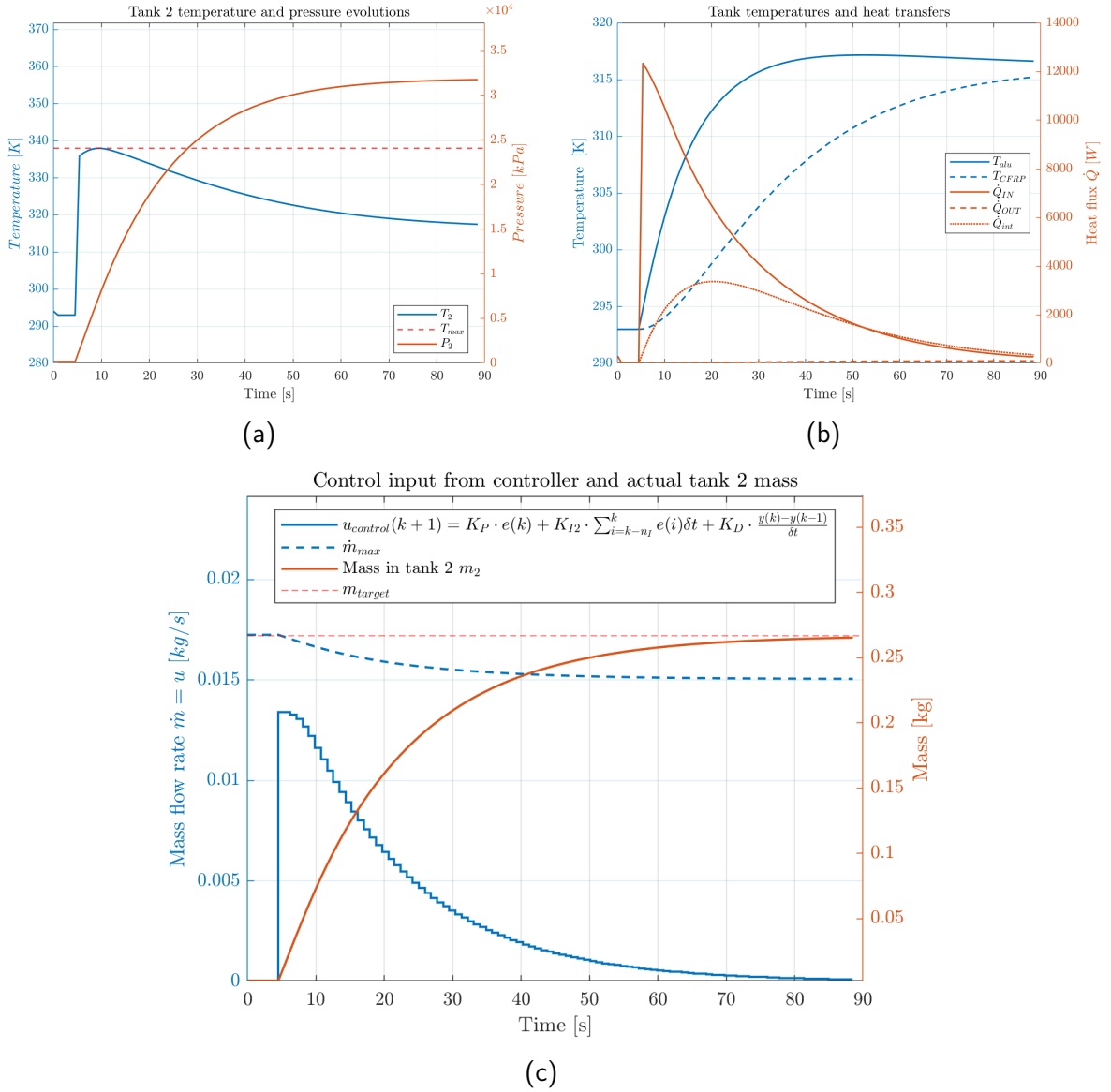


Figure 6.4: Results for the time-optimised optimal PID controller :  $K_p = 1 \cdot 10^{-2}$ ,  $K_I = 4.9 \cdot 10^{-6}$  and  $K_D = 0.77$ , while the integrator integrating period is  $t_I = 4$  [s]

to the 73 [s] obtained before.

### 6.2.3 Adding a second controller

A relatively trivial way of removing these peaks consists in adding a different controller at the beginning of the filling process. Here, a simple integrator has been added in the aim of initiating a very appreciated error reduction for a duration  $t_1$ , while gently increasing the mass-flow rate. Then, the full-scale PID is brought into action. In order to implement that, an additional constraint has to be set regarding the input continuity at time  $t_1$ , formulated as  $u_{t_1} = u_{t_1+1}$ . It can be noticed that an additional variable  $K_{I1}$  must also be added to the optimiser vector of Equation 6.7 and the original PID integrator coefficient  $K_I$  can be renamed  $K_{I2}$ . The resulting trajectories can be observed in Figure 6.5.

Now, the resulting control input shown in Figure 6.5 (c) appears to be much smoother than with the single PID controller. This solves the peaks problem because the single integrator present at the



