Modelling, Identification and Control of a Renewable Hydrogen Production System for Mobility Application

Master Thesis

Author(s):
Laaksonlaita, Timo

Publication date:
2021-01

Permanent link:
https://doi.org/10.3929/ethz-b-000484976

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

AUTOMATIC CONTROL LABORATORY

Modelling, Identification and Control of a Renewable Hydrogen Production System for Mobility Application

Master Thesis

Timo Laaksonlaita

January 2021

Supervision

Marta Fochesato - IfA
Philipp Heer - Empa
Prof. Dr. John Lygeros - IfA

Abstract

Hydrogen-fueled cars are a promising technology for reducing CO2 emissions in the mobility sector. This thesis develops a stochastic receding horizon controller for a hydrogen refueling station that operates the electrolyzer and compressors such that the usage of renewable energies, in this case photovoltaic (PV) energy, is maximized and the usage of grid power is minimized. For this, a model of the electrolyzer and the compressors is derived and identified from data. Historical hydrogen demand data is fitted to a stochastic model such that samples for a scenario-based stochastic MPC can be generated. The available PV power is predicted by a neural network using weather forecast data. These models and predictions are combined into a mixed-integer linear program, which is solved in real-time every 10 minutes. To allow for better scalability of the problem with respect to the number of scenarios, a heuristics to decouple the scenarios in the optimization problem is introduced. The resulting optimization problem is converging within approximately 10 seconds with a prediction horizon length of 24 hours and 96 timesteps. An evaluation of the stochastic MPC and different deterministic controllers shows that the stochastic MPC performs equally well to its deterministic counterpart as long as the storage tanks are not close to their lower limit. In the case of almost empty storage tanks, the stochastic MPC provides an input sequence which leads to a probabilistic satisfaction of the system constraints. Adjustments to the terminal weights in the MPC are proposed in order to increase the performance of the MPC in such situations.

Contents

Chapter 1

Introduction

With the recent medial coverage of the climate change, actions against it are on the uprise. One of these actions aims at reduction of CO_2 coming from the mobility sector, for example by a phased-out of fossil fuelled vehicles [1]. Apart from electric cars which take their electricity from a battery, fuel cell cars are also on the upcoming [2]. They utilize compressed hydrogen in a fuel cell in order to convert it to electricity which can then be used to drive an electric motor. Compared to battery driven cars, this has the advantage that the energy-to-weight ratio of hydrogen is much higher than that of batteries or even of gasoline [3] [4]. However, they present the main disadvantage that the production of the required hydrogen is quite inefficient. This is due to the inefficient conversion of electricity into hydrogen [5] and the required compression of the hydrogen in order to get it to a higher pressure level, such that its energy density is high enough to be used on-board.

The goal of this thesis is to optimize the production of compressed hydrogen in a hydrogen refuelling station by incorporating available renewable energy, in this case a photovoltaic (PV) system installed on the roof of the refuelling station. This case study mirrors the move demonstrator [6] at Empa, Dübendorf (CH).

A simplified setup of the system is shown in Figure 1.1. The main component for hydrogen production is the electrolyzer, which converts electricity and water into hydrogen and oxygen. Next to this, storage tanks are necessary as the electrolyzer is not able to produce hydrogen fast enough to guarantee fast refuelling. Additionally, compressors stages are necessary to bring the hydrogen to high enough pressure such that it can be used to refuel vehicles, where usual fuel-cell cars are charged up to a pressure of 700bar [7].

The optimization is done by a receding horizon controller, which incorporates prediction of available PV power, the hydrogen demand and the system itself, e.g. the storage tanks. By optimizing over forward integrated system states, the optimal input sequence over the prediction horizon is found. This problem is then re-solved after each timestep, in order to incorporate new state information. In a first step we generate predictions which can be used in the setting of an optimization problem. The end-goal is to get control inputs for the electrolyzer and the compressor stages such that the production of the compressed hydrogen is optimal in the sense of using as little grid electricity as possible, while satisfying a stochastic hydrogen demand and incorporating available PV power. In order to capture the stochasticity of the hydrogen demand the optimization problem takes several different hydrogen demand samples into account resulting in a scenario-based MPC.

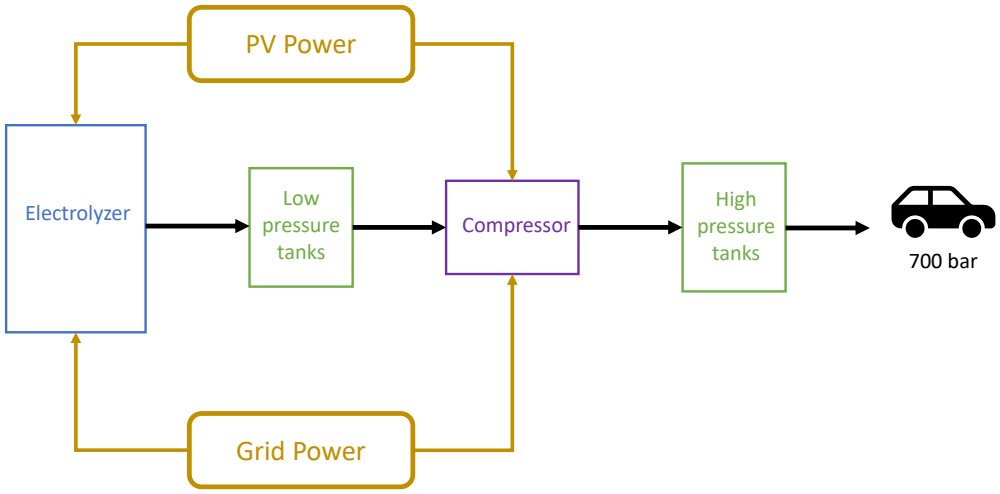

Figure 1.1: A general overview of the hydrogen refuelling facility. The black lines correspond to hydrogen flow, while the brownish lines show an electric current. It consists of the electrolyzer and storage tanks at different pressure levels with compressors between them, where the amount of differently pressured tanks can also be larger than two.

The thesis is split into 6 parts. In the remainder of Chapter 1, the state of the art for the different subsystems and our contribution to them is summarized. In Chapter 2, the electrolyzer, compressors and storage tanks are modelled. The unknown model parts are fitted with experimental data from the real system at Empa. Chapter 3 considers how the available PV power is predicted by using weather forecast data which is available from MeteoSwiss [8]. To this end, different models are fitted to historical data and the results are compared to each other. In Chapter 4, the hydrogen demand is modelled as a stochastic quantity which enter the system as a disturbance. In order to incorporate it into the optimization problem, a hydrogen demand model is derived and samples are generated from it. Chapter 5 shows how by using a scenario-based stochastic optimization approach, the hydrogen demand is taken into account by performing the optimization over various hydrogen demand samples. This is done by incorporating all previously derived predictions and models into the optimization problem which computes the optimal control inputs for the overall system by using a mixed-integer linear program (MILP) solver. Lastly, Chapter 6 concludes the results of this thesis and and gives an outlook to future work that can be done.

1.1 State of the art

Previous research on electrolyzer modelling [9], [10] show the possibility of modelling electrolyzers in an accurate and precise manner by including pressure and temperature dynamics and by directly controlling the power usage of the electrolyzer. However, as they consider much shorter prediction horizons or only a deterministic setting, they can use more detailed models in the optimization problem without running into limitations regarding the convergence time of the optimization problem. This thesis aims at developing a stochastic one-day ahead MPC scheme which needs to run in real-time, therefore

the electrolyzer model necessarily needs to be simplified in order for the optimization problem to be solved in real-time. Furthermore, the electrolyzer available for this thesis is different from the electrolyzer used in [9] and [10], in the aspect that it does not provide direct control of the power usage of the electrolyzer. The details of this are discussed in Section 2.2.
Additionally, we developed a model predictive controller which optimizes the operation of a whole refuelling station. This means that not only the electrolyzer, but also relevant compressor stages are included in the problem formulation, leading to an overall better performance of the whole system. To the best of our knowledge, a combined control scheme for both electrolyzer and compressor stages in a stochastic setting has not been proposed in the literature yet.

Chapter 2

Modelling

As the the name would suggest, a model predictive control scheme requires a model of the system. The model is used to predict future states of the system and can be used in an optimization problem to find optimal input values for the system given some objective function. On a first glance, it might be appealing to find an extremely detailed model in order to minimize the prediction error of the model. However, each added detail has to be traded off against an increase in the convergence time of the optimization problem that it induces.

In a first step, the general construction of the whole system is introduced and then the different subsystems are considered.

2.1 Introduction to the system

The main components of the move demonstrator at Empa are the electrolyzer, the different storage tank stages, the refuelling station and the PV system installed on the roof. An overview can be seen in Figure 2.1. The storage tank stages consist of 30bar, 390bar and 900bar storage tanks: the electrolyzer is connected to the 30bar tank and the refuelling station to the 900bar tanks. The various storage tanks are connected through compressors: one is placed at the input of the 390bar tank and one is placed at the input of the 900bar tank. The 390bar compressor only takes hydrogen from the 30bar tanks, while the 900bar compressor has the possibility to take hydrogen from both the 30bar or the 390bar tank, depending which one has the higher pressure.

2.2 Electrolyzer low-level control loop

The electrolyzer itself does not allow for a direct control of the stack currents or even the overall power consumption as it might be the case for other electrolyzers [9], [10]. The low-level controller implemented within the electrolyzer is controlling the pressure at the output of the electrolyzer to a certain set-point value as displayed in Figure 2.2. In order to control the electrolyzer, this pressure is manipulated. A mass-flow controller is installed at the output of the electrolyzer enabling the control of the mass flow and therefore the indirect control of the output pressure of the electrolyzer. Assuming that we want the electrolyzer to produce more hydrogen, we have to increase the set-point of the mass-flow controller allowing more hydrogen to flow through it. This will decrease the pressure at the output of the electrolyzer, which will then trigger the low-level controller

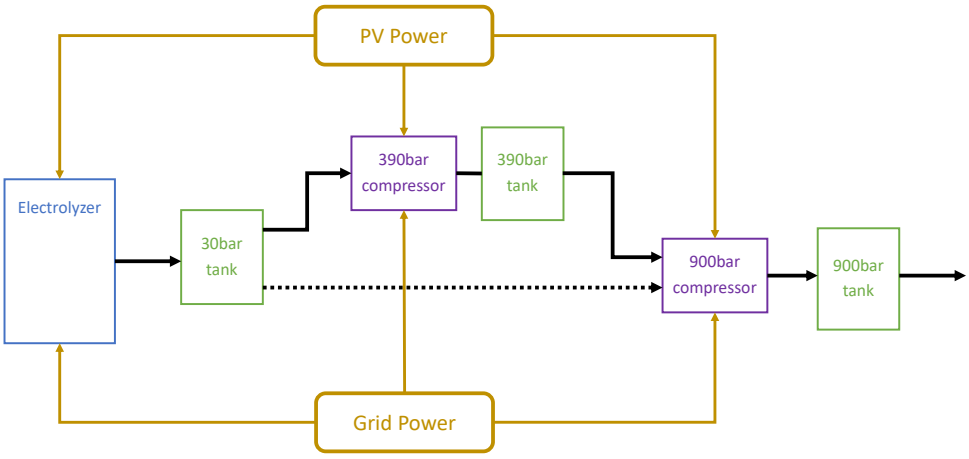

Figure 2.1: An overview of the system without the refuelling station. The black lines correspond to hydrogen flow, while the brownish lines correspond to electric currents.

of the electrolyzer to increase its production rate, i.e. increase the stack currents. Note that no detailed information about the underlying low-level controller of the electrolyzer is available apart from the fact that it controls the output pressure to a set-point. Both the mass flow controller and the low-level controller of the electrolyzer are not perfect and take a certain time to converge to the desired set-point.

In order to get an accurate model of the whole system over all timescales, these low-level controllers would need to be identified and included in the model. However, there are multiple reasons why this is not the best approach for deriving a useful model which can be used in a receding horizon controller.
Firstly, the pressure dynamics at the output of the electrolyzer are extremely fast compared to the timescale of the MPC. Including them in the model would require small discretization steps and therefore small timesteps of the MPC in general. This will inevitably lead to an extreme increase in the convergence time of the optimization problem, such that the MPC cannot be run in real-time, given the desired prediction horizon length of one or two days. Additionally to that, these fast dynamics do not have any major influence on the performance once the system is in steady-state, making them only a burden during such times.
Secondly, the identification of the low-level controllers and especially the dynamics behind them is extremely difficult. This is due to the fact that no temperature or pressure measurements inside the electrolyzer are available to us. As the relationship between stack current and hydrogen production is dependent on the stack temperature and the pressure inside the electrolyzer, this requires the estimation of the dynamics without having direct measurement of all state variables and without being able to directly control the inputs, i.e. the stack currents. Even though this is theoretically possible, it would be enormously time consuming.

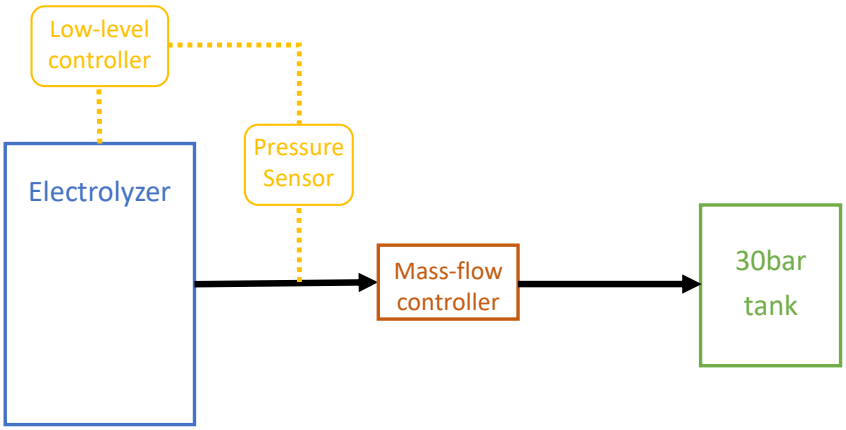

Figure 2.2: The low-level controller of the electrolyzer is measuring the pressure at the output of the electrolyzer and is adjusting the stack currents of the electrolyzer in order to keep this pressure at a reference set-point value.

Therefore, in this thesis, the low-level controllers will be handled as a part of a cascaded control structure, where it is assumed for the construction of the MPC that they provide perfect tracking. This is a reasonable assumption as long as the timescales of the low-level controller and the mass-flow controller are much smaller than the timescale of the MPC, which is true as long as the timesteps of the MPC are not too small. In the following, certain aspects of the non-perfect tracking are discussed and steps to minimizing the error arising from it are discussed.

2.2.1 Non-perfect tracking of low-level controllers

In practice, the assumption of a perfect tracking for the low-level controllers is never fulfilled, however, for most times it is accurate enough such that it is reasonable to use it. In the next few paragraphs we are going to look at situations where the assumption is heavily violated and which, when they occur often enough, would lead to a non-negligible error.

Electrolyzer Start-Up

The start-up procedure of the electrolyzer takes around 5-6 minutes during which the electrolyzer does not produce the full amount of hydrogen anticipated. The mass flow at the output of the electrolyzer during a start-up procedure can be seen in Figure 2.3. By not modelling the transient, the model expects full hydrogen production as soon as it is turned on.

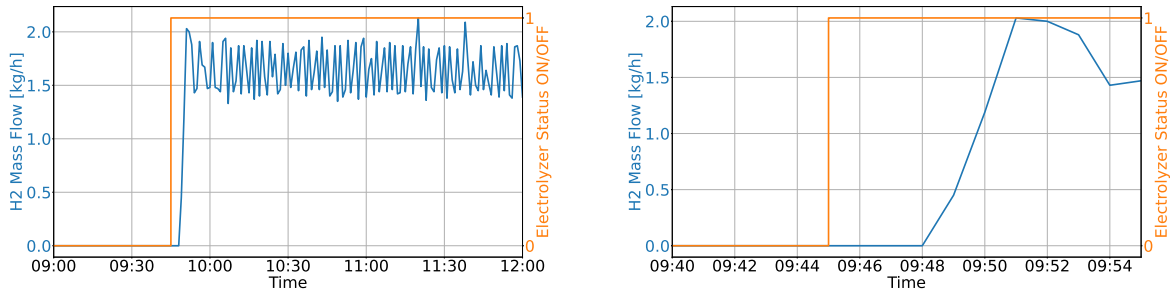

Figure 2.3: The convergence of the mass flow at the output of the electrolyzer during a start-up sequence. The left picture provides a general overview of it, while the right picture provides a close-up look at the mass flow right after giving the start-up command to the electrolyzer. There is a gap of around 3 minutes until the electrolyzer has started up and after which it takes another 3 minutes until the system has reached the peak value for the mass flow.

An almost full 30bar tank

The fact that the electrolyzer is controlled by the pressure at its output leads to a complicated behaviour if the pressure in the 30bar tank gets close to the pressure at the output of the electrolyzer, which is around 30bar. In this case the mass-flow controller has no possibility to generate a large enough mass flow, as the pressure gradient is too small. This will lead to an increase in the pressure at the output of the electrolyzer and therefore an decrease in the hydrogen production. This phenomena is shown in Figure 2.4.

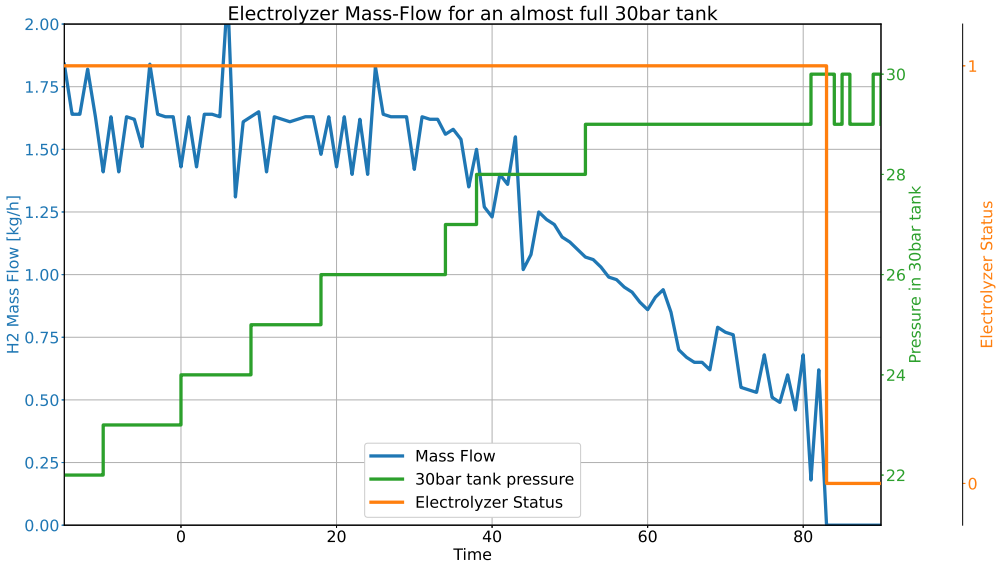

Figure 2.4: The mass flow (blue), 30bar tank pressure (green) and the electrolyzer status (orange) for a situation where the 30bar tank is almost full. Once the pressure in tank reaches 27 bar, the hydrogen production decreases.