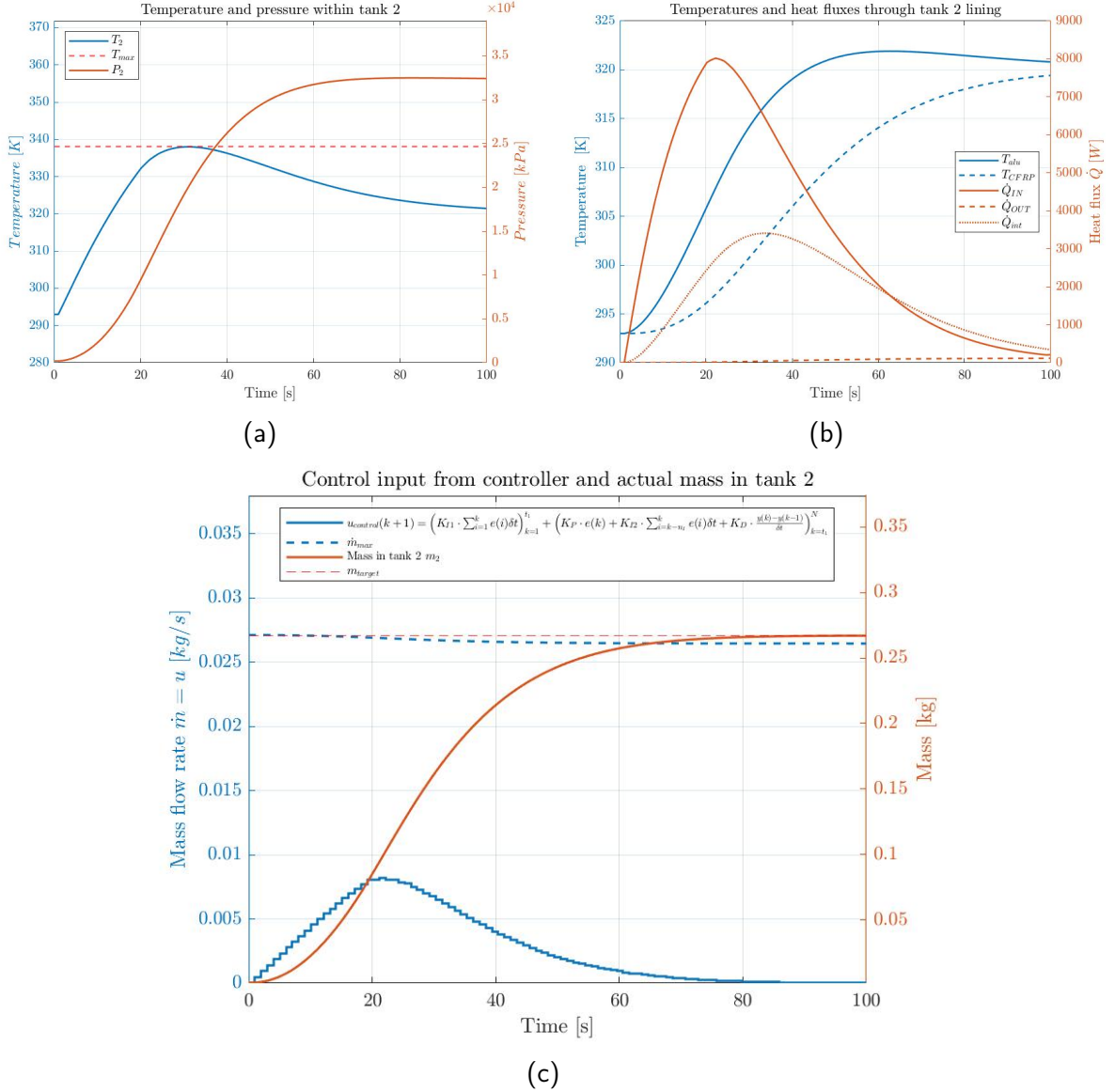


Figure 6.5: Results for the time-optimised optimal I-PID controller :  $K_{I1} = 1.76 \cdot 10^{-3}$ ,  $K_p = 1.39 \cdot 10^{-2}$ ,  $K_{I2} = 2.1 \cdot 10^{-6}$  and  $K_D = 0.7$ . The switching time and the second integrator integrating period are respectively  $t_1 = 20$  [s] and  $t_{I2} = 10$  [s].

beginning provides a smooth control input and temperature raise. Indeed, the first integrator  $K_{I1}$  acts from time 0 until  $t_1 = 20$  [s], which corresponds to the temperature peak of Figure 6.5 (a). This was decided after having performed some tuning, as well as the integrator period, set to 10 [s] since its action starts with 20 [s] of delay due to the first integrator.

## 6.2.4 Simulations

## 6.3 Adaptive optimal PID controller (APID)

At this point, an appropriate, well-working and optimised controller has been defined. However, there is a substantial issue, but it does not concern the controller itself, but rather the model it has been optimised onto. As a matter of fact, several parameters are uncertain, the most likely being the heat transfer coefficients : realistically,  $UA_{IN}$ ,  $UA_{OUT}$  as well as  $UA_{int}$  do vary with both the mass-flow rate and the initial ambient temperature (or more generally the filling process working conditions). Since



these parameters may vary a lot, it would not be that interesting to focus on the enhancement of the system modeling, but the controller shall rather be adapted to these variations. Indeed, a simple PID controller is not robust to parametric uncertainties, by construction. For sure, the reference will be attained anyways, but the primary goal of optimising such a controller was to avoid exceeding several hard and soft physical constraints, such as the temperature and the pressure, while minimising the total filling time. This means that a PID optimised using  $UA_{IN} = 300 [W/m^2K]$  will overshoot the temperature limit if the real-life internal heat transfer coefficient decreases to  $UA_{IN} = 250 [W/m^2K]$ , since it would allow less heat to leave the tank. This is also caused by the fact that the PID controller at hand does not know anything about the temperature since it only operates with mass measurement as a controller input signal. And although the temperature is controllable through the mass-flow rate, the parametric uncertainties that sits between the control input and the temperature prevent any direct or precise control of this quantity. It can be noticed that MPC would provide an excellent solution to this, even though this will not be done for the reasons invoked in Section 6.1.3.

Thereby, the parametric uncertainties caused by the potential variations of the heat transfer coefficients will be studied by considering variations of the internal heat transfer coefficient  $UA_{IN}$  only. Extension to the other heat transfer coefficients as well as other potential parametric variations ensue quite straightforwardly. Then, a solution counteracting this issue will be proposed, and then several parametric variations scenarios will be studied.

### 6.3.1 Parametric uncertainties

Throughout this section, the internal heat transfer coefficient will be varied within the interval  $UA_{IN} \in \{200, 400\} [W/m^2K]$  in the aim of observing the effect of this variation on the temperature overshoot. In order to perform that, the optimal I-PID controller presented on Figure 6.5 will be used, knowing that it was optimised using  $UA_{IN} = 300 [W/m^2K]$ . The results are presented on Figure 6.6 and correspond to 60 simulation that have been performed with the same controller and by continuously varying  $UA_{IN}$  in a linear manner.

Figure 6.6 (a) clearly shows how low is the robustness of the optimal I-PID to parametric uncertainties. The gas temperature overshoots its 65 [K] limit by rising up to 73 [K], which is undoubtedly unacceptable, while the aluminium and CFRP temperatures only vary by 2 or 3 degrees. This is due to the huge changes in the heat transfer rate, peaking at 8808 [W] instead of the nominal 8272 [W]. Also, Figure 6.6 (c) obviously indicates that neither the control input nor the tank 2 final mass suffer from these variations since they only depend on mass measurements.

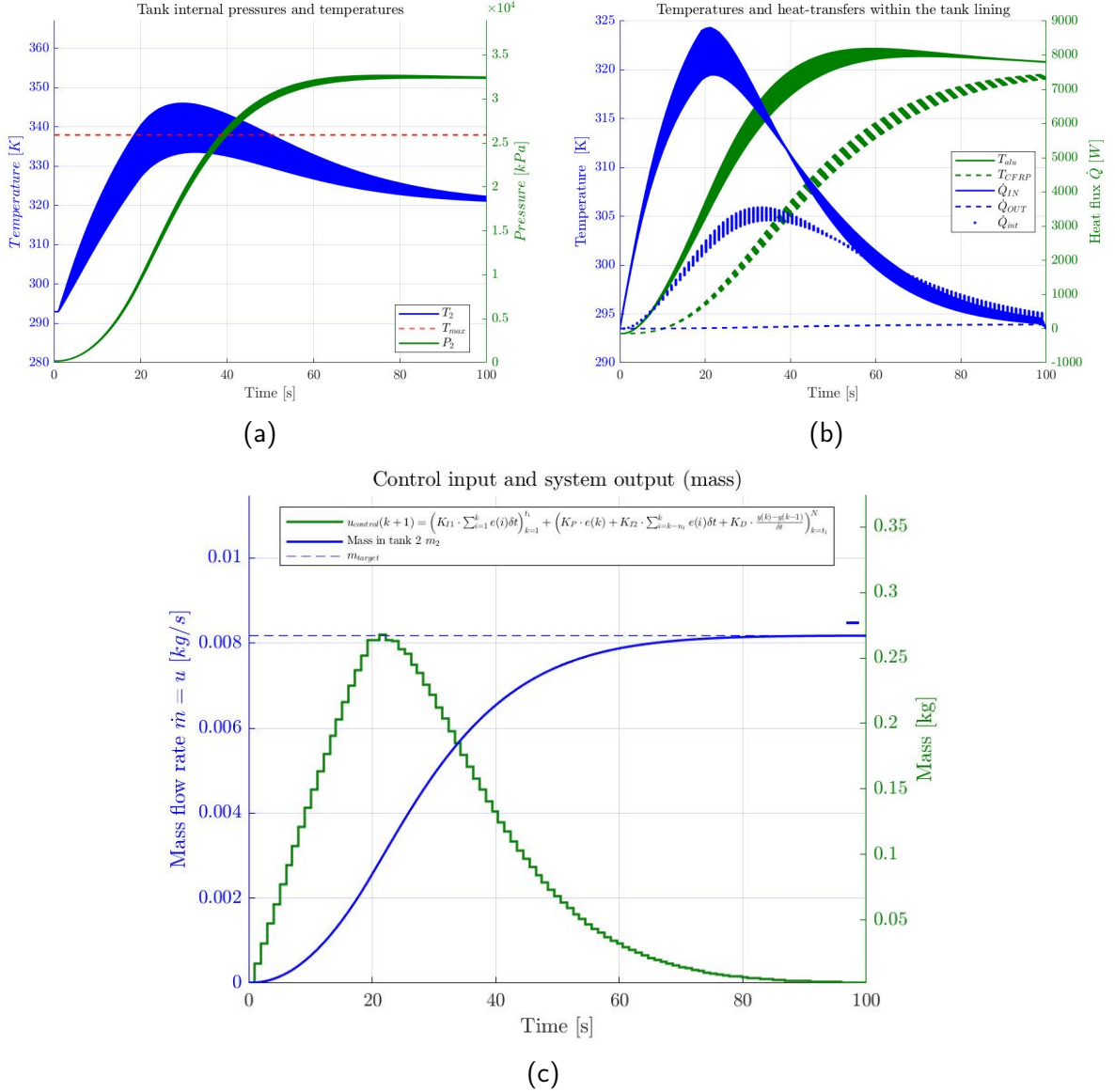


Figure 6.6: 60 simulations performed by using a nominal I-PID controller with  $K_{I1} = 1.76 \cdot 10^{-3}$ ,  $K_p = 1.39 \cdot 10^{-2}$ ,  $K_{I2} = 2.1 \cdot 10^{-6}$  and  $K_D = 0.7$ , while  $UA_{IN} \in \{200, 400\} [W/m^2K]$ .

### 6.3.2 Optimal adaptive PID controller

In the aim of avoiding any problems regarding the CFRP lining temperature, temperature overshoots caused by parametric uncertainties must be mitigated. A first solution for that could consist in performing online I-PID optimisation depending on the temperature evolution. However, even though controller parameters updates would be achieved every  $n$  time-steps, this would lead to the same pitfalls than MPC. Hence, instead of updating an online optimised controller, several controllers can be optimised offline and then switched according to pre-defined conditions. This is actually called an Adaptive controller, or an Adaptive PID controller in the present case (APID). One can refer to Karimi [36] for examples of adaptive controllers.

Since the feedback control law which was designed does not take into account other signals than the mass in tank 2, another method has to be established in the aim of taking the temperature limit into consideration. In practice, a temperature sensor could be set into tank 2 in order to have online measurements at disposal. The idea then consists in feeding those temperature measures to

the controller and perform a simple Recursive Least Squares (RLS) fitting with several temperature models, as Figure 6.7 shows. Those can be "trained" offline by optimising different I-PIDs, with the uncertain parameter  $\omega$  varying around its a priori domain  $\omega \in \Omega$ .

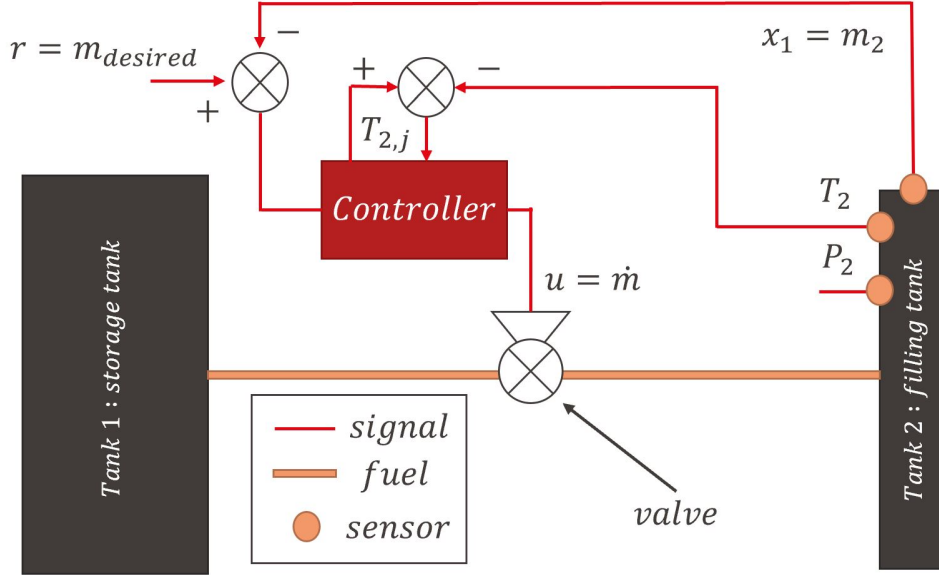


Figure 6.7: Block-diagram representation of the adaptive I-PID controller, including temperature and mass feedback

Now, the methodology used to train the different models will be explained. First, a relatively coarse grid for the uncertain parameter is defined as  $\omega_{grid} = \{\omega_1, \dots, \omega_j, \dots, \omega_M\}$ . Next, an optimal controller is trained for the most limiting parameter in order to find the maximum time-step needed to perform the simulation. Then,  $M$  optimal controllers are designed for each of the  $M$  parameters in  $\omega_{grid}$  by fixing the time-step at the value previously found. For each of these optimisations, both the temperature evolution and the controller parameters are saved. In what follows, the uncertain parameter considered will be the internal heat transfer coefficient  $UA_{IN}$ , taken as an example.