2.3 Electrolyzer modelling

The electrolyzer model should capture the relationship between the control input of the mass-flow controller u and the rate of hydrogen production of the electrolyzer $\dot{m}_{prod}$. Assuming a perfect tracking of both low-level controllers, this can be split up into determining the relationship between the control input u and the power consumption of the electrolyzer p^{el} as well as the relationship between the power consumption of the electrolyzer p^{el} and the hydrogen production $\dot{m}_{prod}$:

$$\dot{m}^{prod} = f(u) = f_1(p^{el}) = f_1(f_2(u)), \tag{2.1}$$

where $p = f_2(u)$ and f, f_1 and f_2 are continuous scalar functions.

2.3.1 Model identification and piecewise affine approximation

In order to find the function f, the relationship between control input u and the hydrogen production rate $\dot{m}_{prod}$, the electrolyzer is run at different control input values and the mass flow for each of them is measured. This is done until the system has converged to a steady-state value and has remained there for a certain amount of time. The values over the time interval are then averaged, as the quantities are generally strongly oscillating. Strictly speaking the signal would not have reached a steady-state value if it oscillates. In this thesis the term steady-state is used to describe a state to which a signal has converged on average. As reference, the oscillations of the mass flow are shown in Figure 2.3, while figure 2.5 shows oscillations in the power consumption of the electrolyzer. As the functions f, f_1 and f_2 are required to be used in a mixed-integer linear problem, they need to be approximated by piecewise affine functions. The breakpoints are chosen such that the global minimum of the specific power plot is conserved. The breakpoints are chosen for u as $[10\%, 58\%, 100\%]$ and the corresponding breakpoints for p are $[18.9966 \text{ kW}, 92.7341 \text{ kW}, 184.4097 \text{ kW}]$. The PWA fit is done in a least squares fashion. The minimum of the specific power specifies the operation point at which the electrolyzer is the most efficient. By choosing the breakpoints in this way, the MPC will see an incentive to run the electrolyzer at this operation point over any other point. The resulting experimental values and the corresponding piece-wise affine approximations are shown in Figure 2.6.

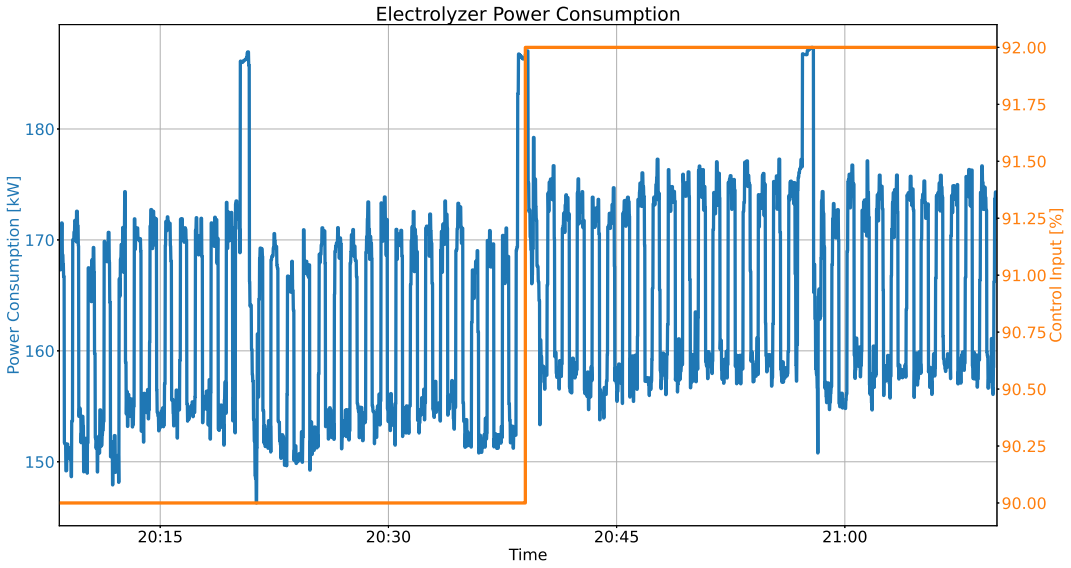

Figure 2.5: The power consumption of the electrolyzer with changing control input to the mass flow controller. The power consumption shows a fast oscillations with a period of around 2 minutes and a slower oscillation with a period of around 20 minutes. This oscillation is not influenced by changes of the control input.

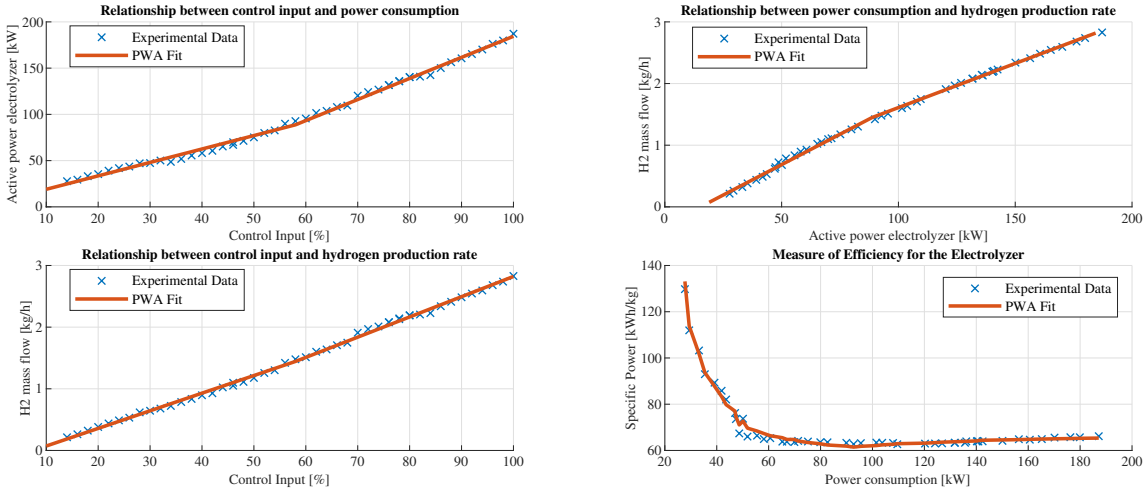

Figure 2.6: Experimental data and piecewise affine approximation of it on the relationships between u and p (top left), u and $\dot{m}_{prod}$ (bottom left), p and $\dot{m}_{prod}$ (top right) and the specific power and p (bottom right). The specific power is the amount of energy which is required to produce one kilogram of hydrogen.

2.4 Compressor modelling

Both compressors, the 390bar and 900bar compressor, are high pressure piston compressors. Unfortunately, there is little to no available information regarding the exact operation of the compressors, except that both compressors have some kind of underlying low-level controller implemented. On top of that a simple high-level control scheme in

the form of a bang-bang controller with hysteresis band is currently used. Within this framework we have the possibility to turn the compressors on or off, but we do not have direct control of their power consumption or any other aspect of the compressors. In the following we will take a look at the 390bar compressor and the 900bar compressor, discuss their currently implemented high-level control scheme and introduce models which can be used in the MPC formulation.

2.4.1 390 bar compressor

In a first step, the existing high-level control scheme of this compressor stage is introduced. This part of the system can be changed and can be replaced by the MPC. Afterwards, the underlying dynamics and implemented low-level control schemes are discussed, which cannot be replaced and have to be included in the MPC model.

High-level controller

Based on the pressure in the 30 bar and 390bar tanks and on whether the compressor is currently running, it will either change states or remain turned on or off. A state transition diagram is shown in Figure 2.7.

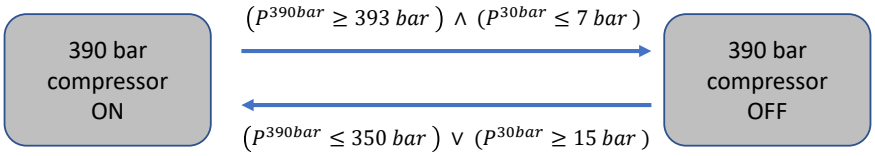

Figure 2.7: The state transition diagram of the 390bar compressor. If none of the transition occurs, the compressor will remain in its current state.

Low-level controller

The low-level controller of the compressor operates it at two different distinct speed levels. It will switch from one compressor speed to another when the pressure in the 30bar tank passes a threshold of 15 bar. This is displayed in Figure 2.8. The desired relationship that we would like to find is between the power consumption of the compressor and the mass flow through it, i.e. how much hydrogen it can compress per time unit. Starting from a physical perspective it is obvious that the compressor will work differently depending on what the input and output pressures are. If they are further apart, the compressor has to put in more work, i.e. more power to generate a mass flow. If the two pressure values are closer to each other, it requires less power to generate a certain mass flow. This means that in order to have an accurate model for the compressors we would need to include the input and output pressures of the compressor in the model. This is problematic in multiple aspects.

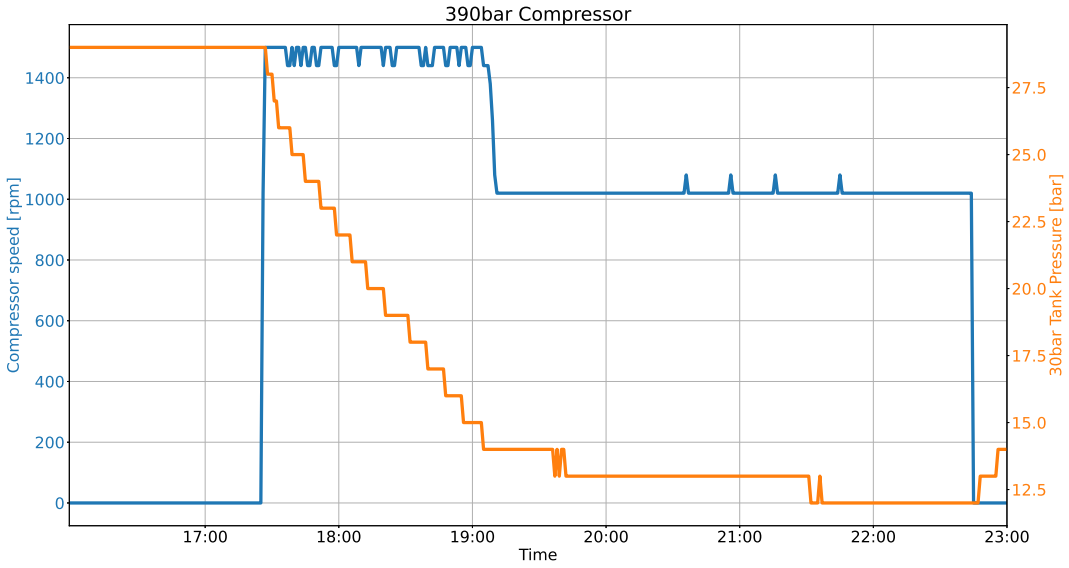

Figure 2.8: The compressor speed for the 390bar compressor with corresponding pressure in the 30bar tank. Once the pressure drops below 15 bar, the compressor speed decreases.

Firstly, as we will see later on, pressure values are not optimal to model the state of a storage tank and we will be using mass units in order to quantify how much hydrogen is inside a tank. Even if the relationship between the pressure gradients and the mass flow is somewhat linear, it will be turned nonlinear by converting the pressure gradient into a mass gradient. As we require a linear model (except integer constraints) for the MPC model this would require piecewise-affine approximations, similar to the ones adopted for the electrolyzer model. This would need to be done for both the input and the output pressure resulting in an increase in optimization variables, which if the approximated functions are not simply linear, also include binary variables. Given the combinatorial nature of mixed-integer programs with respect to the binary variables, this would lead to a non-neglectable increase in solver convergence time.
Secondly, the input of the compressor is not directly connected to the 30bar tanks but rather to smaller, so called, blow down tanks, which act as a buffer between the 30bar tank and the 390bar compressor. The input pressure will be taken from this tank, which is different from the pressure in the 30bar tank. Therefore, we would need to include this tank in the model if we want to capture the influence of the pressure gradient in the compressor model.

All in all, including any kind of pressure dependent dynamics in the compressor model leads to an unreasonable increase in model size and therefore increase in convergence time later on in the optimization problem. Therefore, we will ignore the dependence of the mass flow on the pressure gradient and will simply assume that the compressor will provide a constant mass flow for a given power consumption. The power consumption depends on the compressor speed, which we know depends on the mass inside the 30bar tank. By modelling it in this way we have the possibility to optimize the compressor actuation by controlling the pressure inside the 30bar tank. The model used for the MPC is

$$\dot{m}^{comp,390} = f^c(p^{comp,390}) = \begin{cases} a^1 \cdot p^{comp,390} & m^{30} \geq \bar{m}^{30,15bar} \\ a^2 \cdot p^{comp,390} & m^{30} < \bar{m}^{30,15bar} \end{cases}, \tag{2.2}$$

where $\dot{m}^{comp,390}$ and $p^{comp,390}$ correspond to the mass flow and the power consumption of the compressor average over a timestep, respectively. This is done by running the compressor with a certain duty cycle, such that it is only turned on for a fraction of the timestep size. This is less conservative than fixing the compressor status for a whole timestep and more importantly it requires less binary variables for its implementation. a^1 and a^2 are real values that describe the relationship between mass flow and power consumption for the respective configuration in which the compressor is running. $\bar{m}^{30,15bar}$ denotes the mass inside the 30bar tank when the pressure is at 15 bar. The precise relationship used for this transformation is described in Section 2.5.3. For correct operation we constraint $p^{comp,390}$ to

$$p^{comp,390} \in \begin{cases} [0, p^{comp,390,max,1}] & m^{30} \geq \bar{m}^{30,15bar} \\ [0, p^{comp,390,max,2}] & m^{30} < \bar{m}^{30,15bar} \end{cases}, \tag{2.3}$$

where $p^{comp,390,max,1}$ and $p^{comp,390,max,2}$ are the respective power consumption of the two modi of compressor speeds at which the compressor can run.

2.4.2 900 bar compressor

Similarly to the 390bar compressor, the currently implemented high-level control scheme is investigated and a possible model for the dynamics is proposed which can be used in the MPC. The currently implemented high-level controller works based on a hysteresis band for the pressure inside the 900bar tank, the corresponding state transition diagram can be seen in Figure 2.9.

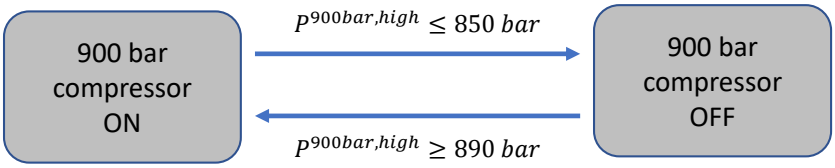

Figure 2.9: The state transition diagram of the 900bar compressor. If none of the transition occurs, the compressor will remain in its current state. $P^{900bar,high}$ denotes the pressure inside the high bank of the 900bar tank.

The 900bar tank consist of a low, middle and high bank. The differentiation between the different banks of the 900bar tanks is important here as, the banks are consecutively used during refuelling, e.g. the high bank is always used first and therefore it reaches the lower bound of 850 bar before the other banks do. Even though we distinguish between the different storage tank banks here, it is possible to model the hysteresis band without considering separate banks as the high bank is the first to cross the lower threshold from above and the last one to cross the upper threshold from below. This is due to the way how the banks are used for refuelling and the way how the compressed mass is distributed over the different banks, which is displayed in Figure 2.10. Therefore, we can assume

that the other two banks are full when $P^{900bar,high}$ crosses the thresholds, allowing us to convert the respective pressure thresholds into mass thresholds that can be represented by the total amount of mass inside the 900bar tanks. This conversion is described in more detail in Section 2.5.3.

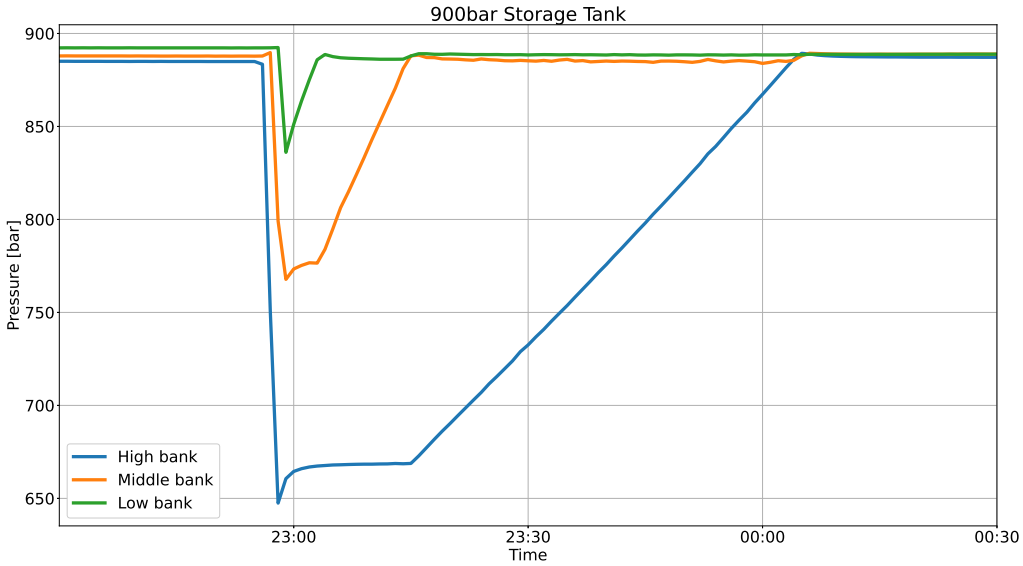

Figure 2.10: The discharging and charging behaviour of the 900bar tank. The 900bar tanks consists of a high, middle and low bank, which are used in this order to refuel vehicles. In a first step, the high bank is used until its pressure is not high enough anymore in order to have a mass flow between tank and vehicle.

Low-level controller

The power consumption of the low-level controller does not show any kind of particular level at which it runs, but rather oscillates quite a bit. Therefore, a distinction between different compressor speeds is not done here. Due to similar reasons as already explained for the 390bar compressor, the pressure gradient's influence on the mass flow is not considered here and a simple relationship between mass flow and power consumption is proposed:

$$\dot{m}^{comp,900} = a^3 \cdot p^{comp,900}, \tag{2.4}$$

$$p^{comp,900} \in [0, p^{comp,900,max}], \tag{2.5}$$

where we use the same PWM scheme as for the 390bar compressor. Therefore, $\dot{m}^{comp,390}$ denotes the mass flow through the compressor and $p^{comp,900}$ the corresponding power consumption.