The adaptation algorithm which will be used here makes use of a certain form of the Recursive Least Square algorithm (RLS). The loss is formulated in Equation 6.9, with  $T(i)$  being a temperature measurement and  $\Phi$  representing the temperature model matrix, stated in Equation .

$$\mathcal{J}(i+1) = \mathcal{J}(i) + \lambda^n (T(i) - \Phi(i))^2 = \mathcal{J}(i) + \lambda^n \epsilon^2(i) \quad (6.9)$$

$$\Phi = \begin{bmatrix} T_1(1) & T_1(2) & T_1(3) & \dots & T_1(i) & \dots & T_1(N) \\ T_2(1) & T_2(2) & T_2(3) & \dots & T_2(i) & \dots & T_2(N) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ T_j(1) & T_j(2) & T_j(3) & \dots & T_j(i) & \dots & T_j(N) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ T_M(1) & T_M(2) & T_M(3) & \dots & T_M(i) & \dots & T_M(N) \end{bmatrix} \quad (6.10)$$

Now, the model can be chosen in a way which minimises this loss at each time step, recursively.

$$K_m^*(i) = \underset{j}{argmin} \mathcal{J}(i) \quad (6.11)$$

The  $\lambda$  term in Equation 6.9 is called the *forgetting factor* since it weighs how much the current loss "forgets" about previous losses, and thus represents the algorithm memory, in a certain sense. However, stated like this, the algorithm would be extremely responsive since it would adapt the controller each time the system response differs from the actual model. In order to prevent this effect, a *hysteresis time* can be set, thus delaying controller changes by a few time-steps. One can notice that this method is reminiscent of the Model Reference Control adaptation algorithm, as formulated in Fahmy et al. [37].

### 6.3.3 Adaptive I-PID controller training

First, the actual "training" of several IPID controllers must be performed in the aim of saving them and switching among them. Hence, the whole optimisation routine for IPID controllers described in Section 6.2.3 has to be launched for each parameter of the parametric grid. As an example, this section will present the full development and several simulations for an APID controller with parametric uncertainties ranging between  $UA_{IN} \in \{250, 350\} [W/m^2K]$ . The uncertainty discretisation step has been heuristically chosen to be equal to  $10 [W/m^2K]$ . Therefore, 11 IPIDs have been optimised for, each of them being trained for one specific internal heat transfer coefficient value. This amounts to solving the problem presented in Equation ?? 11 times, each time increasing  $UA_{IN}$  by  $10 [W/m^2K]$ . Although the optimisation procedure remains identical as before, the time-step cannot be optimised for anymore. This is due to the IPID switching time having to be fixed in order for both end of the control input trajectory to match, as explained in Section 6.2.3. Even though time is not minimised anymore, mass to reference minimisation has been kept in order to steer the optimisation problem towards fast reference reaching anyways.

$$\begin{aligned}
 & \underset{\epsilon, \eta, m_2}{\text{minimize}} \quad \mathcal{J}(\mathcal{M})_{PID} = Q_\epsilon \epsilon^2 + Q_\eta \eta^2 + (r - m_2)^T Q_m (r - m_2) \\
 & \text{subject to : } m_{i+1} = f(x_{N_k}(m_i, u_i), p_{N_k}), \quad i = 1, \dots, N-1 \quad (\text{Dyn. continuity}) \\
 & \quad T_{2,i} \leq 338, \quad i = 1, \dots, N \quad (\text{Tank 2 temp.}) \\
 & \quad P_{2,i} \leq 300 + \eta, \quad i = 1, \dots, N \quad (\text{Tank 2 pressure}) \\
 & \quad 0 \leq \eta \leq 30, \quad (\text{Pressure slack}) \\
 & \quad r - \epsilon \leq m_{2,N-n_\epsilon:N} \leq r, \quad (\text{Mass reference}) \\
 & \quad 0 \leq \epsilon \leq 0.01r, \quad (\text{Mass slack}) \quad (6.12) \\
 & \quad u_i \geq 0, \quad i = 1, \dots, N-1 \quad (\text{Input positiveness}) \\
 & \quad u_i \leq \dot{m}_{\max}(\rho_{1,i}, u_{1,i}), \quad i = 1, \dots, N-1 \quad (\text{Input saturation}) \\
 & \quad u_{t_1} = u_{t_1+1}, \quad i = 1, \dots, N-1 \quad (\text{Input continuity}) \\
 & \quad h = 1, \quad i = 1, \dots, N-1 \quad (\text{Time - step fixed}) \\
 & \quad \mathcal{M} \in \mathcal{X} \quad (\text{Optimisers bounds})
 \end{aligned}$$

The controllers and temperature trajectories resulting from the aforementioned procedure can be viewed on Figure 6.8. As expected, the temperature rise happens to be steeper each time the internal heat transfer coefficient is increased. This controllers set will be used in what follows.

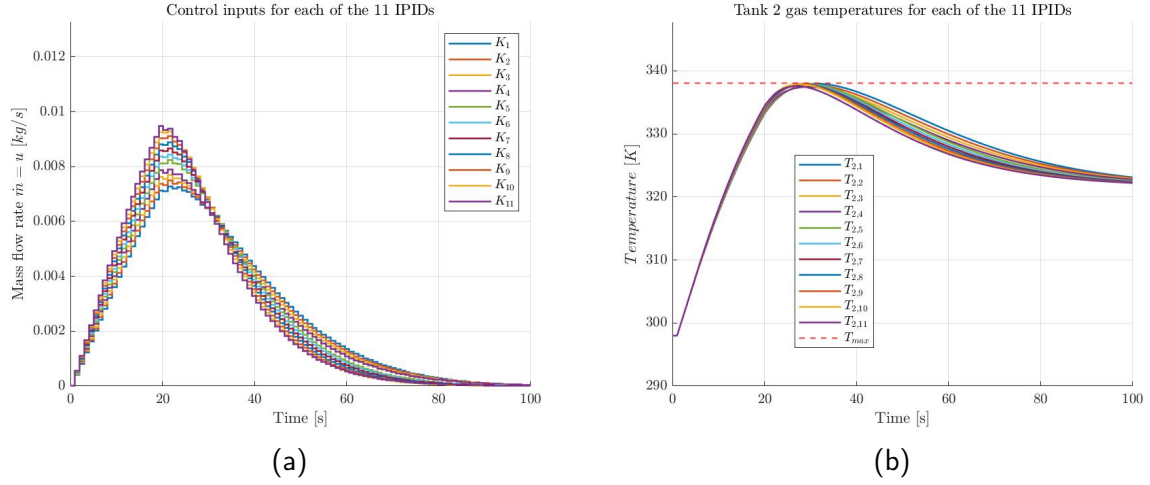


Figure 6.8: Control input and tank 2 gas temperature trajectories for each of the 11 IPIDs trained for the current APID. Please note that  $UA_{IN} = 250 \text{ [W/m}^2\text{K]} \Rightarrow \{K_1; T_{2,1}\}$  and  $UA_{IN} = 350 \text{ [W/m}^2\text{K]} \Rightarrow \{K_{11}; T_{2,11}\}$ .

#### 6.3.4 Adaptive I-PID controller simulations

A first test on the adaptive PID scheme was performed using a nominal controller optimised for  $UA_{IN} = 300 \text{ [W/m}^2\text{K]}$  (so the same as for the simulation shown on Figure 6.6) but with a real internal heat transfer coefficient of  $UA_{IN} = 250 \text{ [W/m}^2\text{K]}$ . Figure 6.9 presents these results. So, if there was no adaptive scheme, the resulting trajectory would have substantially overshoot the temperature limit, as shown with the nominal trajectory on Figure 6.9 (a). However, if the adaptive feature is set on, one can observe on Figure 6.9 (c) that there is a clear controller switch at around 10 [s] of simulation, as well as a few other, all indicated with a black vertical line. These correspond to instants where the algorithm knows that it has sufficiently waited to be aware that there is definitely a variation of the temperature trajectory compared to the one the system was supposed to follow in the first place (i.e. the nominal trajectory). The term "sufficiently" conveys the notion of switching hysteresis, which effectively consists in waiting for several time-steps before switching controller, as long as the controller choice for each of these time-step stayed the same. Also, Figure 6.9 (c) shows that the final tank mass happens to perfectly match its reference of 267 [g], even though there are several controller switches. Furthermore, no mass overshoot occurs since the high derivative term prevents it.

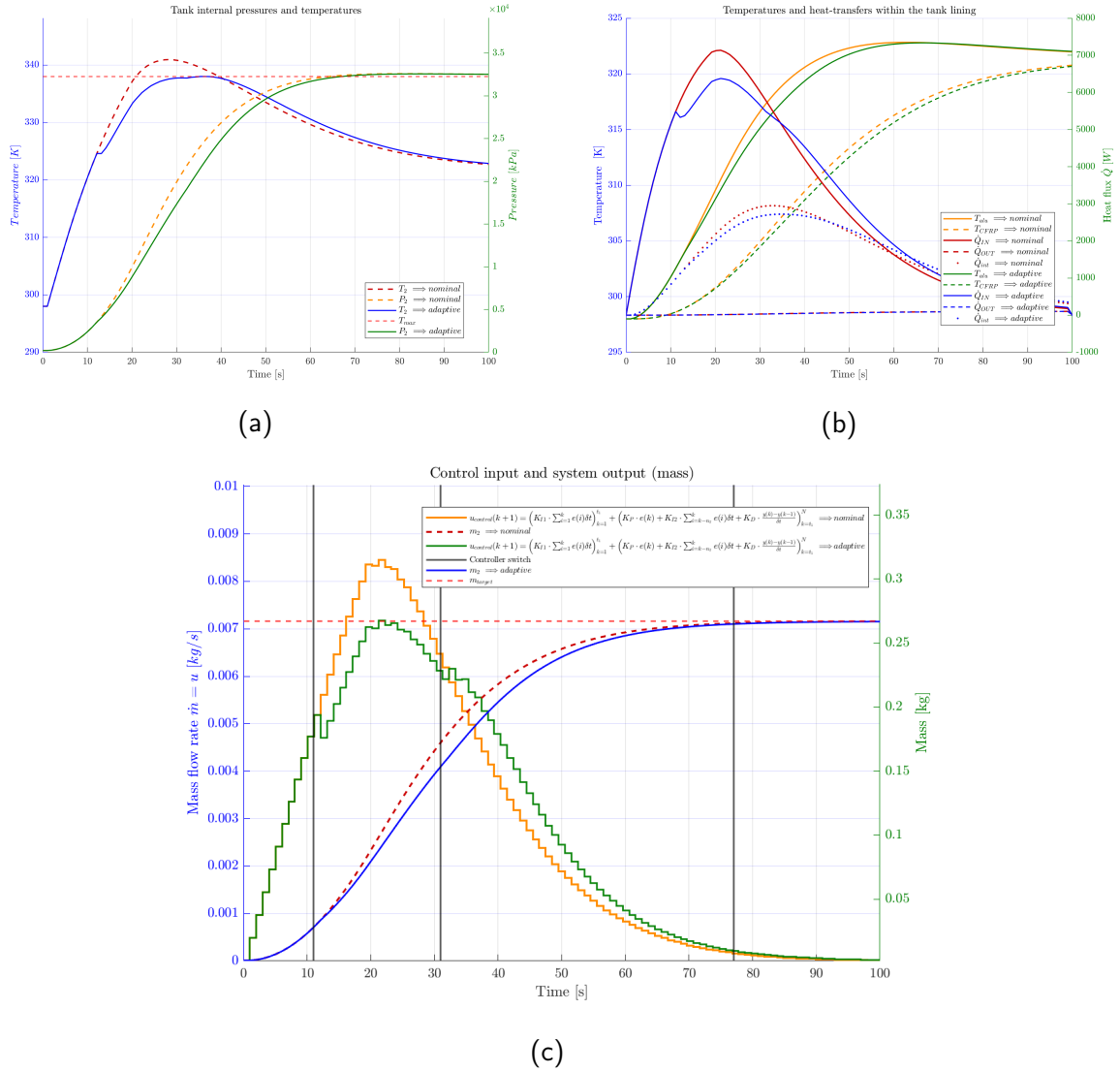


Figure 6.9: Simulation performed with a nominal controller optimised for  $UA_{IN} = 300$  [W/m<sup>2</sup>K] and with a real internal heat transfer coefficient of 250 [W/m<sup>2</sup>K]. The APID hyper-parameters have been set to  $\lambda = 0.8$  and  $n_h = 5$ .

Table 6.2 allows a better understanding of the logic actions performed by the switching algorithm, where the hysteresis can clearly be observed, preventing an immediate action and experimentally set to  $n_h = 5$  here.

Table 6.2: Logic actions performed by the algorithm during the simulation

Time-step	1	2	3	4	5	6	7	8	9	10	11	12	13	...
Optimal model	2	3	4	6	9	11	11	11	11	11	11	11	11	...
Change ?	no	no	no	no	no	no	wait	wait	wait	yes	no	no	no	...

Now, the adaptive controller can handle fixed uncertain parameters, however the internal heat transfer coefficient is more likely to continuously vary with respect to time. Hence, another simulation was performed, this time with a linearly varying internal heat transfer coefficient, as shown on Figure 6.10 (c). Its value peaks right at the time where the mass flow rate of Figure 6.10 (b) increases a lot, at the end of the first integrator working period  $t_1$ . Again, the adaptive scheme appears to yield excellent performance. Nevertheless, appropriate hyper-parameters tuning was performed heuristically

since the  $n_h = 5$  proposed before turned out to be too short for this geometry of parametric variations.