2.4.3 Compressor Identification

In order to approximate the unknown variables in the compressor models (2.2), (2.3), (2.4) and (2.5) we approximate the mass flow through the compressor $\dot{m}^{comp}$ by taking

the difference in mass estimations, see Section 2.5.3, inside the storage tanks over a given time period[1], while calculating the average power consumption during this time. The average over all data points is taken, resulting in

$$p^{comp,390,max,1} = 20.6350\ kW \tag{2.6}$$
$$\dot{m}^{comp,390,max,1} = 4.2323\ kW \tag{2.7}$$
$$a^1 = \frac{p^{comp,390,max,1}}{\dot{m}^{comp,390,max,1}} = 4.8756\ \frac{kWh}{kg} \tag{2.8}$$

$$p^{comp,390,max,2} = 13.5400\ kW \tag{2.9}$$
$$\dot{m}^{comp,390,max,2} = 2.0720\ kW \tag{2.10}$$
$$a^2 = \frac{p^{comp,390,max,2}}{\dot{m}^{comp,390,max,2}} = 6.5348\ \frac{kWh}{kg} \tag{2.11}$$

$$p^{comp,900,max} = 12.5768\ kW \tag{2.12}$$
$$\dot{m}^{comp,900,max} = 4.9379\ kW \tag{2.13}$$
$$a^3 = \frac{p^{comp,900,max}}{\dot{m}^{comp,900,max}} = 2.5470\ \frac{kWh}{kg} \tag{2.14}$$

By simply taking the average over the whole dataset, we are biasing the estimates towards pressure regions which occurred most often in the dataset. This might not be optimal, however, as there is no obvious better approximation, it is used nonetheless. Note that this kind of formulation for the compressor model is robust against certain model uncertainties which might lead to constraint violation. For example, if the storage mass in the 390bar tank is close to the lower bound, the mass flow through the 390bar compressor will be larger than the modelled one, as the pressure gradient will be quite small, leading to a situation where the mass is further away from the constraints. The same goes for the upper bound, where the compressor will transfer less hydrogen than the model would anticipate and therefore prevent overfilling the storage tanks.

Concluding, the most important factor which can be optimized by including the compressor model is the switching between the different compressor speeds for the 390bar compressor. By keeping the mass in the 30bar tanks high enough, the 390bar compressor will operate at higher compressor speeds which are generally more efficient than lower compressor speeds.

2.5 Storage tank modelling

In the following, different possible models for the storage tanks are constructed and their advantages and disadvantages are listed. Thereafter, the physical limits of the storage tanks are determined and calculated in a way such that they can be used in a receding horizon scheme.

[1]There are no direct mass-flow measurements available.

2.5.1 One-stage model

The simplest possible model for the storage tanks is to group all of them together into one unit and disregard the different storage tank stages and the compressors in between. The corresponding mass dynamics are

$$\begin{aligned} m_{n+1} &= m_n + \Delta m_n^{prod} - \Delta m_n^{dem}, \\ \Delta m_n^{prod} &= f_1(p_n^{el}), \end{aligned} \tag{2.15}$$

where m_n denoted the total mass, Δm_n^{prod} the discretized mass-flow which comes from the electrolyzer and Δm_n^{dem} the hydrogen demand in the timestep n. For this approach to work correctly, new high-level controllers need to be implemented in the compressor stages, making sure that the hydrogen gets correctly distributed in the tanks. The high-level controllers have to make sure that the 30bar tank is never full, unless all other tanks are also full, and the 900bar tank should always have enough hydrogen left, such that it can refuel a car.

The currently implemented controllers on the compressors partially already work in this way. One refuel of a car is usually enough to bring the pressure in the high bank of the 900bar tank below 850bar, therefore turning the compressor on. Therefore, this compressor effectively turns on every time a car comes to refuel. The hysteresis bands of the 390bar compressor would need to be adjusted, as it can easily happen that the 30bar tank is full while the 390bar tank is not completely full but still above 350bar, which is the threshold at which the compressor turns on.

This approach has the major disadvantage that it does not take the power consumption of the compressors into account, which could be optimized by the MPC. Moreover, it does not allow to optimize the compressors input in a way such that the compressors work more efficiently. However, the approach comes with the advantage that it contains fewer continuous and especially binary variables, allowing for a faster convergence of the optimization problem.

2.5.2 Multi-stage model

As just mentioned, it might be desirable to include the compressors in the model, in order to optimize their power consumption in the optimization problem. In the following, two approaches are proposed, one modelling both compressors and one only modelling the 390bar compressor.

Three-stage model

The most detailed model for the storage tanks contains all three pressure stages and both the 390bar and the 900bar compressor as separate variables in the model. This way the optimization problem has information on the power consumption of both compressors and can optimize them. The resulting mass dynamics are described by

$$
\begin{aligned}
m_{n+1}^{30} &= m_n^{30} + \Delta m_n^{prod} - \Delta m_n^{comp,390}, \\
m_{n+1}^{390} &= m_n^{390} + \Delta m_n^{comp,390} - \Delta m_n^{comp,900}, \\
m_{n+1}^{900} &= m_n^{900} + \Delta m_n^{comp,900} - \Delta m_n^{dem}, \\
\Delta m_n^{prod} &= f_1(p_n^{el}), \\
\Delta m_n^{comp,390} &= h_1(p_n^{comp,390}), \\
\Delta m_n^{comp,900} &= h_2(p_n^{comp,900}),
\end{aligned}
\tag{2.16}
$$

where m_n^{30}, m_n^{390}, m_n^{900} denote the storage mass in the 30bar, 390bar and 900bar storage tanks respectively. Between the storage tanks the compressors operate mass flows given by $\Delta m_n^{comp,390}$ and $\Delta m_n^{comp,900}$. While the model is accurate and provides the most room for optimization, it has the major disadvantage that the number of optimization variables drastically increases. In order to accurately model the compressor stages multiple binary variables are necessary such that the resulting optimization problem is not converging in a reasonable amount of time unless the timesteps are significantly increased, the prediction horizon is shortened and the number of scenarios is decreased. All these steps themselves lead to a worse end result, such that the introduction of this additional model complexity brings more problems than it resolves. In the following paragraph a middle way between modelling all compressors and no compressors at all is presented.

Two-stage model

Due to the structure of the storage tanks and the limited available capacity of the 900bar tanks, the 900bar compressor does not have much room for optimization, as it usually has to turn on immediately after a car has been refuelled. Therefore, it is a good trade-off to leave it out of the problem in order to improve convergence time of the solver. The storage tanks can be split up into tanks with low pressure, i.e. 30bar tanks, and high pressure tanks, i.e. 390bar and 900bar tanks. This structure is shown in Figure 2.11 and the mass dynamics are described in Equation (2.17).

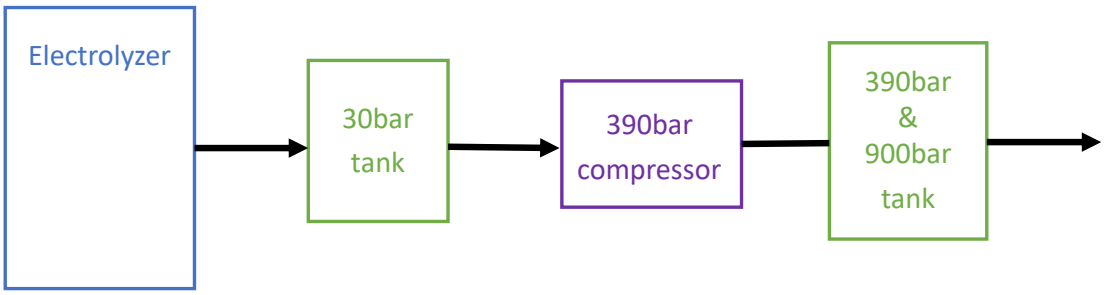

Figure 2.11: A simplified model structure where the 390bar and 900bar tanks are grouped together and the 900bar compressor is excluded. Note that we are neglecting the connection between the 30bar tank and the 900bar tank, which is denoted by the dotted line in Figure 2.1, as it is rarely active.

$$
\begin{aligned}
m_{n+1}^{low} &= m_n^{low} + \Delta m_n^{prod} - \Delta m_n^{comp} \\
m_{n+1}^{high} &= m_n^{high} + \Delta m_n^{comp} - \Delta m_n^{dem} \\
\Delta m_n^{prod} &= f_1(p_n^{el}) \\
\Delta m_n^{comp} &= h_2(p_n^{comp})
\end{aligned}
\tag{2.17}
$$

Concluding, the two-stage model gives a reasonable trade-off between reasonable runtime and accurate modelling compared to the three-stage model. Therefore, the three-stage model is not tested as a model for MPC.

2.5.3 Pressure to mass relationship for compressed hydrogen

All storage tank models described above contain the mass inside the storage tanks as the variable to describe the state of the tanks. This is because it allows for simple dynamics, where the hydrogen produced by the electrolyzer is given by a mass flow and the hydrogen demand is also given by a certain mass that gets refuelled. However, the mass inside the tanks cannot be directly measured, whereas pressure and temperature measurements for the storage tanks are available. Therefore, the goal of this Section is to discuss how a mass estimate can be calculated from pressure and temperature measurements.
The simplest law to give a relationship between pressure, density and temperature of a gas is the ideal gas law. From the density and the fixed volume of the storage tank one can easily calculate the mass contained in it. However, as the name suggests, it is an idealization. It assumes no interaction between the gas molecules, something that is definitely not the case for hydrogen stored at 900bar. The Van der Waals equation for real gases is a bit better as it includes two additional parameters, capturing interactions between gas molecules. However, both of them are not estimating the mass for the highly compressed 900bar tank correctly. Therefore, the relationship of pressure, density and temperature is determined solely based on experimental data, i.e. from data tables [11].
In order to test the accuracy of the relationship based on experimental data, the mass dynamics shown in Equation (2.15) are calculated by estimating the initial value and then it is forward integrating it by using the measured mass flow from the electrolyzer and the measured mass flow at the refuelling station. Next to this, the mass over the same time interval is estimated purely based on the pressure and temperature measurements. If the two quantities match closely with each other, the relationship used for the estimation of the mass is assumed to be usable for the storage mass estimation. The resulting curves are shown in Figure 2.12.

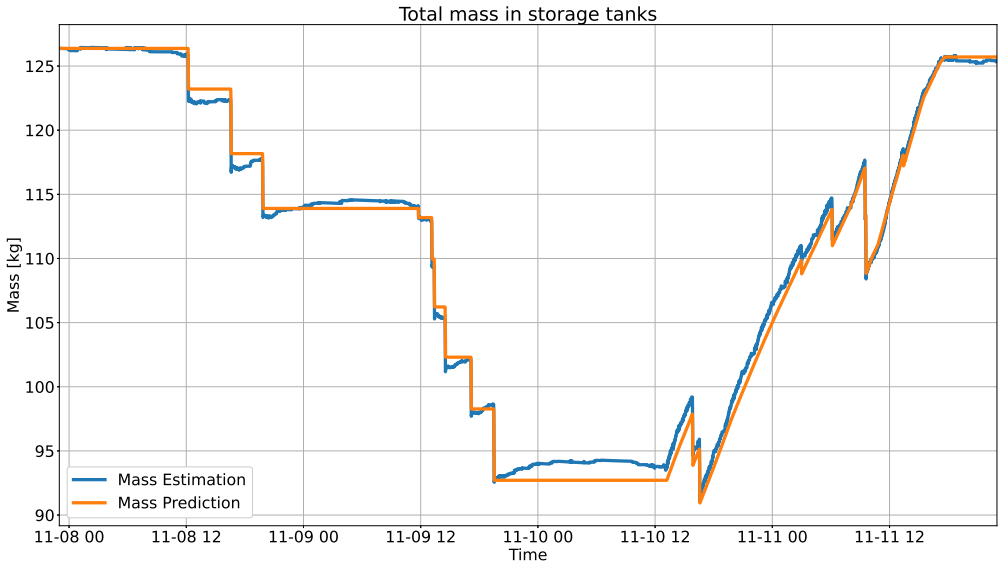

Figure 2.12: The forward integrated total storage mass (orange) and the estimated mass based on pressure and temperature measurements (blue).

2.5.4 Evaluation of electrolyzer and compressor model

In the following the accuracy of the electrolyzer model is tested by applying a control input sequence to the system, considering the one-stage model (2.15), in order to see how the forward integrated states deviate from the mass estimate that we get if we apply the same control input sequence on the real system. The results can be seen in Figure 2.13. For the fact that the model has various instances where it actually fails to predict the hydrogen production accurately as described in Section 2.2.1, the results are definitely within an acceptable range, especially considering the long prediction horizon of two days.

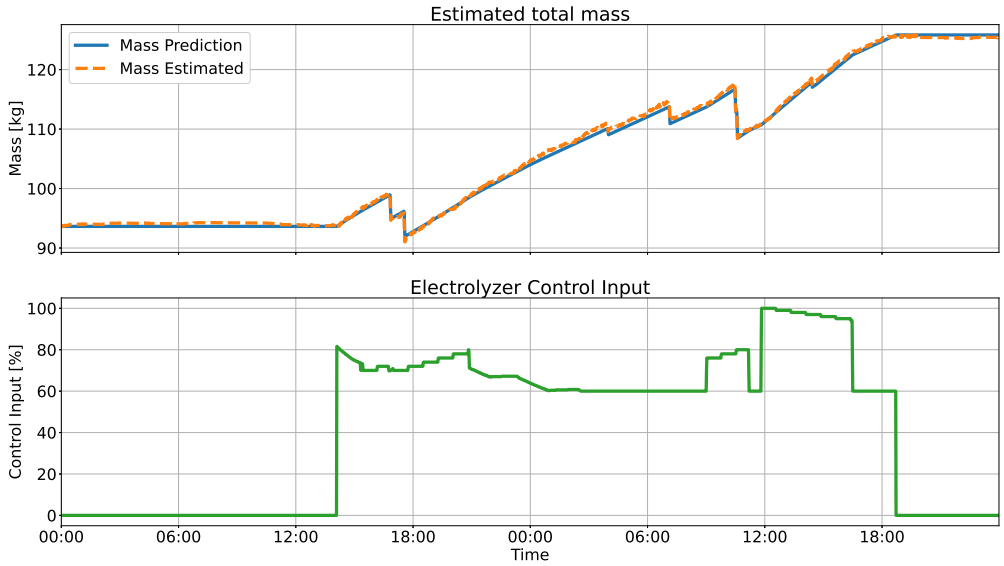

Figure 2.13: The control input and the corresponding mass estimates and mass predictions. The mass estimates are calculated based on pressure and temperature measurements collected from the real system when the shown control input sequence is applied. There are multiple recharging events which lead to sharp drops of the storage mass. The mass prediction is initialized in the first timestep to the same value as the mass estimate and then forward integrated, based on the MPC model and the corresponding control input.

Next, we are going to evaluate the compressor models that we have derived in Section 2.4. For this the three-stage-model (2.16) is used in order to forward integrate the same state as for the evaluation of the electrolyzer. The results are shown in Figure 2.14. Unfortunately, the results are a lot worse compared to the previous case where only the electrolyzer model was considered. However, this result should not be a major surprise, as large and especially fundamental parts of the compressor dynamics were left out in the model. A small offset in the mass flow prediction can already lead to a constantly increasing error in the predicted mass values. As a more complicated compressor model is not possible due to reasons explained in Section 2.4, this mismatch cannot be improved in a simple way.

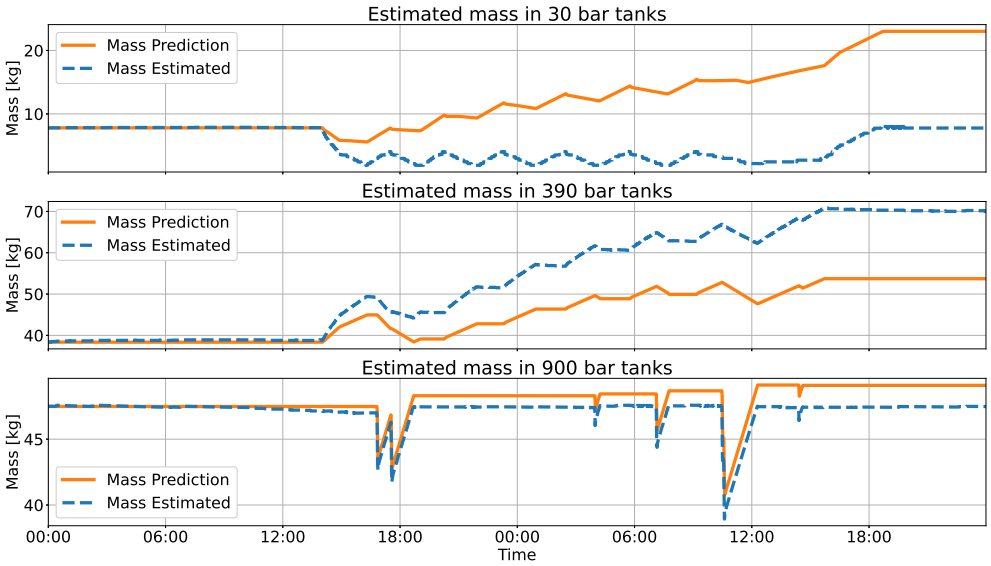

Figure 2.14: The forward-integrated states for the separate storage tanks with the same underlying data as in Figure 2.13. It uses the model (2.16) in order to forward integrate the states with knowledge on the electrolyzer control input, and the status of both compressors.

Concluding, the electrolyzer model delivers a reasonable prediction over a horizon of two days, while the compressor model performs worse. Nonetheless, as the electrolyzer is the main power consumer of the overall system an accurate model is especially important for it. While modelling errors for the compressors are certainly not desirable, in this case they are not preventable and therefore we let the feedback loop handle them.

2.6 Storage tank constraint prediction

The storage tanks limits are not given by mass constraints but rather by pressure constraints. As the chosen model uses mass dynamics in the storage tanks, a pressure-to-mass conversion has to be done. Similarly to the way how the storage mass is estimated from the current temperature and pressure measurements in the tanks, the tank constraints for the prediction horizon are calculated by tank temperature prediction and the fixed pressure limits. Therefore, tank temperature predictions are calculated as described in the following paragraph.

2.6.1 Input/Feature selection

In a first step, the variables that have a physical influence on the storage tank temperature are summarized. As the temperature of the tank depends on the temperature inside the building, we are first quickly going to summarize the variables having an influence on the building temperature. Ambient temperature and solar irradiation are the two most prominent variables for building temperature. Therefore, we are going to consider those two variables for further analysis on their correlation with the storage tank temperature. Additionally to the variables that have an influence on the building temperature, pressure changes in the tank itself should also have some influence on the temperature inside

the tanks. In the following we are going to consider the ambient temperature, the solar irradiation, the pressure and pressure changes as possible variables which are tested on whether they are correlated strongly enough to the temperature inside the storage tank. In order to gain an inside into the correlation of the variables to the storage tank temperature, the Person correlation coefficient (PCC) is calculated which is a measure for linear correlation between two variables. The resulting scores are displayed in the Table 2.1, where the score is always between -1 and 1, where a score close to zero indicated no linear correlation and a score close to 1 or -1 indicate a strong linear correlation between the two variables.

Variable	Pearson correlation coefficient (PCC)
Ambient temperature	0.9743
Solar irradiation	0.4821
Pressure changes	0.0316
Pressure	-0.1141

Table 2.1: The Pearson correlation coefficient for the four variables and their correlation to the storage tank temperature.