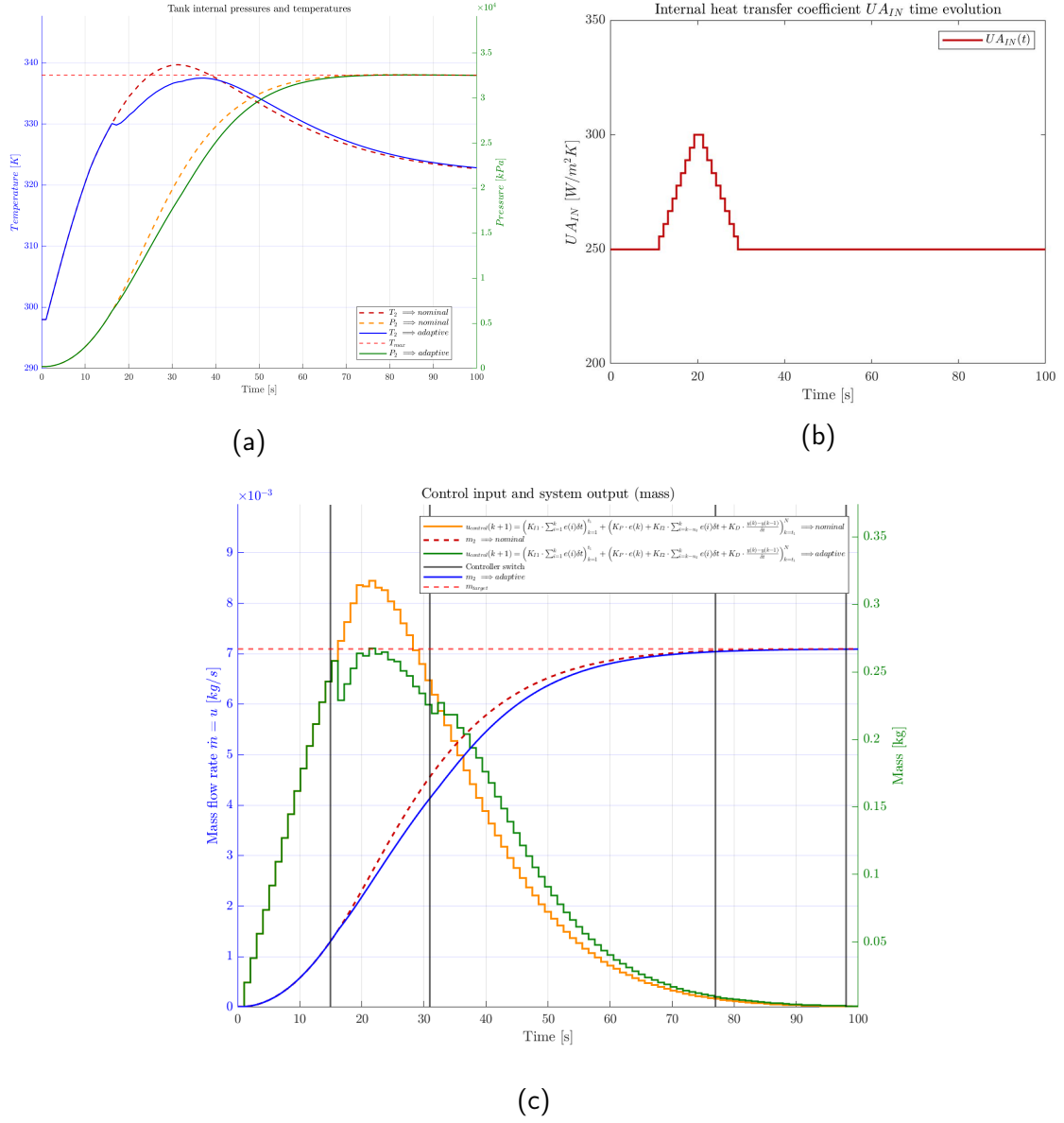


Figure 6.10: Simulation performed with a nominal controller optimised for  $UA_{IN} = 300$  [W/m<sup>2</sup>K] and with a real internal heat transfer coefficient (b) linearly varying between  $\{250, 300\}$  [W/m<sup>2</sup>K] at the time the mass-flow rate is maximum (c). The APID hyper-parameters have been set to  $\lambda = 0.8$  and  $n_h = 9$ .

A last simulation was performed using a slightly different internal heat transfer profile this time, as presented on Figure 6.11. Although the  $UA_{IN}$  profile differs from the previous simulation, the adaptation algorithm still manages to enforce appropriate changes to the controller without having to modify  $\lambda$  or  $n_h$ . Hence, sufficiently many simulations would have to be executed in the aim of finding the right hyper-parameters and avoiding having to modify them on the run. This can certainly be considered as one of the primary limitation of the present algorithm.

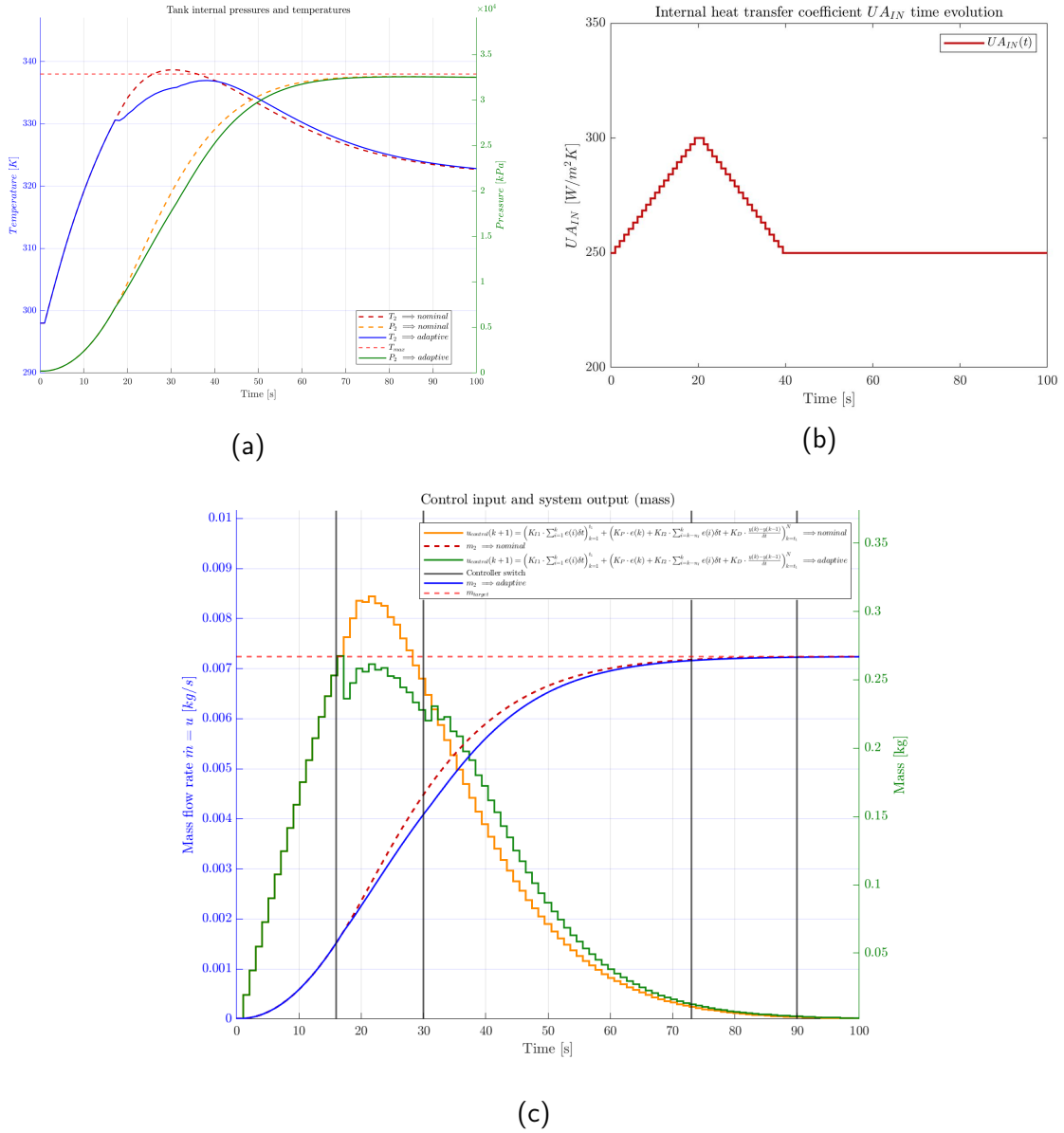


Figure 6.11: Simulation performed with a nominal controller optimised for  $UA_{IN} = 300$  [W/m<sup>2</sup>K] and with a real internal heat transfer coefficient (b) linearly varying between  $\{250, 300\}$  [W/m<sup>2</sup>K] at the time the mass-flow rate is maximum (c), but with a lower rate of increase compared to Figure 6.10. The APID hyper-parameters have been set to  $\lambda = 0.8$  and  $n_h = 9$ .

## 6.4 Discussion and further improvements

Throughout this whole section, mainly 3 different control strategies were proposed in the aim of sticking to the system limits. In light of their respective performance and limitations, one can now weigh the pros and cons to determine which could be the most appropriate for the system at hand. First, MPC-like control inputs can yield the best performance and satisfy the constraints while maintaining the temperature on the edge. Since actually using MPC for controlling the system would certainly be too risky, an interesting idea could consist in developing an algorithm which automatically switches the plant to a simple, conservative feedback controller each time the MPC fails, instead of dead-stopping the system. For that, an a priori knowledge of a safe working state for the system would be required, and the feedback law reference could be set onto this working condition. Such a



feedback law could very well be a PID, designed with the method proposed in Section 6.2.1. Indeed, even though this design does not provide robust constraint enforcement, it would not be an issue since it is only supposed to take the lead when the MPC routine fails.

Another application of the proposed optimal PID design would consist in implementing it as an initial controller for performing closed-loop identification. As a matter of fact, open-loop identification might be dangerous as well as costly for such a system since constraint violation would also lead to system failure, resulting in gas leakage through the damaged tank wall. Therefore, adding a feedback controller could enable to identify the unmodeled system dynamics, such as more accurate heat transfer dynamics, as well as benefiting the controller optimisation since the controller real performance would be tested.

Next, the APID controller proposed in Section 6.3.2 provides the robustness which the simple PID lacked of. However, although the adaptive controller provides excellent performance, this has to be nuanced. One important assumption which was made consists in supposing that all feedback signals (such as the temperature measurements) are well-filtered, resulting in smooth, de-noised signals. This is required for the adaptive scheme to work, and studying the impact of measurement noise on its reactivity might be essential. Next, in the probable case where several parameters are subject to wide ranges of uncertainty, it might be necessary to train a huge number of models in order to cover the full multi-dimensional uncertainty set. This can also be considered as a limitation of the current algorithm since having close to another temperature model curves would probably confuse the algorithm. This could also be relatively time-consuming as well as challenging regarding the implementation. Of course, both the forgetting factor and the hysteresis number allow one to obtain different controller behaviours, but real-life testing might yield other conclusions. Finally, finding a well-suited combination for both the forgetting factor  $\lambda$  and the hysteresis number  $n_h$  might be slightly tricky since the worst case possible must be identified.

## 7 Conclusion

Throughout this work, a hydrogen fast-filling system has been studied. First, a thorough model was proposed, with the distinctiveness of using a NIST database instead of an analytical equation of state. Indeed, the literature study has shown that analytical or CFD models were generally preferred. However, it was demonstrated that using such database might provide excellent accuracy. Therefore, the database was implemented within both simulation and optimisation routines and then validated using experimental data. Next, an optimal control problem was proposed in the aim of finding an optimal control law for the system at hand. Then, three distinct control strategies have been established.

First, an optimal free control input was optimised for, yielding excellent performance and letting glimpse what performance MPC might offer. However, MPC was not used for safety reasons. Next, an optimal feedback law was designed. Under the form of a simple PID controller, and then under the form of two controllers in series, this law delivers very good performance, enabling a filling time of around 70 [s] while conforming to the system constraints. It was however discussed that model uncertainties could deteriorate the controller performances. Therefore, an adaptive control scheme was suggested in the aim of switching the controller when the temperature constraints are about to be broken. Although sub-optimal, this adaptive algorithm for our controller also exhibits decent performance. Nevertheless, the main downsides were the scheme tuning and implementation, which might pose a challenge for the user.

Therefore, this work has shown that many control strategies might offer good performance. However, due to model uncertainties, online optimisation routines might still be the best option. The controller optimisation methods proposed here could be used for developing a background controller,

thus ensuring the safe operation of the MPC-based main controller.