The results in Table 2.1 have to be taken with a grain of salt. The Pearson correlation coefficient only gives information about the linear correlation between two variables. If the two variables are connected through a nonlinear function, this kind of relationship is not well captured through the Pearson correlation coefficient. However, the overall trend should still be captured well enough as long as the underlying function connecting the two variables is monotonically increasing or decreasing. Another issue that arises is possible time delays between the variables, as they have an influence on the covariance of the two variables. This can potentially lead to a lower score than the non-delayed variables would have. In the following the scores in Table 2.1 are discussed.
The ambient temperature is strongly linearly correlated with the temperature in the storage tank with a PCC of around 0.96. The solar irradiation seems to have some linear correlation with the storage tank temperature. However, if we calculate the the PCC between ambient temperature and solar irradiation, it is around 0.53, which means that the correlation between solar irradiation and storage tank temperature can result from the correlation between solar irradiation and ambient temperature and the strong correlation between ambient temperature and storage tank temperature. In this scenario it would not make sense to include the solar irradiation in a prediction model, as all information should be contained in the ambient temperature. Therefore, the solar irradiation is not included as a feature of the model, as it seems to not have a significant impact on the desired prediction. This hypothesis is further validated later on by including it in the neural network approach and showing that it leads to an increase in the overfitting of the model. Lastly, pressure and pressure changes do not seem to have any kind of major impact on the storage tank temperature. Therefore, they are not going to be considered as inputs to the prediction model.

Concluding, we have one definite input to the prediction model, the ambient temperature. Solar irradiation, pressure and pressure changes are not included in the model, as their PCC is too low to justify an inclusion. In the following two different models are introduced that can be used to predict the tank temperature.

Auto-regressive approach

Due to the laws of thermodynamics we know that the temperature inside the tanks are not changing arbitrarily fast and it is clear that temperature values at a certain timestep depend on temperature values in the previous timesteps. Therefore, an auto-regressive approach is introduced in order to predict the storage tank temperature values.
The auto-regressive approach takes the last q measurements (including the current one) of the storage tank temperature and the last p prediction values for the ambient temperature and the next ambient temperature into account and estimates from these the next value for the storage tank temperature. This can be thought of as a simple discrete-time system with a transfer function $G(z)$, where the degree of the numerator is q and the degree of the denominator is p, with the storage tank temperature as an output and the ambient temperature as an input. In practice, there is always noise from some kind of source acting on this system. In this approach, the noise is assumed to be i.i.d. with zero mean, acting on the output of the system resulting in the well known ARX model. A generalization of the approach would be to include error dynamics, i.e. to introduce a transfer function $H(z)$, which acts on the noise, before it enters the system. As an example, in the case where we assume that the denominator of $H(z)$ is the same as the one of $G(z)$, the resulting approach is the well known ARMAX model, which stands for auto-regressive moving average with exogenous inputs. In this thesis we restrict ourselves to the simpler ARX approach, as the usage of a ARMAX model for multi-step ahead predictions is non-trivial and is not the main focus of the thesis.

The main structure of the ARX prediction scheme is

$$\begin{aligned} y[n] = -a_1 \cdot y[n-1] - a_2 \cdot y[n-2] - ... - a_q \cdot y[n-q] \\ + b_0 \cdot T[n] + b_1 \cdot T[n-1] + b_2 \cdot T[n-2] + ... + b_p \cdot T[n-p], \end{aligned} \quad (2.18)$$

where $y[n]$ is predicted based on past measurements of itself and past and current ambient temperature forecasts T. The results of this predictions are then fed into the prediction for the value of $y[n+1]$. This way a prediction of arbitrary length can be generated. The values for q and p are determined by performing parameter sweeps and optimizing the mean-squared error calculated by k-fold cross-validation for predictions of length 24, i.e. 1-day ahead predictions. The resulting values for p and q are both 12.

Neural network predictions

The auto-regressive approach has the major disadvantage that in its default form it only captures linear relationships. This can be solved by mapping the regressors of the linear model through a non-linear function, e.g. instead of taking $u[n-1]$ in Equation (2.18) the squared variables $u[n-1]^2$ can be inputted into the model. However, this requires knowledge about what kind of non-linear relationship exists between u and y which is difficult to determine. Therefore, a neural network scheme is implemented which is supposed to learn the non-linear relationship by itself [12]. The prediction is done in a single-shot manner, i.e. a whole day is predicted in one piece. This is a bit less flexible compared to the recursive structure of the ARX approach, however, it is much easier to train and implement than an autoregressive neural network model. If multi-day ahead predictions are required, previously calculated estimates can be fed into the model.

The neural network is implemented in Python with the help of Keras [13] as a multi-layer perceptron and its parameters, i.e. number of hidden layers, number of neurons per layer and the batch size are determined by choosing an initial search area, then performing parameter-sweeps on it and optimizing over the mean square error calculated by k-fold cross-validation with 10 folds. The resulting network consists of two hidden layers, an input and output layer with the following dimensions:

- Input layer: 28 neurons
- Hidden layer 1: 28 neurons, ReLU activation function
- Hidden layer 2: 106 neurons, ReLU activation function
- Output layer: 24 neurons, linear activation function
- Batch Size: 16

The number of epochs is set to 500 epochs, however, this number of epochs was never reached during training as early stopping was used to prevent overfitting of the model. Additionally, as the second hidden layer contains a large number of neurons which means it is susceptible to overfitting it has kernel and bias regularizers and a Dropout layer at the output.

Including the solar irradiation as a feature in the neural network and re-optimizing the whole network architecture leads to a generally better score on the training set compared to the case where we do not include it, but it degrades the performance on the validation set, a clear sign of over-fitting due to non-informative features. This happens even though we are using regularization, dropout layers and early stopping. This further validates the decision to not include the solar irradiation predictions in the models.

Results of the two approaches

In this section, the two approaches are compared against each other and the better method is chosen as the prediction model for the MPC. Table 2.2 shows the scores for a one-day ahead prediction calculated by averaging the results from a k-fold cross-validation. The neural network performs considerably better than the ARX model.

Method	MSE
NN	1.9215
ARX	2.6524

Table 2.2: The mean-square-error (MSE) of the prediction for a one-day ahead prediction calculated. The ARX approach integrates the temperature forward, i.e. for later timesteps in the prediction horizon it uses previously calculated estimates. The neural network predicts a whole day by construction. The errors are calculated through k-fold cross-validation.

In order to use the prediction for the MPC, not only the one-day ahead prediction is important, but also the two- and three-day ahead predictions. Therefore, the two models are also tested on a independent test-set and 1-, 2- and 3-day ahead predictions are calculated. The results are shown in Table 2.3. The neural network not only performs better for the 1-day ahead prediction, but it also manages to reduce the error propagation better than the ARX approach. The most likely reason is that the ARX model cannot

capture any nonlinear relationships between the storage tank temperature and the ambient temperature, which leads to a slight overfitting of the ARX model to the auto-regressive part of the model, introducing larger errors if the auto-regressive part contains errors, which is the case for multi-step ahead predictions.

Prediction Length	MSE - NN	MSE - ARX
1 d.a.p.	1.5806	2.3163
2 d.a.p.	2.2075	3.5285
3 d.a.p.	2.3669	3.9948
Overall	2.0517	3.2799

Table 2.3: The mean-square-error (MSE) on a test set for a three day ahead prediction (d.a.p.) for both the neural network and ARX model.

Concluding, the neural network approach for the storage tank temperature predictions are used. Figure 2.15 shows an exemplary temperature prediction during summer. Based on these temperature predictions, storage mass constraint predictions can be calculated.

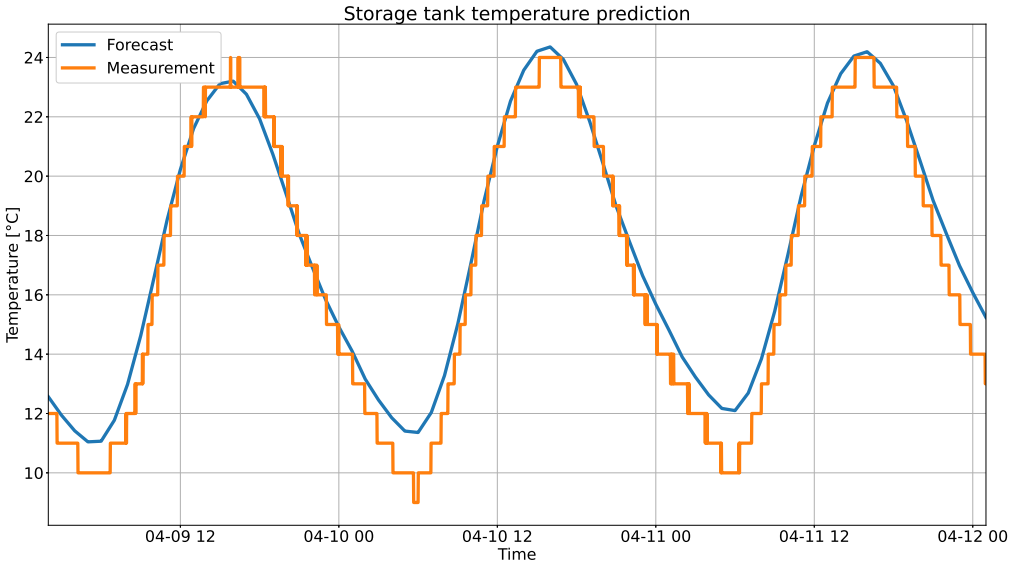

Figure 2.15: The tank temperature prediction (blue) and the tank temperature measurements (orange) during the summer 2020.

Chapter 3

PV Power Prediction

In order to incorporate available PV power in the MPC, predictions of it are necessary. We propose and test two different models, one based on linear regression and one based on a neural network. Both models use solar irradiation and ambient temperature prediction data that is available from MeteoSwiss in order to predict the PV power. Additionally, different models for sunny and cloudy days are constructed. This is done under the premise that for sunny days the model can be pretty accurate while not necessarily robust against prediction errors in the weather forecast, but the cloudy model needs to be most of all robust against such prediction errors, as weather forecasts for cloudy days usually contain a larger prediction error. In order to split up the problem into a sunny and cloudy model, each day needs to be classified as sunny or cloudy. Note that this needs to be automated, as it is part of the control pipeline.

3.1 Weather classification

The weather classification is done in a similar fashion to [14]. The solar irradiation data x is compared to a global horizontal clear sky model x_{ghi}, which predicts the perfect solar irradiation for a certain location and a certain time. The shape of the curve of the solar irradiation prediction is compared to the clear-sky model predictions and if the curves differ by too much the day is classified as not being sunny. The whole procedure is done for x and x_{ghi} of length 24 with 60 minutes between the values, i.e. for hourly values during one whole day. The criteria and the hand-tuned threshold values used to define 'closeness' of the two curves are listed below.

- **Criteria 1:** Absolute difference between mean value

$$|\text{mean}(x_{ghi}) - \text{mean}(x)| < \lambda_1$$

- **Criteria 2:** Absolute difference between maximum values

$$|\max(x_{ghi}) - \max(x)| < \lambda_2$$

- **Criteria 3:** Difference in line length

$$\lambda^3 < L(x_{ghi}) - L(x) < \lambda_4,$$

where

$$L(x) = \sum_i \sqrt{x_i^2 + \Delta t_i^2} = \sum_i \sqrt{x_i^2 + 60^2}$$

- **Criteria 4:** Standard deviation of irradiation slope

$$\sigma(x) < \lambda_5,$$

where

$$\sigma(x) = \frac{1}{\text{mean}(x)} \sqrt{\text{mean}((s(x) - \text{mean}(s(x)))^2)}$$

and

$$s_i(x) = \frac{x_{i+1} - x_i}{\Delta t_i} = \frac{x_{i+1} - x_i}{60}$$

- **Criteria 5:** Maximum value difference in slope of irradiation

$$\max(|s(x_{ghi}) - s(x)|) < \lambda_6$$

If any of the criteria above is violated, the day is classified as not sunny. The thresholds are hand-tuned to

- $\lambda_1 = 55$
- $\lambda_2 = 69$
- $\lambda_3 = -73$
- $\lambda_4 = 100$
- $\lambda_5 = 0.015$
- $\lambda_6 = 1.25$

The algorithm[1] is also able to detect general clear sky periods in time-series data, e.g. it is able to classify sunny and cloudy parts within a day. Here we only use it on full days, i.e. as soon as a day shows slight signs of cloudy behaviour, it is classified as such.

3.2 Data preprocessing

The predictions are made in one hour timesteps, as the weather predictions are available at that frequency. The PV power measurements are available at much higher frequency and have to be averaged or sub-sampled. For sunny days it does not really make large of a difference whether a specific value is taken or not. It might be even beneficial to subsample the PV power data, i.e. simply take the PV power measurements at one particular point in the data without averaging it. This is because averaging over a certain time window always biases the measurements. However, if we are not sure whether a certain sample also represents the surrounding samples well enough, i.e. as it would be the case for a short drop in PV power production due to a cloud moving in front of the sun,

[1] We are use the detect_clearsky algorithm from the pvlib python library

sub-sampling might perform worse than averaging. This becomes especially problematic for cloudy days, where frequent drops in PV power occur. As there is no way of predicting these with the available weather forecasts, this thesis focuses on trying to predict the average power during the 1-hour intervals. Therefore, the PV power measurements are averaged over a 1-hour window.

3.3 Linear regression on PVUSA model

The PVUSA model is a often used model to perform regression on for PV power predictions [15]. It assumes the following nonlinear relationship between solar irradiation I, ambient temperature T and the PV power production p^{PV}:

$$p^{PV} = \theta_1 \cdot I + \theta_2 \cdot I^2 + \theta_3 \cdot I \cdot T \tag{3.1}$$

The usual PVUSA model also includes the wind speed, however, there is no strong correlation for it as described in [16] and we do not have any prediction data for it. θ_1, θ_2 and θ_3 denote the unknown parameters which have to be determined through regression. The parameters that we get by using a least squares approach are $\theta_1 = -5.1195 \times 10^{-2}$, $\theta_2 = -2.2385 \times 10^{-5}$ and $\theta_3 = 6.8615 \times 10^{-4}$ for sunny days and $\theta_1 = -4.9277 \times 10^{-2}$, $\theta_2 = 5.1174 \times 10^{-7}$ and $\theta_3 = 1.1195 \times 10^{-4}$ for cloudy days with a root mean squared error of 3.6643 and 4.6113, respectively.

3.4 Neural network predictions

As already seen before, multilayer perceptron networks have the capability of modelling arbitrary continuous functions [12]. In the following, two networks are introduced, one for sunny days and one for cloudy days and afterwards their performance compared to the PVUSA model is calculated.

3.4.1 Neural network architecture

Both neural network models are implemented in Python with the help of Keras [13] and consist of a total of 3 hidden layers with a variable amount of neurons in them. Similarly to the network constructed for storage tank temperature prediction in Section 2.6, the network parameters here are partially identified through parameter sweeps over a preselected set of parameters by choosing the parameters resulting in a minimal root mean squared error calculated by k-fold cross-validation. All relevant information for the sunny and the cloudy model are summarized in the following paragraph.
The model for sunny days has the following parameters:

- Input layer: 34 neurons
- Hidden layer 1: 34 neurons, ReLU activation function
- Hidden layer 2: 64 neurons, ReLU activation function
- Hidden layer 3: 48 neurons, ReLU activation function
- Output layer: 17 neurons, ReLU activation function

- Batch Size: 32

Additionally, there are dropout layers at the output of the second and third hidden layers with a drop-rate of 0.1 for the second hidden layer and 0.2 for the third hidden layer. Moreover, both of these layers contain kernel and bias regularizers with a value of 0.1 for both regularizers and both layers.

The model for cloudy days has the following parameters:

- Input layer: 34 neurons
- Hidden layer 1: 34 neurons, ReLU activation function
- Hidden layer 2: 96 neurons, ReLU activation function
- Hidden layer 3: 64 neurons, ReLU activation function
- Output layer: 17 neurons, ReLU activation function
- Batch Size: 64

Similarly to the sunny model, there are dropout layers at the output of second and third hidden layers with a drop-rate of 0.1 for the first hidden layer and 0.2 for the second hidden layer and kernel and bias regularizers with a value of 0.1 and 0.0001 for both regularizers and the second and third hidden layer respectively.
Both models train for a maximum of 1250 epochs, while early stopping is used leading to a lower number of actual epochs which is usually around 200-300.

3.5 Results of linear regression and neural network approach

In order to compare the two approaches, the root mean squared error and the mean absolute error are calculated by performing k-fold cross-validation with 10 folds. The resulting errors are then averaged over all folds and are shown in Table 3.1.

Method	RMSE	MAE
NN - sunny	2.7800	1.6565
PVUSA - sunny	3.6601	2.5597
NN - cloudy	4.4488	2.5750
PVUSA - cloudy	4.9885	3.0591

Table 3.1: The root-mean-square-error (RMSE) and the mean-absolute-error (MAE) for the one-day ahead prediction calculated by k-fold cross-validation.

It is directly obvious that the neural network approach is performing better than the linear regression based model. Therefore, the neural network models are used for the PV power predictions. In the following a simple scheme is introduced to further improve the performance of the neural network approach.

3.6 Rolling window approach

In order to adapt to seasonal differences and varying external factors which we do not measure, a rolling time window approach is implemented. We pre-train the neural network offline on the whole dataset. This model is then retrained each time we make a

new prediction based on the data contained in the time window. This means we are only incorporating recent data into the training which results in a small temporal distance between the prediction- and the training data. In order to exclude outliers from entering the training phase of the time window, a list of previous days is constructed that contain useful and good-quality training data. This list is updated at the end of each day by evaluating whether the day was an outlier or not and including the day in the list if it was no outlier. This is done by comparing the predicted PV power against the measured PV power averaged over the whole day. If the averages differ by too much, the day is labelled as an outlier.

In order to validate the performance of the rolling time window approach, the data is split up into a training set and a test set. First, the simple neural network described above is trained on the training set. Thereafter, the rolling time window approach is tested on the test set with the pre-trained model as a base model. As the training and test set are completely independent of each other, we have no influence on the score during the evaluation of the rolling window approach.

3.6.1 Results of rolling window approach

Even though, the approach has a multitude of hyper-parameters, we are going to concentrate on only two of them: the learning rate and the window size. The rest is fixed to either the default values given by TensorFlow and/or Keras or are fixed to some custom selected value. Note that the selection of learning rate here is especially important as the training set is incredibly small and the main idea of the rolling window approach is to only slightly change the weights of the network, such that the prediction is biased towards data from the time window, without actually overfitting to it.
The learning rate and the window size are optimized in order to minimize the root mean square error on the test set. This is done in an iterating fashion, i.e. the optimization variable is switched between learning rate and window size. This iterative approach is stopped once the optimal values for both variables have not changed during one whole iteration. The function between the hyper-parameters and the error are non-convex, such that we get multiple local minima. The minima for the learning rate was clearly the best for 0.0001 (independent for the window size), where the default value (also used for the offline training) is equal to 0.001. The lowest of these local minima with respect to the window size are shown in the Tables 3.2 and 3.3 for sunny and cloudy days respectively:

Method	RMSE
NN - No retraining	2.7865
NN - WS: 15	2.4694
NN - WS: 17	2.4510
NN - WS: 25	2.4496

Table 3.2: The root-mean-square-error (RMSE) for the one-day ahead prediction of sunny days with the rolling time window scheme for different window sizes (WS). The first entry corresponds to the case where no retraining is done before the fitting, which would correspond to a window size of zero. The optimal learning rate is equal to 0.0001 for all of the window sizes above.

Table 3.2 shows a clear improvement of the PV power prediction for sunny days. For cloudy days, however, a large improvement fails to appear. The data in Table 3.3 still shows a slight improvement such that it is still useful to use the new approach. The dif-

ference between the sunny and cloudy day models is most likely due the fact that in the sunny model we can assume little to no prediction error in the weather forecasts. That way the model can fit the data accurately to the most recent data in the time window. For cloudy days on the other hand, the error in the weather forecast can be considerably higher such that the network has difficulties to accurately fit the data and it remains in a state where it is robust against most forecasting errors, but fails to provide an extremely accurate prediction. However, this issue cannot be solved by further tuning the network, but has to be solved by improving the weather forecasts in some way. In the end we choose a window size of 25 for the sunny model and 28 for the cloudy model[2].

Method	RMSE
NN - No retraining	4.1788
NN - WS: 15	4.1090
NN - WS: 28	4.1131

Table 3.3: The root-mean-square-error (RMSE) for the one-day ahead prediction of cloudy days with the rolling time window scheme for different window sizes (WS). The optimal learning rate is equal to 0.0001 for all of the window sizes above.

Concluding, the PV power of a particular day is estimated by classifying that day as being either sunny or cloudy, retraining a previously trained model on recent data and then predicting the PV power based on this retrained model. In this way the model is able to adapt to changing parameters automatically. This approach is made robust against outliers by keeping track of recent high-quality data which can be used for the retraining phase. Figure 3.1 shows some predictions during April 2020.

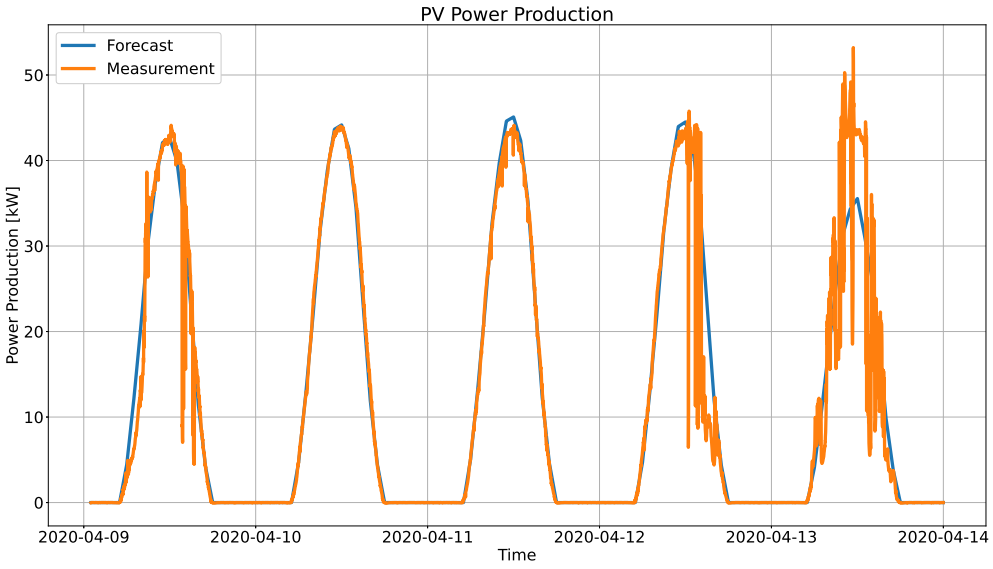

Figure 3.1: The actual PV power (orange) and the predicted PV power (blue) during April 2020.

[2]A WS of 15 results in a lower value in this case according to Table 3.3, however the difference compared to a WS of 28 is less than 0.005 and a larger window size is generally less problematic from the aspect of overfitting. Therefore, the larger window size value is chosen here.

Chapter 4

Hydrogen Demand

Car refuelling cannot be precisely predicted and is most accurately modelled by a stochastic variable. In this sense, the major goal of this section is to determine statistical information about the hydrogen demand and to construct a statistical model in order to generate artificial samples which can be used in the MPC.

The modelling of the hydrogen demand is split up into three sub-problems. Firstly, how many cars arrive within a day, secondly, what is the arrival time distribution within the day and lastly, how much hydrogen gets refuelled.
In a first step we split the data into two parts: weekdays and weekends. This is done under the consideration that noticeably fewer cars refuel during weekends. Therefore, the assumptions listed below only have to hold separately within weekdays and weekends.

Assumptions:

- The cars behave independently from each other, i.e. the arrival of one car does not have an influence on the arrival of other cars.
- The probability that a car arrives within a day is constant over all days (apart from the distinction of weekdays and weekends).

With these two assumptions it is possible to model the number of cars per day through a Poisson distribution. The Poisson distribution is given by the following probability mass distribution:

$$Pr(X = k) = \frac{\lambda^k \cdot e^{-\lambda}}{k!}. \tag{4.1}$$

It contains one free parameter which is fitted through a maximum likelihood estimate:

$$\hat{\lambda}_{MLE} = \frac{1}{n} \cdot \sum_{l=1}^{n} k_l, \tag{4.2}$$

where k_l describes all observed number of cars per day, i.e.

$$k_l \in \{0, 1, 2, ..., n-1\}.$$

In essence this simply means that the mean value of the samples and of the fitted distribution is equal and therefore we have an unbiased estimator. Before performing the actual fit, the data has to be preprocessed.

4.1 Data Preprocessing

For the Poisson distribution the average arrival rate for each day has to be equal. In the available data this is not inherently given over the whole dataset as the number of customers has increased in recent years. This can be seen in Figure 4.1. There are different possibilities to cope with the problem of increasing number of customers. We can take the data and actively remove customers from it until we have a constant number of customers at each timestep over the whole dataset. Another possibility is to make use of the somewhat constant number of customers at the beginning of the data-set and only perform the fit on this part. The first approach is flawed if the data do not have enough regular customers as the amount of data we need to remove is too large. Furthermore, the less customers we have the more the assumptions of the Poisson distribution are violated, as unique decisions of customers have a greater impact in practice and do not get averaged out by a large number of other customers. As the estimated number of active customers is already quite small and therefore the number of regular customers is even smaller, the second approach for preprocessing the data is chosen.

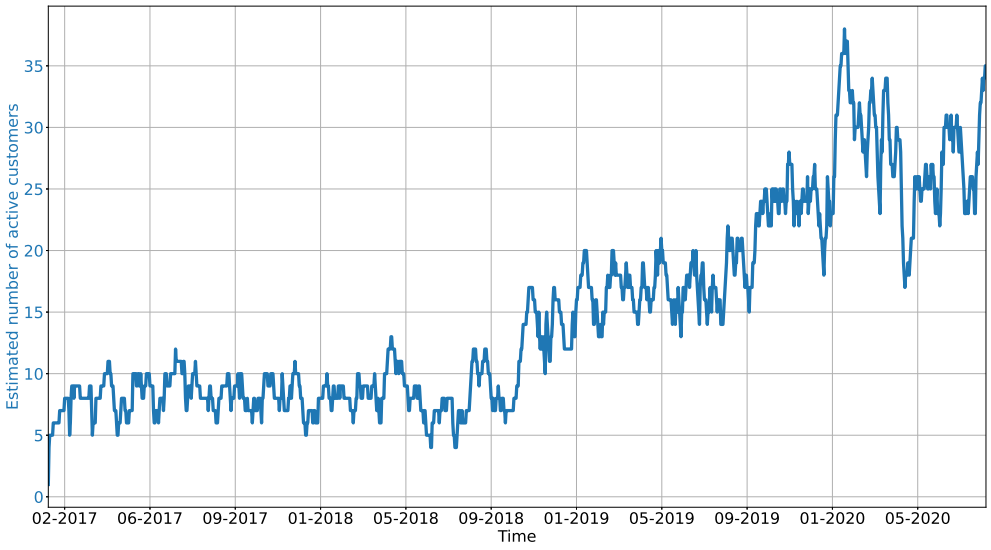

Figure 4.1: The estimated number of active customers which is calculated for each day by taking the sum of all customers that were recharging within the last 4 week.

4.2 Poisson distribution fitting

The interval which is used for the fit starts at the start of April 2017 and goes until the end of September 2018. The fitted parameters are $\lambda_{weekday} = 1.29745$ and $\lambda_{weekend} = 0.3544$. The resulting Poisson distributions can be seen in Figure 4.2 and 4.3 for weekdays and weekends respectively.

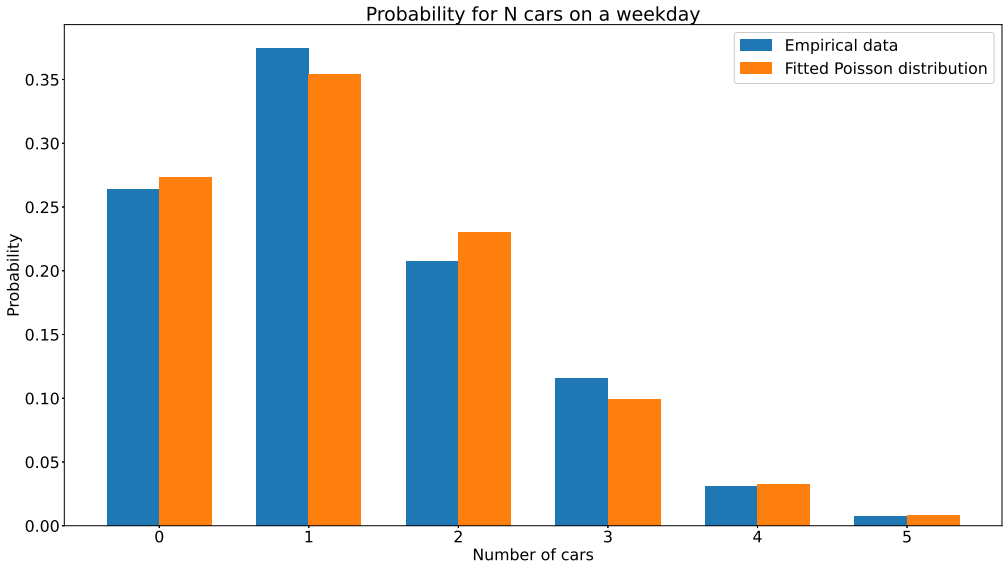

Figure 4.2: Comparison of the fitted Poisson distribution (orange) to the collected data for the weekday model.

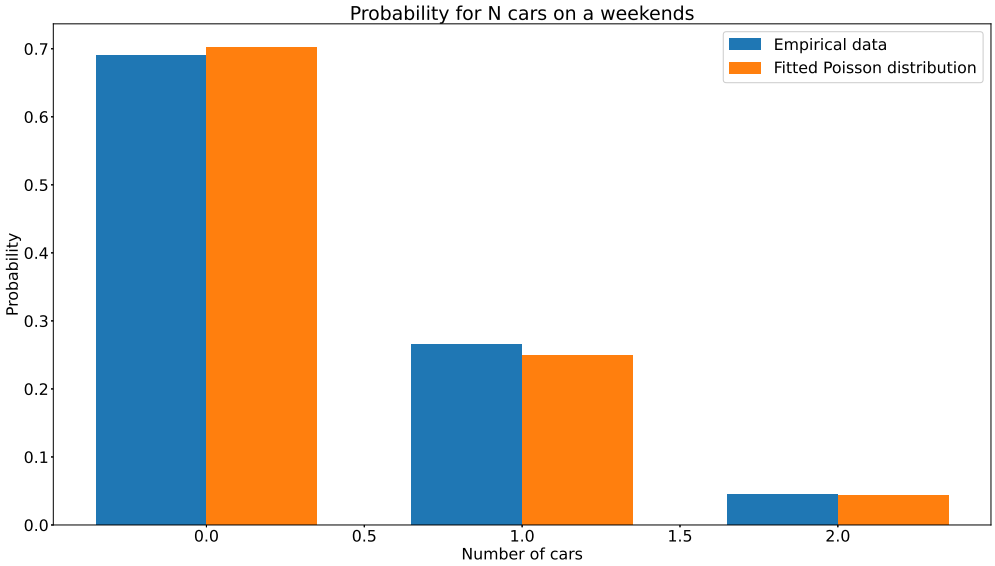

Figure 4.3: Comparison of the fitted Poisson distribution (orange) to the collected data for the weekend model.

4.3 Evaluation

In a first step a Pearson's chi-squared test is performed in order to validate the fit of the Poisson distribution to the samples [17]. Under the null hypothesis that the sample come from the fitted Poisson distribution we get a p-value of 0.7536. Therefore, for normal

significant levels, e.g. 5%, there is no reason to reject the null hypothesis. In the Table 4.2 the results of a Pearson's chi-squared test performed on the samples of different weekdays, i.e. Monday, Tuesday, etc., is shown. Similar to the general case, there is no statistical reason to reject the null hypothesis, e.g. the number of cars on a Monday are drawn from the fitted Poisson distribution, under a reasonable significance level. The same holds true for weekends, see Table 2.1.

Weekday	p-value
Overall	0.7536
Monday	0.2487
Tuesday	0.7717
Wednesday	0.7867
Thursday	0.5367
Friday	0.7037

Table 4.1: The p-value obtained through a Pearson's chi-squared test under the null hypothesis that the samples are drawn from the fitted Poisson distribution for weekdays. The scores for Monday, Wednesday and Friday have to be taken with caution as they include in more than 20% of their bins less than 5 bin counts which is often a minimal criteria for the Pearson's chi-squared test. This problem cannot be avoided with the current amount of data available.

Weekend	p-value
Overall	0.7906
Saturday	0.8484
Sunday	0.8178

Table 4.2: The p-value obtained through a Pearson's chi-squared test under the null hypothesis that the samples are drawn from the fitted Poisson distribution for weekends and holidays.

In order to use the Poisson distribution later on to generate samples for the stochastic MPC, the mean arrival rate has to be changed according to the number of active customers. This can be done by multiplying λ with the ratio $\frac{n_{test}}{n_{train}}$, where n_{train} is the number of customers in the data set that we use to fit the Poisson distribution and n_{test} is the number of customer at the time when the sample should be generated.

4.4 Arrival rate within a day

In order to get useful hydrogen demand data for the stochastic MPC the number of cars alone is not enough as we also need to estimate their arrival time. The most straight forward approach is to take the empirical arrival distribution in order to approximate the probability distribution of the arrival time. Due to the finite amount of samples the empirical probability distribution has to be split up into bins. Due to the low number of samples that are available, the bins are chosen rather large with a size of half an hour for the weekends and holidays and a quarter hour for the weekdays. This is done under the consideration that if a bin does not contain a sufficient amount of samples it does not represent the underlying probability distribution well enough. Furthermore, bins that still only contain a small number of samples are assigned a probability of zero, under the argumentation that they represent outliers that are not worth to be considered. In order to use this discrete probability distribution for the MPC, assumptions about the arrival

times within each bin have to be done, as the MPC has variable sized timesteps. As there is no indication that cars always arrive within a certain part of one hour more often than in others, an uniform distribution is assumed within each bin. This way we can generate samples with minute precision and allocate them to the correct timesteps in the MPC.

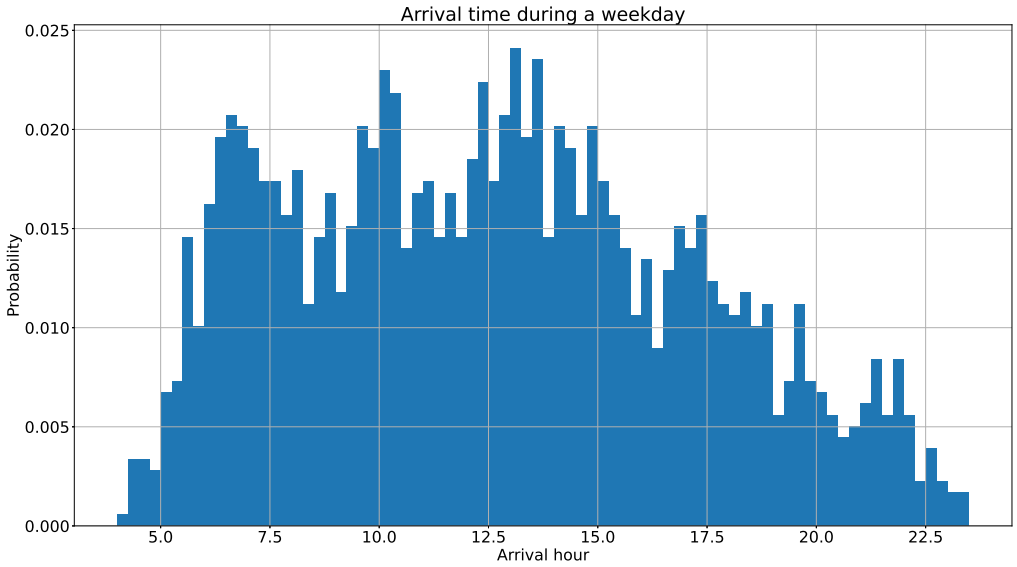

Figure 4.4: The empirical distribution for the arrival time of cars during a weekday. The bin size is chosen as 15 minutes. Within each bin, an uniform distribution is assumed.

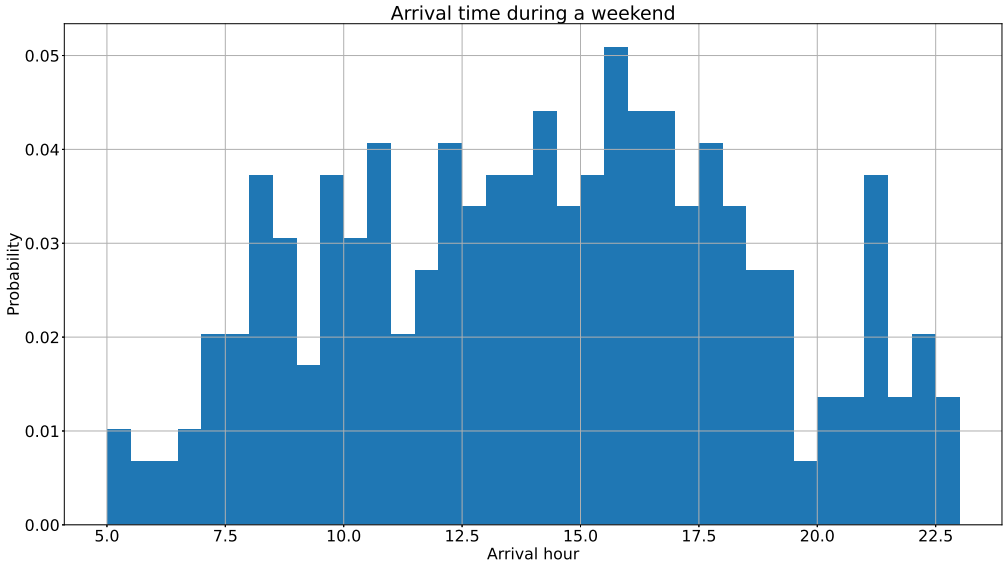

Figure 4.5: The empirical distribution for the arrival time of cars during a weekend day. The bin size is chosen as 30 minutes due to a lack of data for weekends. Within each bin, an uniform distribution is assumed.