## References

- [1] M. Heitsch, D. Baraldi, and P. Moretto. *Numerical investigations on the fast filling of hydrogen tanks. International Journal of Hydrogen Energy* 36, (2011) 2606-2612, 2010.
- [2] M. Cristina Galassi, Daniele Baraldi, Beatriz Acosta Iborra, and Pietro Moretto. *CFD analysis of fast filling scenarios for 70 MPa hydrogen type IV tanks. International Journal of Hydrogen Energy* 37, (2012) 6886-6892, 2012.
- [3] M. C. Galassi, B. Acosta-Iborra, D. Baraldi, C. Bonato, F. Harskamp, N. Frischaua, and P. Moretto. *Onboard Compressed Hydrogen Storage: Fast Filling Experiments and Simulations. Energy Procedia* 29, (2012) 192 – 200, 2012.
- [4] Daniele Melideo, Daniele Baraldi, Maria Cristina Galassi, Rafael Ortiz Cebolla, Beatriz Acosta Iborra, and Pietro Moretto. *CFD model performance benchmark of fast filling simulations of hydrogen tanks with pre-cooling. International Journal of Hydrogen Energy* 39 (2014), 4389-4395, 2013.
- [5] Daniele Melideo and Daniele Baraldi. *CFD analysis of fast filling strategies for hydrogen tanks and their effects on key-parameters. International Journal of Hydrogen Energy* 40 (2015), 735-745, 2014.
- [6] Jun Liu, Shuiying Zheng, Zhixin Zhang, Jinyang Zheng, and Yongzhi Zhao. *Numerical study on the fast filling of on-bus gaseous hydrogen storage cylinder. International Journal of Hydrogen Energy* 45 (2020), 9241-9251, 2020.
- [7] Ji-Qiang Li, No-Seuk Myoung, Jeong-Tae Kwon, Seon-Jun Jang, and Taeckhong Lee. *A Study on the Prediction of the Temperature and Mass of Hydrogen Gas inside a Tank during Fast Filling Process. Energies* 2020, 13, 6428, 2020.
- [8] Li Wang, Feng Ye, Jinsheng Xiao, Pierre Bénard, and Richard chahine. *Heat transfer analysis for fast filling of on-board hydrogen tank. Energy Procedia* 158 (2019) 1910-1916, 2019.
- [9] Peter Lloyd Woodfield, Masanori Monde, and Yuichi Mitsutake. *Measurement of Averaged Heat Transfer Coefficients in High-Pressure Vessel during Charging with Hydrogen, Nitrogen or Argon Gas. Journal of Thermal Science and Technology*, 2006.
- [10] FJinxing Guo, Jian Yang, Yongzhi Zhao, Xiangmin Pan, Lifang Zhang, Lei Zhao, and Jinyang Zheng. *Investigations on temperature variation within a type III cylinder during the hydrogen gas cycling test. International Journal of Hydrogen Energy* 39, (2014) 13926-13934, 2014.
- [11] Toshihiko Ooi, Takafumi Iijima, Koichi Oshino, Hiroyuki Mitsuishi, and Shogo Watanabe. *Hydrostatic Pressure Burst Test and Pressure Cycling Test of Compressed Hydrogen Tanks. World hydrogen energy conference* 16, 2006.
- [12] Peter L. Woodfield, Masanori Monde, and Toshio Takano. *Heat Transfer Characteristics for Practical Hydrogen Pressure Vessels Being Filled at High Pressure. Journal of Thermal Science and Technology*, 2007.
- [13] Md. Tawhidul Islam Khan, Masanori Monde, and Toshiaki Setoguchi. *Hydrogen Gas Filling into an Actual Tank at High Pressure and Optimization of Its Thermal Characteristics. Journal of Thermal Science Vol.18*, 2009.
- [14] Emmanuel Ruffio, Daniel Petit, and Didier Saury. *Thermodynamic analysis of hydrogen tank filling. Effects of heat losses and filling rate optimization. International Journal of Hydrogen Energy*, 2014.
- [15] P.L. Woodfield and M. Monde. *A Thermodynamic Model for a High-Pressure Hydrogen Gas Fill-*

- ing System Comprised of Carbon-Fibre Reinforced Composite Pressure Vessels. 17th Australasian Fluid Mechanics Conference*, 2010.
- [16] M. Deymi-Dashtebayaz, M. Farzaneh-Gord, N. Nooralipoor, and H. Niazmand. *The complete modelling of the filling process of hydrogen on board vehicle cylinders. Brazilian Journal of Chemical Engineering*, 2016.
  - [17] Fernando Olmos and Vasilios I. Manousiouthakis. *Hydrogen car fill-up process modeling and simulation. International Journal of Hydrogen Energy*, 38 (2013) 3401-3418, 2013.
  - [18] Sitra Pregassame, Frédéric Barth, Laurent Allidieres, and K. Barral. *Hydrogen refuelling station: filling control protocols development. WHEC 16 / 13-16*, 2006.
  - [19] Jitian Liu Xian Wu, Jianwang Shao, and Guoming Deng. *Fast filling strategy of type III on-board hydrogen tank based on time-delayed method. International Journal of Hydrogen Energy* (2021), 2021.
  - [20] Jinyang Zhenga, Jianjun Yea, Jian Yanga, Ping Tanga, Lei Zhaoa, and Maradee Kern. *An optimized control method for a high utilization ratio and fast filling speed in hydrogen refueling stations. International Journal of Hydrogen Energy* 35, (2010) 3011-3017, 2009.
  - [21] Yunfeng Bai, Caizhi Zhang, Hao Duan, Shangfeng Jiang, Zhiming Zhou, Didier Grouset, Mingjun Zhang, and Xuefeng Ye. *Modeling and optimal control of fast filling process of hydrogen to fuel cell vehicle. Journal of Energy Storage*, 2021.
  - [22] Fernando Olmos and Vasilios I. Manousiouthakis. *Gas tank fill-up in globally minimum time: Theory and application to hydrogen. International Journal of Hydrogen Energy* 39, (2014) 12138-12157, 2014.
  - [23] W.G.Kortekaas, C.J.Peters, and J.deSwaanArons. *Joule-Thomson expansion of high-pressure-high-temperature gas condensates. Fluid Phase Equilibria* 139, 1997.
  - [24] M. W. T. Pratt. *The Temperature of Inversion of the Joule-Kelvin Effect for Hydrogen. Nature* 65, 1902.
  - [25] Honggang Chen, Jinyang Zheng, Ping Xu b, Lei Li, Yanlei Liu, and Haiyan Bie. *Study on real-gas equations of high pressure hydrogen. International Journal of Hydrogen Energy* 35 (2010) 3100-3104, 2009.
  - [26] Eric W. Lemmon, Marcia L. Huber, and Mark O. McLinden. *NIST Standard Reference Database 23: Reference Fluid Thermodynamic and Transport Properties-REFPROP, Version 9.1*, 2013. URL <https://www.nist.gov/srd/refprop>.
  - [27] Brian J. Cantwell. *Fundamentals of Compressible Flow. Course notes, Stanford University, Stanford, California 94305*, 2018.
  - [28] Mahmood Farzaneh-Gord, Mahdi Deymi-Dashtebayaz, Hamid Reza Rahbari, and Hamid Niazmand. *Effects of storage types and conditions on compressed hydrogen fuelling stations performance. International Journal of Hydrogen Energy* 37, 2012.
  - [29] Moritz Diehl and Sébastien Gros. *Numerical Optimal Control*. 2017.
  - [30] H. G. Bock and K. J. Plitt. *A multiple shooting algorithm for direct solution of optimal control problems. IFAC 9th Triennial World Congress*, 1984.
  - [31] P.T. An, N.N. Hai, and T.V. Hoai. *Direct multiple shooting method for solving approximate shortest path problems. Journal of Computational and Applied Mathematics*, 2013.
  - [32] Martin J. Gander. *50 Years of Time Parallel Time Integration. Multiple Shooting and Time*

- Domain Decomposition Method, 2015.
- [33] Shi-Chung Chang, Tsu-Shuan Chang, and Pter B. Luh. *A Hierarchical Decomposition for Large-scale Optimal Control Problems with Parallel Processing Structure*. Automatica, Vol. 25, No 1, 1989.
  - [34] Said Munzir, Vera Halfiani, and Marwan. *An optimal control solution using multiple shooting method*. 2010 Mathematics Subject Classification, 2012.
  - [35] Joel A E Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. *CasADi – A software framework for nonlinear optimization and optimal control*. Mathematical Programming Computation, In Press, 2018.
  - [36] Dr. Alireza Karimi. *Advanced Control Systems*. Course notes, EPFL, 1015 Lausanne, Suisse, 2020.
  - [37] Rania A. Fahmy, Ragia I. Badr, , and Farouk A. Rahman. *Adaptive PID Controller Using RLS for SISO Stable and Unstable Systems*. Advances in Power Electronics, Volume 2014, 2014.