4.5 Amount of hydrogen refuel

Similarly to the arrival rate distribution introduced above, the amount of hydrogen is also determined from the empirical probability distribution. The empirical distributions for weekdays and weekends can be seen in Figures 4.6 and 4.7. Here we have made the assumption that the amount of hydrogen that gets refuelled is independent on the arrival time of the car. It is thinkable that there is a dependence between the arrival time and the amount that gets refuelled, however due to a lack of data this kind of a dependence would be difficult to capture. If more data is available in the future, this point of the hydrogen demand model can be reconsidered.

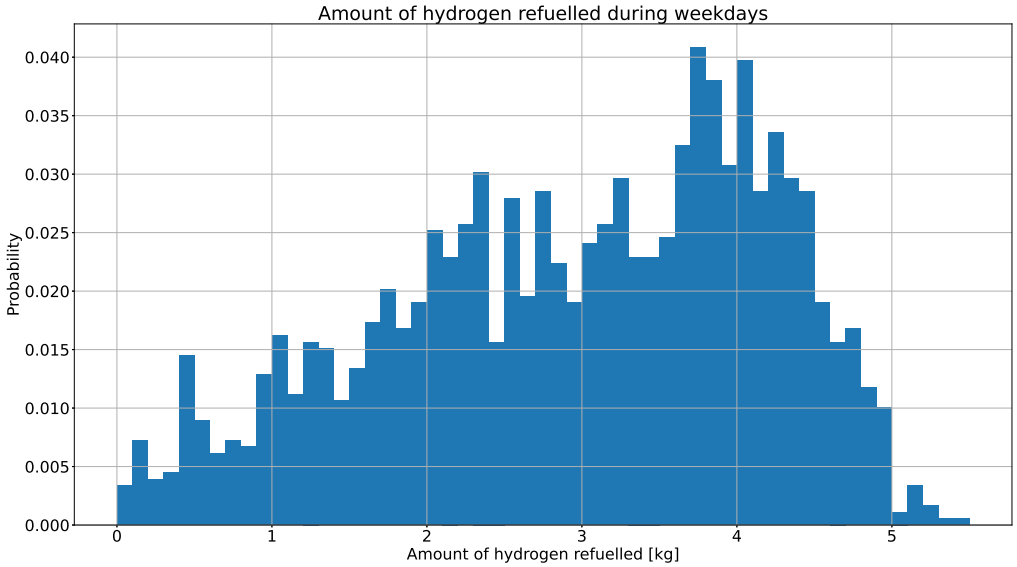

Figure 4.6: The empirical distribution for the amount of hydrogen a car refuels if it arrives during a weekday. The bin size is chosen as 0.1 kg. Within each bin, an uniform distribution is assumed. Note that there are sometimes cars coming that do only charge very little hydrogen. Therefore, there are even bins close to 0kg.

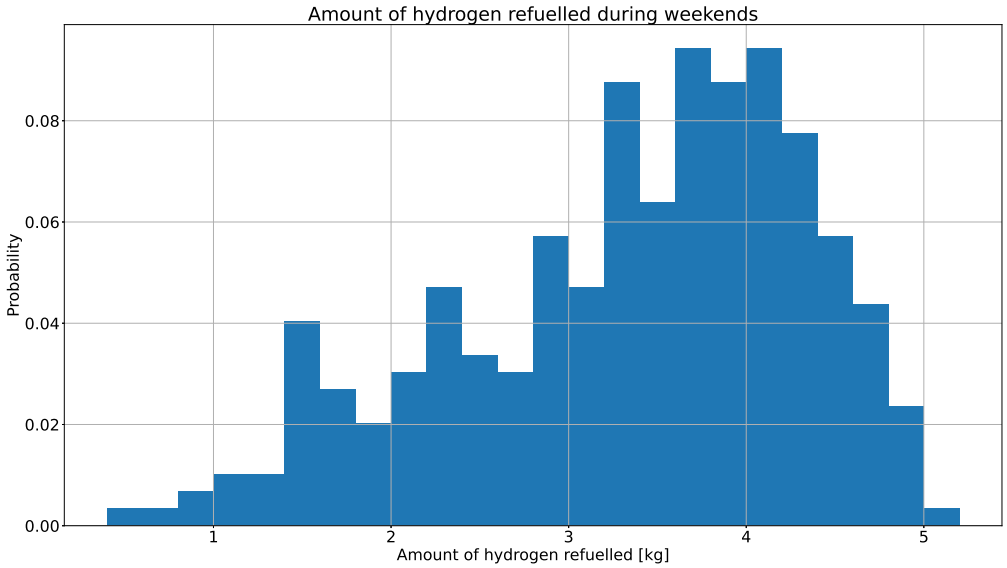

Figure 4.7: The empirical distribution for the amount of hydrogen a car refuels if it arrives during a weekend day. The bin size is chosen as 0.2 kg due to a lack of data for weekends. Within each bin, an uniform distribution is assumed.

4.6 Sample Generation

In order to generate samples for the MPC, we first determine whether the day for which the samples should be generated for is a weekday or not. Then, the number of active customers is estimated and the mean arrival rate in the Poisson distribution is scaled accordingly. Afterwards, a sample from the Poisson distribution is taken, giving the number of cars for the particular day. For each of the cars arriving within the day, a sample from the arrival time distribution and the refuel amount distribution is taken and allocated to the corresponding car. This results in a set of events which contains all relevant information for the MPC. Each event within the set is then allocated to the right timestep within the prediction horizon of the optimization problem. This procedure can be repeated multiple times in order to generate samples for multiple days.

Chapter 5

Controller Design

In this chapter the previously developed models and predictions are combined into an optimization framework which is repeatedly solved by taking the latest state measurements into account. In a first step a simple deterministic problem, based on the one-storage-stage model, is constructed. Thereafter, the problem is extended to include the 390bar compressor according to the two-storage-stage model and the two models are compared to the existing bang-bang controller. Following this, the deterministic problem structure is adjusted to a stochastic setting to capture the stochasticity of the hydrogen demand. As the stochastic approach scales poorly with increasing problem size, it is relaxed and different relaxing heuristics are compared to the original stochastic problem in a downsized problem formulation. The resulting relaxed stochastic problem performance is compared to the deterministic controllers. Afterwards, the scalability of the MPC with respect to prediction horizon length and the available PV power is tested. Lastly, the MPC behaviours for realistic varying electricity prices is tested which shows the flexibility of the derived MPC formulation.

5.1 Simple MPC controller

The simplest possible problem formulation for the system is introduced in this section. It uses the one-stage storage model and therefore does not include the compressors and their power consumption.

5.1.1 Problem structure

Before stating the problem, some notation is introduced. Δt_i denotes the timestep size of the i-th time step in the prediction horizon, which is used in combination with p_i^{el} and $\dot{m}_i^{prod}$, the power consumption and mass flow of the electrolyzer, respectively. The energy consumption and the mass production during one timestep can be simply calculated by multiplying p_i^{el} and $\dot{m}_i^{prod}$ with Δt_i. This allows for a problem formulation with variable

timestep size in the prediction horizon.

$$
\begin{aligned}
\min \quad & \sum_{i=0}^{N-1} (c_i^{grid} \cdot p_i^{grid} \cdot \Delta t_i + c_i^{PV} \cdot (p_i^{el} - p_i^{grid}) \cdot \Delta t_i + c^{switch} \cdot |\delta_i^{el} - \delta_{i-1}^{el}|) \\
& \quad + c^{term} \cdot m_N && (5.1) \\
\text{s.t.} \quad & m_{i+1} = m_i + \Delta m_i^{prod} - \Delta m_i^{dem} && (5.2) \\
& m_i^{min} \leq m_i \leq m_i^{max} && (5.3) \\
& m^{end} \leq m_N && (5.4) \\
& \Delta m_i^{prod} = f^1(p_i^{el}) \cdot \Delta t_i && (5.5) \\
& p^{el,min} \cdot \delta_i^{el} \leq p_i^{el} \leq p^{el,max} \cdot \delta_i^{el} && (5.6) \\
& \dot{m}^{prod,min} \cdot \Delta t_i \cdot \delta_i^{el} \leq \Delta m_i^{prod} \leq \dot{m}^{prod,max} \cdot \Delta t_i \cdot \delta_i^{el} && (5.7) \\
& p_i^{el} \leq p_i^{grid} + p_i^{PV} && (5.8) \\
& \delta_i^{el} \in \{0, 1\} && (5.9) \\
& m_0, \ \delta_{-1}^{el} \text{ given.}
\end{aligned}
$$

Once this problem is solved, we take p_0^{el}, the power consumption of the electrolyzer, and δ_0^{el}, the electrolyzer status, and apply those inputs to the system. p_0^{el} is first mapped through the inverse function of f^2, see Section 2.3.1, in order to get the control input which can be directly applied to the system.
In the following we quickly go through all important parts of the optimization problem, describe their use and how they are effectively implemented.

- Objective function (5.1): In each timestep we have a cost term for power that we take from the grid and for PV power that we use. The reason why the used PV power is also weighted is in order to prevent the controller from using PV power in an inefficient manner. This can be thought of as a term that represents the cost at which we could sell the PV power instead of using it ourselves.
  Additionally, we weight the switching of the electrolyzer state δ^{el} in order to give the controller an incentive for not switching the electrolyzer status too often, as it contains transient dynamics that we have not modelled. The 1-norm used for this can be easily implemented by using epigraph variables.
  Last but not least, in order to prevent the optimization problem from converging towards the lower bound of the mass in the storage tanks, we give the controller an incentive of having hydrogen in the tank at the end of the horizon. Section 5.2.1 discusses the precise choice of this weight in more detail.

- Storage mass dynamics (5.2) and limits (5.3), (5.4): The dynamics of the one-stage storage tank model with the corresponding constraints. The limits for the mass are assumed to be deterministic and come from the storage tank constraint prediction that is performed in Section 2.6. The hydrogen demand Δm_i^{dem} in this case is also assumed to be known, and is drawn from the probability distribution derived in Chapter 4. In order to improve the performance of the MPC for lower storage mass values a terminal constraint is introduced which requires the storage mass at the end of the prediction horizon to be slightly higher than the usual lower storage mass constraint.

- Relationship between power consumption and hydrogen production of the electrolyzer (5.5): Here the piecewise affine approximation derived in Section 2.3.1 is used. As the approximation is a concave function, which can be modelled through a min-function and the function value is indirectly minimized (it is upper bounded by the variable that is actively minimized), the equality constraint can be modelled by using inequality constraints. This approach is valid as long as these inequality constraints are tight.
  Lets consider the case where the bounds are not tight. This corresponds to a situation where the optimization problem actively decides that the electrolyzer should run with a higher power consumption while producing less hydrogen than it could with this level of power consumption. There are two situations where this can occur. Firstly, if the desired hydrogen production is below the lower bound of what is physically possible with the given discretization step-size. By actively constraining the hydrogen production as it is done in Equation (5.7), this scenario will never occur. Secondly, if the available PV power is available in abundance and we do not weight the use of it, the optimization problem might converge to a solution where the inequality constraints are not tight. In this case, however, we also know that the same optimal solution is also attained at the constraints, as the underlying problem with fixed binary variables is a linear program. In this case we can simply map the input to the border of the feasible space, which will not change the optimal objective value. Note that this is a highly artificially created situation and is not happening with the available PV power of the system.

- Power constraints (5.8): The power used for powering the electrolyzer p_i^{el} has to be taken from the grid and/or the available PV power. An inequality constraint is used in order to allow the possibility of not using all available PV power, for example in order to prevent inefficient use of it by running the electrolyzer at low efficiency levels, i.e. at low power consumption levels.

Note that for some of the constraints it is assumed that the power from the grid is minimized, which means that we require $c_i^{grid} > 0$. This makes sense in the context that we want to use as little power from the grid as possible.

Before evaluating the performance of this MPC formulation, we are going to introduce the two-storage-stage MPC formulation such that a direct comparison between the two can be made.

5.2 Complex Problem Structure

The previous problem structure has the major disadvantage that it does not take the power consumption of the compressor stages into account. As already discussed in Section 2.5.2, from a practical point of view, the two-stage storage tank model has the best trade-offs between accurate modelling and reasonable runtime, by only modelling the compressor stages that have actually the flexibility to be optimized. In the following the adjusted optimization problem structure is presented.

$$\min \quad \sum_{i=0}^{N-1} (c_i^{grid} \cdot p_i^{grid} \cdot \Delta t_i + c_i^{PV} \cdot (p_i^{el} + p_i^{comp} - p_i^{grid}) \cdot \Delta t_i + c^{switch} \cdot |\delta_i^{el} - \delta_{i-1}^{el}|)$$
$$+ c^{term} \cdot (m_N^{low} + m_N^{high}) + c^{term,high} \cdot m_N^{high} \tag{5.10}$$

$$\text{s.t.} \quad m_{i+1}^{low} = m_i^{low} + \Delta m_i^{prod} - \Delta m_i^{comp} \tag{5.11}$$
$$m_{i+1}^{high} = m_i^{high} + \Delta m_i^{comp} - \Delta m_i^{dem} \tag{5.12}$$
$$m_i^{low,min} \leq m_i^{low} \leq m_i^{low,max} \tag{5.13}$$
$$m_i^{high,min} \leq m_i^{high} \leq m_i^{high,max} \tag{5.14}$$
$$m^{low,end} \leq m_N^{low}, \; m^{high,end} \leq m_N^{high} \tag{5.15}$$
$$\Delta m_i^{prod} = f^1(p_i^{el}) \cdot \Delta t_i \tag{5.16}$$
$$\Delta m_i^{comp} = f^c(p_i^{comp}) \cdot \Delta t_i \tag{5.17}$$
$$p^{el,min} \cdot \delta_i^{el} \leq p_i^{el} \leq p^{el,max} \cdot \delta_i^{el} \tag{5.18}$$
$$0 \leq p_i^{comp} \leq p_i^{comp,max} \tag{5.19}$$
$$p_i^{comp,max} = h^c(m_i^{low}, \bar{m}^{30,15bar}) \tag{5.20}$$
$$p_i^{el} + p_i^{comp} \leq p_i^{grid} + p_i^{PV} \tag{5.21}$$
$$\delta_i^{el} \in \{0, 1\} \tag{5.22}$$
$$m_0^{low}, \; m_0^{high}, \; \delta_{-1}^{el} \text{ given.}$$

After finding the optimal solution, p_0^{el}, δ_0^{el} and p_0^{comp} are applied to the system. The electrolyzer inputs p_0^{el} and δ_0^{el} are applied in the same fashion as in the simpler model described above. From the compressor power consumption the ratio of it to the maximum possible compressor power consumption is calculated and the compressor is run for that fraction of the time during the timestep. This follows the procedure described in Section 2.4. In the following, the aspects of the problem formulation which differ from the simpler problem described above are summarized:

- Objective function (5.10): Different from (5.1), we include here different terminal state weights. The weight on the total mass that remains inside the tanks at the end of the prediction horizon is the same as in the simpler case. Additionally, a weight term for mass in the higher pressure tanks is included in order to incentives the controller to move hydrogen into it. The details of how the weight term $c^{term,high}$ is calculated is described in Section 5.2.1.

- Storage mass dynamics (5.11), (5.12) and limits (5.13), (5.14), (5.15): In contrast to the simpler model, the storage tanks are split up into two stages as described in Section 2.5.2. They are coupled by the discretized mass flow through the compressor given by Δm_i^{comp}

- Relationship between mass flow through the compressor and its power consumption (5.17): As described in Section 2.4, the compressor can operate in two different operation points where it is assumed that it compresses a constant amount of hydrogen per time unit. The upper limit of p_i^{comp} is therefore different depending on the mass in the 30bar tank, which is modelled in (5.19) and (5.20). This requires the introduction of an additional binary variables which is 0 or 1 depending on whether m_i^{low} is

below or above the threshold of 15 bar. The precise implementation of (5.19), (5.20) and (5.17) and this additional binary variable is quite tedious as multiple additional continuous variables have to be introduced in order to model it in a linear fashion. For interested reader, the implementation can be found in the Appendix A.

Before performing the evaluations of the two model predictive controllers and the currently implemented bang-bang controller, the terminal state weight is discussed in more detail.

5.2.1 Terminal state cost

The goal of the terminal state cost is to give an incentive to having mass inside the storage tanks at the end of the prediction horizon. This way, the optimization problem is not just producing the bare minimum it has to in order to remain inside the constraints. The choice of this weighting term has a large influence on the behaviour of the controller. Choosing it too high leads to the electrolyzer running all the time as long as the storage tanks are not full. Choosing it too small does not solve the problem at hands. The goal is to have a weight which keeps the storage tanks approximately at a constant level. The general idea is to approximate the amount of energy necessary to charge the storage tanks to that level and the cost associated to doing that. Assuming that the electrolyzer can be run at its optimal operation point, the energy necessary can be easily calculated. The cost associated with this energy, however, is non-trivial to calculate as the system has access to both grid and PV power, where the cost for PV power is lower than the cost for grid power. In a first step we assume that no PV power is available which results in

$$c^{term} = -\frac{\hat{c}^{grid,avg} \cdot p^{el,nom}}{f^1(p^{el,nom})}, \tag{5.23}$$

with $\hat{c}^{grid,avg}$ being the estimated average cost for power taken from the grid and $p^{el,nom}$ is the nominal operation point of the electrolyzer. Looking at the units of the weight c^{term} we get

$$\frac{\frac{\text{CHf}}{\text{kWh}} \cdot \text{kW}}{\frac{\text{kg}}{\text{h}}} = \frac{\text{CHf}}{\text{kg}}, \tag{5.24}$$

as we would expect it to be. Choosing this terminal weight will lead to the situation where the electrolyzer is running all the time until the storage tanks are full. This is due to the fact that the electrolyzer essentially can run for free, as the power used to produce the hydrogen gets compensated by the terminal cost. As there is a small weight on turning the electrolyzer on or off, there is no incentive to turn the electrolyzer off before the storage tanks are full. This kind of a behaviour is non-optimal for the goal we are trying to achieve. As long as the hydrogen demand is not unreasonably high, it is valid to assume that a certain amount of PV power is available to use after the prediction horizon in order to produce excess hydrogen. Taking this into account in Equation (5.23) results in a lower terminal weight, such that running the electrolyzer without enough available PV power does not get fully compensated by the terminal cost. Therefore, a better alternative for the cost is

$$c^{term} = -\gamma \cdot \frac{\hat{c}^{grid,avg} \cdot p^{el,nom}}{f^1(p^{el,nom})}, \tag{5.25}$$

with $\gamma \in [0,\ 1]$ being a tuning variable. For all of the following experiments, unless otherwise stated, $\gamma = 0.9$ is chosen, as it prevents the electrolyzer from running all the time while still giving the electrolyzer an incentive to run if enough PV power is available. An intuitive understanding for this factor can be gained by rewriting (5.23) into the form

$$c^{term} = -\frac{\hat{c}^{grid,avg} \cdot (p^{el,nom} - \hat{p}^{PV}) + \hat{c}^{PV,avg} \cdot \hat{p}^{PV}}{f^1(p^{el,nom})}, \tag{5.26}$$

where $\hat{p}^{PV}$ is the estimated available PV power which can be used to produce excess hydrogen and $\hat{c}^{PV,avg}$ is the estimated cost for using PV power. For interested readers a possible way to estimate $\hat{p}^{PV}$ based on the hydrogen demand and PV power predictions is introduced in Appendix B. Because of its sensitivity to the hydrogen demand and the high influence the terminal weight has on the controller behaviour, it is not considered here, as it would significantly influence other experiments.

Additionally, the mass in the high pressure storage tanks, i.e. the 390bar and 900bar storage tanks, should be weighted in order to give the controller an incentive to move hydrogen from the low pressure to the high pressure tanks. The weight for this is calculated in a similar fashion as the weight for the whole storage mass, with the slight difference that we do not take PV power into account, making the calculation much easier resulting in

$$c^{term,high} = -\hat{c}^{grid,avg} \cdot \frac{p^{comp,max,1}}{\dot{m}^{comp,max,1}}, \tag{5.27}$$

where we assume that the compressor can be run at high compressor speeds after the prediction horizon has ended.

Concluding, we introduce one cost function term that weights the overall mass in the storage tanks and one cost function term that weights the mass in the high pressure storage tanks. In order to avoid large numeric values of the objective function, the initial mass in the storage can also be also incorporated into the objective function as a fixed offset. This way only deviations from the initial mass are incentivized or penalized.

5.2.2 Evaluation of the deterministic controller formulations

In order to evaluate the performance of the different problem formulations, their performance is compared to each other and to the simple bang-bang controller which is currently used in the move demonstrator in a deterministic setting. This is done for different initial state values, e.g. different initial storage tank levels. In a first experiment, the initial storage levels are chosen in the middle of the lower and upper bounds, such that the mass in the high pressure tanks is not actively constraint by the bounds. In a second experiment, the initial storage levels are chosen close to the lower bound, such that the optimization itself cannot be done without considering the lower bounds of the system. This leads to a situation where the electrolyzer is forced to work in sub-optimal operation points and has to turn on even though no PV power is available.
For both experiments, a constant cost for grid and PV power is assumed, equal to $c^{grid} = 1$ and $c^{PV} = 0.5$. The weight for electrolyzer state switching is chosen as $c^{switch} = 1$. Furthermore, all experiments are done with the same hydrogen demand realization. The

performance is calculated based on the electrolyzer and 390bar compressor's power consumption. In order to factor in different storage mass levels at the end of the simulation, weight terms similarly to the ones introduced in Equations (5.23) and (5.27), resulting in

$$C = \sum_{k=0}^{S-1} (p_k^{grid} \cdot c_k^{grid} \cdot \Delta t_k + (p_k^{el} + p_k^{comp} - p_k^{grid}) \cdot c_k^{PV}) \cdot \Delta t_k + c^{end} \cdot (m_S^{low} - m_0^{low}) + c^{end,high} \cdot (m_S^{high} - m_0^{high}), \tag{5.28}$$

where M is the total number of timesteps during the simulation and

$$c^{end} = -c^{grid} \cdot \frac{p^{el,nom}}{f^1(p^{el,nom})} \tag{5.29}$$

$$c^{end,high} = -c^{grid} \cdot \frac{p^{comp,high}}{\Delta m^{comp,high}}. \tag{5.30}$$

Note that we are subtracting the initial hydrogen inside the tanks in an effort to keep the performance score in a reasonable numerical range. Due to the same reason as mentioned in Section 5.2.1, weighting the mass inside the storage tanks at the end of the simulation should be done while incorporating some available PV power. Simply running the electrolyzer during times when no PV power is available should only be done if necessary and should therefore not be incentivized in the performance score. This leads to a slight adjustment of c^{end}, resulting in

$$\tilde{c}^{end} = -c^{grid} \cdot \frac{p^{el,nom}}{f^1(p^{el,nom})} \cdot 0.9. \tag{5.31}$$

Nonetheless, the scores for both c^{end} and $\tilde{c}^{end}$ are calculated in order to show which controller works best depending on how the stored hydrogen is weighted at the end of the simulation.

Table 5.1 shows the results of the two different experiments for the deterministic one-storage-stage MPC model, the deterministic two-storage-stage MPC model and the currently implemented bang-bang controller. Figure 5.1 shows the states of the system for all three controllers for the first experiment.

Controller	Exp. 1 - c^{end}	Exp. 1 - $\tilde{c}^{end}$	Exp. 2 - c^{end}	Exp. 2 - $\tilde{c}^{end}$
Bang-Bang Controller	1457.4196	1539.7449	1460.8001	1543.1253
One Storage-Stage Model	1441.1081	1359.6032	1446.5334	1464.1899
Two Storage-Stage Model	1406.5500	1335.8640	1400.2189	1433.2631

Table 5.1: The performance score C for two different experiments with two different terminal weights.

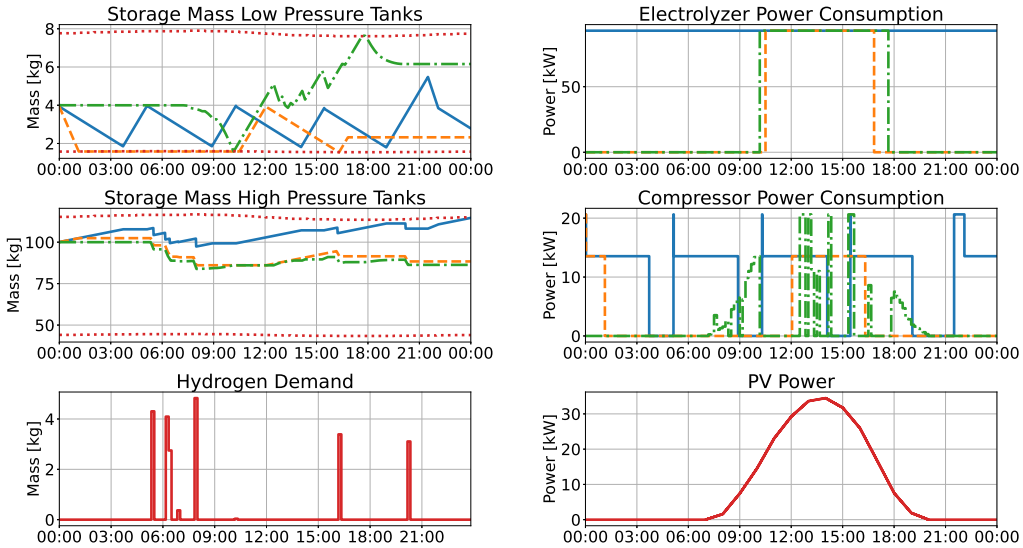

Figure 5.1: The behaviour of the bang-bang controller (blue, solid), the one-storage-stage MPC controller (orange, dashed) and the two-storage-stage MPC controller (green, dash-dotted) for the experiment 1 in a 1-day simulation. Additionally, in red all deterministic values are displayed, including the storage mass constraints, the hydrogen demand and the available PV power. In the case of the one-storage-stage MPC controller, the compressor actuation is based on a similar hysteresis controller which is also used for the bang-bang controller.

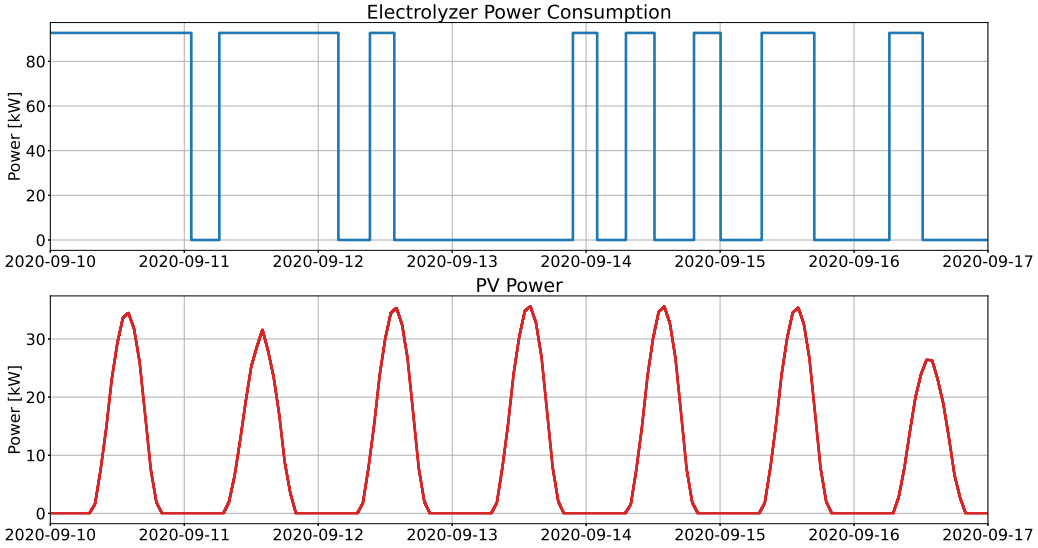

Figure 5.2: The electrolyzer actuation during a full week of the bang-bang controller. As the controller does not take PV power predictions into account, the electrolyzer sometimes runs during the night will remaining turned off during day, even if there is available PV power which could be used.

As it would be expected, the two-storage-stage MPC comes with the lowest score, therefore performing better than the other controllers according to the introduced metric. This is no surprise, as it optimizes both the electrolyzer and the compressor, such that the compressor uses available PV power at times when the electrolyzer is not running. Furthermore, it keeps the pressure in the 30bar tank high enough such that the 390bar compressor can be operated at higher compressor speeds. While with this score the performance of the bang-bang controller does not seem too bad, this is rather by chance, as the performance score was calculated over a single day. Figure 5.2 shows the electrolyzer power consumption for the bang-bang controller over a whole week, where the first day corresponds to the day on which the performance score for all controllers was determined. Because the bang-bang controller does not take any knowledge about available PV power into account, its performance highly depends on the available PV power and the hydrogen demand.
Concluding, the inclusion of the 390bar compressor into the MPC model brings an increase in performance as the MPC as it is able to allocate the PV power usage optimally between electrolyzer and compressor. In the following, the two-storage stage deterministic controller is extended to a stochastic setting.

5.3 Stochastic setting

In practice it is impossible to know how many cars will come to recharge at a particular day and when they will arrive within that day. In order to cope with this problem, the hydrogen demand model, developed in Chapter 4, is used to generate various samples for possible hydrogen demands which result in different scenarios. The optimization problem is then constructed in a way such that the average value over all scenarios is optimized.

$$\begin{aligned}
\min \quad & \frac{1}{M}\sum_{l=1}^{M}\Bigg[\sum_{i=0}^{N-1}\Big(c_i^{grid}\cdot {}_lp_i^{grid}\cdot\Delta t_i + c_i^{PV}\cdot({}_lp_i^{el}+{}_lp_i^{comp}-{}_lp_i^{grid})\cdot\Delta t_i \\
& \qquad + c^{switch}\cdot|\,{}_l\delta_i^{el}-{}_l\delta_{i-1}^{el}|\Big) + c^{term}\cdot({}_lm_N^{low}+{}_lm_N^{high}) + c^{term,high}\cdot {}_lm_N^{high}\Bigg] && (5.32)\\
\text{s.t.} \quad & {}_lm_{i+1}^{low} = {}_lm_i^{low} + {}_l\Delta m_i^{prod} - {}_l\Delta m_i^{comp} && (5.33)\\
& {}_lm_{i+1}^{high} = {}_lm_i^{high} + {}_l\Delta m_i^{comp} - {}_l\Delta m_i^{dem} && (5.34)\\
& m_i^{low,min} \le {}_lm_i^{low} \le m_i^{low,max} && (5.35)\\
& m_i^{high,min} \le {}_lm_i^{high} \le m_i^{high,max} && (5.36)\\
& m_i^{low,end} \le {}_lm_N^{low},\ m_i^{high,end} \le {}_lm_N^{high} && (5.37)\\
& {}_l\Delta m_i^{prod} = f^1({}_lp_i^{el})\cdot\Delta t_i && (5.38)\\
& {}_l\Delta m_i^{comp} = f^c({}_lp_i^{comp})\cdot\Delta t_i && (5.39)\\
& p^{el,min}\cdot {}_l\delta_i^{el} \le {}_lp_i^{el} \le p^{el,max}\cdot {}_l\delta_i^{el} && (5.40)\\
& 0 \le {}_lp_i^{comp} \le {}_lp_i^{comp,max} && (5.41)\\
& {}_lp_i^{comp,max} = h^c({}_lm_i^{low}, \bar{m}^{30,15bar}) && (5.42)\\
& {}_lp_i^{el} + {}_lp_i^{comp} \le {}_lp_i^{grid} + p_i^{PV} && (5.43)\\
& {}_lm_0^{low} = m_0^{low}, \quad {}_lm_0^{high} = m_0^{high}, \quad {}_l\delta_{-1}^{el} = \delta_{-1}^{el} && (5.44)\\
& {}_lp_0^{el} = p_0^{el}, \quad {}_l\delta_0^{el} = \delta_0^{el}, \quad {}_lp_0^{comp} = p_0^{comp} && (5.45)\\
& {}_l\delta_i^{el} \in \{0,1\} && (5.46)\\
& m_0^{low},\ m_0^{high},\ \delta_{-1}^{el} \text{ given.}
\end{aligned}$$

The only schematic differences to the problem introduced before are the additional sum in the objective function (5.32) which implements the expected value of the scenarios and the coupling constraints (5.44) and (5.45). The coupling constraints are necessary as all scenarios have to agree on a joint input value for the first timestep which can then be applied to the system.

Unfortunately, this problem formulation comes with an issue. By introducing M scenarios, the number of optimization variables is essentially multiplied by M. While this might not be problematic for linear programs, it is highly problematic for mixed-integer linear programs due to the combinatorial nature. The convergence time of the problem scales poorly with the number of scenarios such that the controller cannot be run in real-time anymore, if a reasonable amount of scenarios, e.g. $M = 10$, and a reasonable length of the prediction horizon, e.g. 1 or 2 days, is required.

An usual approach to prevent the problem of an unreasonable increase in the number of optimization variables is the introduction of a control law which allows for the removal of scenario-specific input values. In the case of a mixed-integer problem, however, this is not a feasible solution as there is no obvious way of designing a control law which fixes the binary variables.

In the following paragraphs different possible heuristics are proposed which can approximately solve the optimization problem much more efficiently than the problem stated in (5.32)-(5.46).

5.3.1 Complete decoupling of the scenarios

The major difficulty that leads to an increase in the convergence time is not necessarily the number of binary decision variables but the fact that all of them are coupled with each other. Fixing a binary variable in one scenario will have an impact on all other scenarios. A simple approach is to remove the coupling constraints and solve the optimization problem independently for each scenario. While solving the problem in this case is somewhat efficient, the question is, how do we choose the inputs? What are we doing if different scenarios have different binary decision variables in the first timestep, e.g. different inputs for whether to turn on the electrolyzer or not. In a first step, we are determining how the binary inputs, i.e. the electrolyzer status is fixed. For this we simply solve two almost identical sub-problems, one where the first electrolyzer status is equal to zero and another one where it is equal to one. The problem resulting in a lower objective function value is chosen as a first electrolyzer input. This procedure can be efficiently implemented in parallel, as these problems can be solved independently from each other.

The question on how the continuous input values of the system, i.e. p_0^{el} and p_0^{comp} are fixed remains. Regarding this, two different approach are introduced in the following paragraphs.

Averaging of the input

Probably the simplest solution is to take the average of all initial inputs. For a linear problem this will lead to a solution which remains inside all possible realizations. For the linear as well as the mixed-integer case, there is no reason why this kind of a solution should be optimal in any sense. Nevertheless, it is simple and straight-forward to implement and will therefore be tested.

Relative majority voting of the input

In most cases, a majority of the scenarios is choosing the same input for the electrolyzer and the compressor. Therefore, it is a reasonable approach to choose the inputs which a relative majority of the scenarios deem to be optimal, while hoping that the input will also be close to optimal for the rest of the scenarios.

Both of these approaches might end up with a solution which is not feasible for each scenario, which takes away a large part of the robustness when simply applied to the system. Therefore, before applying the inputs to the system, it is checked whether they are feasible for all scenarios. If this is not the case, the averaging approach tries out

whether the relative majority input is feasible. If this is not the case, the inputs from each scenario are iterated and checked for feasibility for all scenarios. Once a feasible input is found, it is applied to the system. If none of the inputs is feasible for all scenarios, the electrolyzer and the compressor are fully actuated. The last case never happened during simulation and should only happen in specific constellations when the high pressure tanks are extremely close to the lower bounds. In these cases, it is usually a good idea to produce as much hydrogen as possible and move it to the higher pressure tanks.

5.3.2 Progressive Hedging

Progressive Hedging is a popular algorithm in stochastic programming and uses the principles of the augmented Lagrangian method, where a combination of penalty term and Lagrangian multipliers are used in order to relax an equality constraint. [18] demonstrates the use of progressive hedging for a mixed-integer linear program by using a L_1 norm for the penalty function and an adjusted Lagrangian multiplier update step. A similar algorithm is implemented for our system in order to relax the coupling constraints which has the form:

Algorithm 1: Progressive Hedging

Set k=0

Let $\boldsymbol{x}_l^{(k)}$ contain the input values to the system in the first timestep

Solve the relaxed problem with no coupling constraints

Compute $\bar{\boldsymbol{x}}^{(0)} = \frac{1}{M}\sum_{l=1}^{M} \boldsymbol{x}_l^{(0)}$ and $\boldsymbol{\lambda}_l^{(0)} = \boldsymbol{\rho} \cdot (\boldsymbol{x}_l^{(0)} - \bar{\boldsymbol{x}}^{(0)}), \quad \forall l = 0, 1, 2, ..., M$

while $\sum_{l=0}^{M} |\boldsymbol{x}_l^{(k)} - \bar{\boldsymbol{x}}^{(k)}|_1 > \epsilon$ *or* $k < k_{max}$ **do**

  Solve the augmented relaxed problem with additional cost function term: $\sum_{l=1}^{M} ... + (\boldsymbol{\lambda}_l^{(k)})^T \boldsymbol{x}_l + \boldsymbol{\rho} \cdot |\boldsymbol{x}_l^{(k)} - \bar{\boldsymbol{x}}^{(k)}|_1$

  Update $\bar{\boldsymbol{x}}^{(k+1)} = \frac{1}{M}\sum_{l=1}^{M} \boldsymbol{x}_l^{(k+1)}$ and

  $\boldsymbol{\lambda}_l^{(k+1)} = \boldsymbol{\rho} \cdot sign(\boldsymbol{x}_l^{(k+1)} - \bar{\boldsymbol{x}}^{(k+1)}), \quad \forall l = 0, 1, 2, ..., M$

  Set k = k + 1

end

where the variable $\boldsymbol{x}_l$ contains the input values within the coupling horizon, i.e. the power consumption of the electrolyzer and the 390bar compressor for the first timestep. The algorithm is iteratively solving relaxed problems until the solutions of all scenarios are sufficiently close together or until a maximum number of iterations is reached. The algorithm has the tuning factor $\boldsymbol{\rho}$ included which has a large impact on the convergence of the algorithm. Choosing it too small leads to an unreasonably long convergence time, choosing it too large leads to oscillations of the solution and generally cyclic behaviour. While $\boldsymbol{\rho}$ could be chosen as a scalar value, better convergence is achieved when different values of $\boldsymbol{\rho}$ are used for different elements in $\boldsymbol{x}_l$. Therefore, $\boldsymbol{\rho}$ is also chosen as a vector and multiplications with it should be interpreted as element-wise multiplications.

It can happen that the algorithm is not converging, such that the maximum number of iterations is reached. In this case, the solution is similar to the one where no coupling

constraint at all are used. If this is the case, the input is calculated based on the relative majority voting system, i.e. the input that occurs most often is chosen if it is feasible.

5.3.3 Comparison of the approaches

In order to compare the approaches, the prediction horizon length is chosen as 24 hours, the number of scenarios is fixed to $M = 10$ and the timestep size is chosen as 60 minutes, i.e. $\Delta t = 1$. In this way the problem with coupling constraints is converging within a reasonable amount of time, as the total number of timesteps is only 24. The different models are run independently from each other with the same deterministic PV power, same hydrogen demand realization, which is unknown to the MPC, and the same hydrogen samples, which are available to the MPC, for a simulation time duration of 24 hours. The initial storage mass values are set to 4 and 80 kilograms for the low pressure and high pressure tanks respectively. The objective function of the resulting trajectory is calculated, which should be somewhat close for all approaches as they have the exact same data available to them. As this comparison is sensitive to the hydrogen demand realization, i.e. the actual hydrogen demand affecting the system, and the hydrogen samples used, i.e. the hydrogen demand samples produced by our hydrogen demand model, the experiment is repeated multiple times for different hydrogen realization and samples and the results over all experiments are averaged. The performance of each trial is calculated similarly to the deterministic case with

$$C = \sum_{k=0}^{M-1} (p_i^{grid} \cdot c_k^{grid} + (p_i^{el} + p_i^{comp} - p_i^{grid}) \cdot c_k^{PV}) \\ + c^{end} \cdot (m_M^{low} - m_0^{low}) + c^{end,high} \cdot (m_M^{high} - m_0^{high}), \tag{5.47}$$

where

$$c^{end} = -c^{grid} \cdot \frac{p^{el,nom}}{f^1(p^{el,nom})} \tag{5.48}$$

$$c^{end,high} = -c^{grid} \cdot \frac{p^{comp,high}}{\Delta m^{comp,high}}. \tag{5.49}$$

The averaged results of 50 independent experiments can be seen in Table 5.2.

Approach	Averaged performance score	Avg. Convergence Time [s]
Coupled	401.5974	83.8498
Decoupled - Relative majority	410.1493	0.8549
Decoupled - Averaging	485.0009	0.8940
Progressive Hedging	411.7554	3.0617

Table 5.2: The performance score C and the average convergence time for the optimization problem in one timestep.

As it would be expected, the coupled approach provides the best objective function value, but also has the highest convergence time. An upper time limit of 300 seconds is used for all optimization problems, which is reached multiple times by the coupled approach. This makes it not usable for real-time applications. The averaging approach provides the worst results, which is expected as it tends to almost never operate at optimal operation unless all scenarios jointly decide on it. The relative majority approach and the

progressive hedging are somewhat similar for the objective function value, however, the progressive hedging algorithm has difficulties to converge, if the initial input values of the scenarios differ to much from each other. This leads to reaching the maximum number of iterations for the algorithm, leading to the increase runtime in this case. Furthermore, if the algorithm does not converge, the input is most likely not optimal either.

Concluding, the relative majority voting system provides the best trade-offs between convergence time and objective function value and will be used for further evaluations of the stochastic MPC formulation. In the following, the performance of the stochastic MPC is tested with the decoupling majority voting input selection implemented.

5.3.4 Evaluation of the stochastic controller formulation

In order to evaluate the overall performance of the stochastic MPC controller, the stochastic MPC formulation is compared to the deterministic MPC formulations with the same experiments and the same performance metric as already conducted in Section 5.2.2. The same hydrogen realization is used, however the MPC has no knowledge about it and is only using samples drawn from the same distribution as the hydrogen realization. The results are shown in Table 5.3, while Figure 5.3 shows the system states and inputs for the deterministic and stochastic two-storage-stage controller for second experiment.

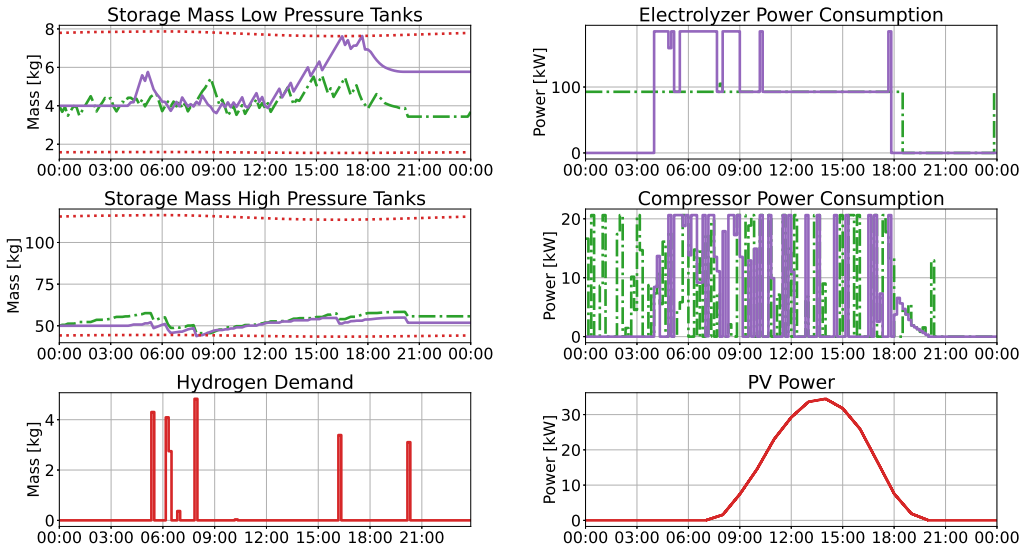

Figure 5.3: The behaviour of the deterministic two-storage-stage MPC (green, dash-dotted) and the corresponding stochastic two-storage-stage MPC controller (violet, solid) for the second experiment. Additionally, in red all deterministic values are displayed, including the storage mass constraints, the hydrogen demand and the available PV power. The stochastic controller does not have any concrete information about the hydrogen demand, apart from its probability distribution. The PV power is assumed to be exactly known for both controllers.

The stochastic approach performs incredibly well for the first experiment. This should be expected as in the case where the constraints in the high pressure tanks are not active, all scenarios have a similar objective and therefore similar inputs as they would

Controller	Exp. 1 - c^{end}	Exp. 1 - $\tilde{c}^{end}$	Exp. 2 - c^{end}	Exp. 2 - $\tilde{c}^{end}$
Bang-Bang Controller	1457.4196	1539.7449	1460.8001	1543.1253
Det. 1-Storage-Stage MPC	1441.1081	1359.6032	1446.5334	1464.1899
Det. 2-Storage-Stage MPC	1406.5500	1335.8640	1400.2189	1433.2631
Stoch. 2-Storage-Stage MPC	1404.4507	1335.3103	1486.6310	1508.6563

Table 5.3: The performance score C of the bang-bang controller, the deterministic one-storage-stage MPC, the deterministic two-storage stage MPC and the stochastic two-storage-stage MPC for two different experiments and two different terminal weights.

have in a deterministic setting. The more interesting case is the second experiment. As the stochastic setting has no precise knowledge about the exact hydrogen demand, it results in a sub-optimal behaviour compared to the deterministic case. Moreover, if the hydrogen available at the last timestep is weighted with c^{end}, it performs worse than the bang-bang controller in this situation. However, this is only the case in this particular setting. Choosing $\tilde{c}^{end}$ shows that it still performs better than the bang-bang controller on our actual performance metric. Nonetheless, the stochastic MPC shows sub-optimal behaviour if it gets close to the lower bounds. This is because the MPC still tries to turn on the electrolyzer only when there is PV power available. In such critical situations it might be more beneficial to slightly adjust the terminal cost in the MPC. By adjusting the terminal weight, the MPC is penalized less for running the electrolyzer with no PV power available. Running the stochastic MPC for experiment 2 with adjusted terminal weight equal to $c^{end} = -c^{grid} \cdot \frac{p^{el,nom}}{f^1(p^{el,nom})}$, results in no penalty to run the electrolyzer at nominal power. The states and inputs for this configuration are shown in Figure 5.4 and the resulting performance score is summarized in Table 5.4.

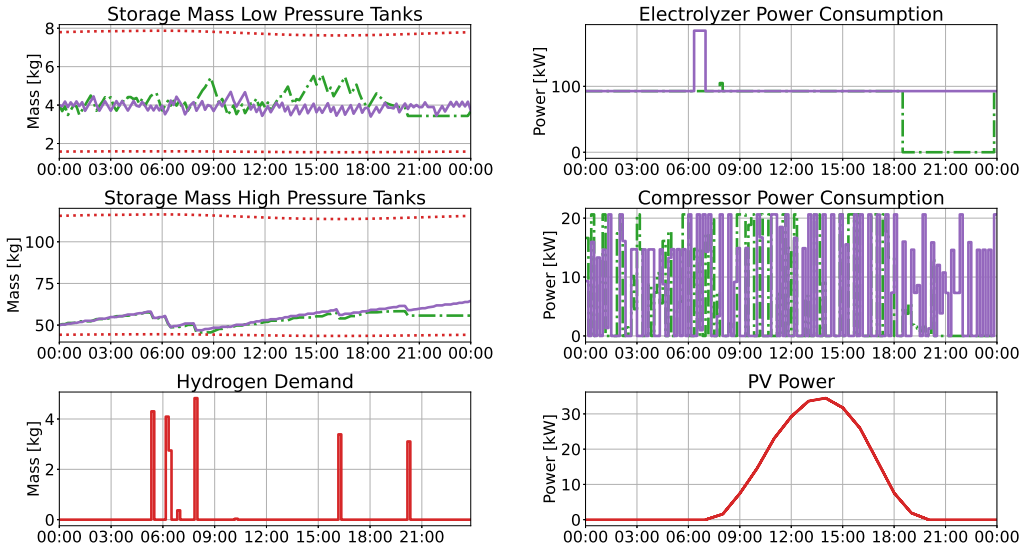

Figure 5.4: The behaviour of the deterministic two-storage-stage MPC (green, dash-dotted) and the corresponding stochastic two-storage-stage MPC controller (violet, solid) for the second experiment. Additionally, in red all deterministic values are displayed, including the storage mass constraints, the hydrogen demand and the available PV power. The stochastic controller does not have any concrete information about the hydrogen demand, apart from its probability distribution. The PV power is assumed to be exactly known for both controllers.

Controller	Exp. 2 - c^{end}	Exp. 2 - $\tilde{c}^{end}$
Bang-Bang Controller	1460.8001	1543.1253
Det. 1-Storage-Stage MPC	1446.5334	1464.1899
Det. 2-Storage-Stage MPC	1400.2189	1433.2631
Stoch. 2-Storage-Stage MPC	1486.6310	1508.6563
Adjusted stoch. 2-Storage-Stage MPC	1407.8921	1495.5351

Table 5.4: The performance score C for the adjusted stochastic two-storage-stage MPC and the second experiment. For the terminal weight c^{end} it performs much better, which is no surprise as the electrolyzer is running all the time. For the more relevant terminal weight $\tilde{c}^{end}$ the performance score also shows improvements of the adjusted MPC compared to the original stochastic two-storage-stage MPC.

This clearly shows that a better performance can be achieved if the terminal weight is adjusted to the situation. Obviously this would need to be automatically adjusted, which raises the need for a higher-level controller, which influences the behaviour of the MPC by adjusting the terminal weight. Further research into this direction is left as future work.
Concluding, if the system is not close to the lower bounds, the stochastic MPC performs similarly to its deterministic counterpart. In the case where the system is close to the lower bounds, it is reasonably robust against constraint violations, however the inputs are generally not optimal anymore. This can be mitigated by adjusting the terminal weight of the MPC.
This concludes the performance evaluation of the developed controllers. In the rest of

this chapter, it is examined how the MPC performs for up-scaled versions of the original problem structure.

5.4 Scalability of the MPC

In the following, the advantages of prolonging the prediction horizon to a length of 2 days instead of 1 day is presented and the behaviour of the MPC in the presence of a significant increase in available PV power is examined.

5.4.1 Extending the prediction horizon length

The extension of the prediction horizon to a length of two days comes with an increase in the number of optimization variables, which itself will increase the convergence time of the optimization problem. Figure 5.5 shows the system states and inputs for a prediction horizon of two days used in a deterministic two-storage-stage MPC and a simulation length of two days.

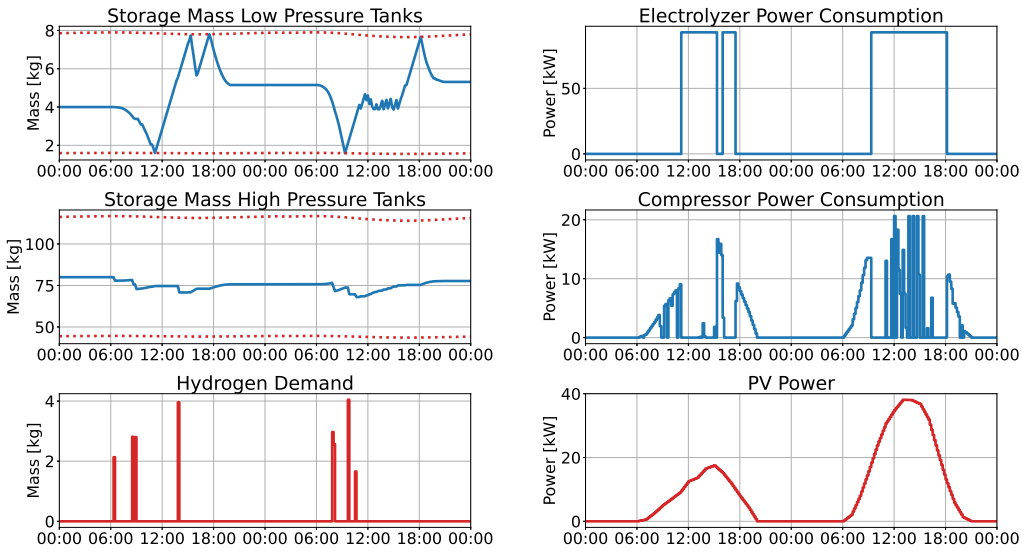

Figure 5.5: Simulation results for a deterministic two-storage stage MPC with a prediction horizon length of two days and a total simulation duration of two days.

The first day provides less available PV power than the second day, which is why the MPC is running the electrolyzer for a longer time during the second day. Overall, including a second day into the prediction horizon allows for certain adjustments compared to the one day ahead prediction, if either the available PV power or the hydrogen demand differs. A two-day ahead prediction can definitely improve the long-term performance of the controller, however it highly depends on the overall configuration how big the difference will be.

5.4.2 PV power upscaling

Up to this point the considered PV power was always smaller than the nominal power of the electrolyzer, i.e. the available PV power was not able to completely power the electrolyzer itself. In the following, we consider an upscaled version of the PV power system, such that the whole system can be powered by PV power itself. For this the available PV power is scaled up by a factor of 4. The results of the deterministic one day-ahead MPC is shown in Figure 5.6.

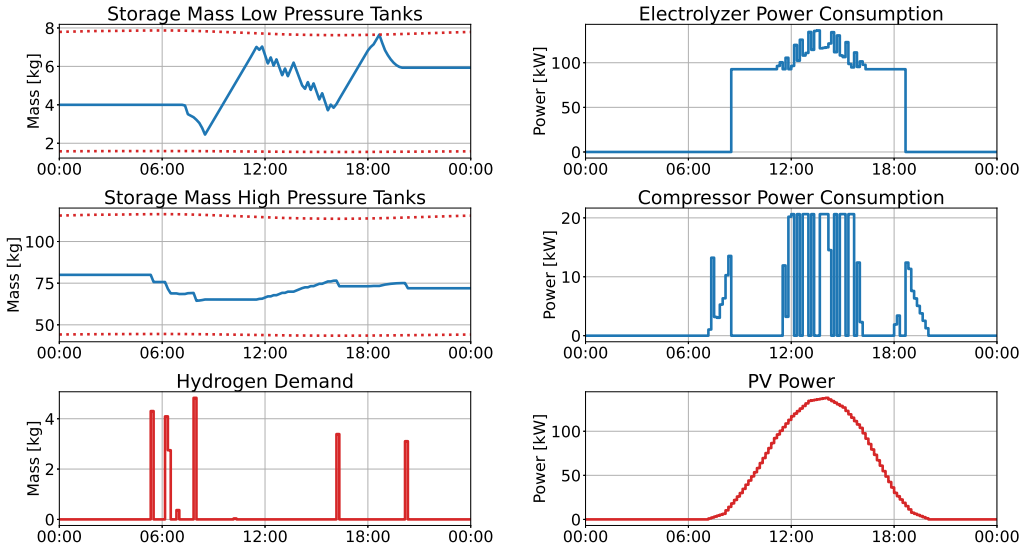

Figure 5.6: Simulation results for a deterministic two-storage stage MPC with a prediction horizon length of one day and a total simulation duration of one day. The PV power is scaled up by a factor of 4, resulting in peak power production of around 140kW. This enables the system to power the electrolyzer solely on PV power during peak hours.

Note that the MPC decides to run the electrolyzer at a power consumption higher than the nominal one. Grid power is mostly used to provide power during the the late morning and early afternoon.

5.5 Economically driven MPC

Up to this point only a constant cost term for the power taken from the grid is taken into account in the MPC. In a realistic setting, it is not only desired to maximize the use of PV power for the system but also minimize the overall cost, given an electricity price prediction for the grid power. By including a varying cost term in the MPC, the controller has the possibility to run the electrolyzer when either enough PV power is available or when the electricity price is lower than usual. In order to test the behaviour of the MPC in such situation, a realistic electricity price is used as a cost for c_i^{grid} taken from the Swiss spot market price in January 2017 [19]. The cost for used PV power is still assumed to be constant over the prediction horizon, however, it is calculated from the average value

for the grid electricity cost by

$$c_i^{PV} = \frac{1}{2 \cdot N} \sum_{k=0}^{N-1} c_k^{grid},$$

i.e. half of the average electricity cost over the prediction horizon. The upscaled version for PV power from the previous section is also used here. In this case the MPC uses the average grid electricity prices in order to calculate the terminal weights c^{end} and $c^{end,high}$. Simulation results for the deterministic one day ahead MPC can be seen in Figure 5.7.

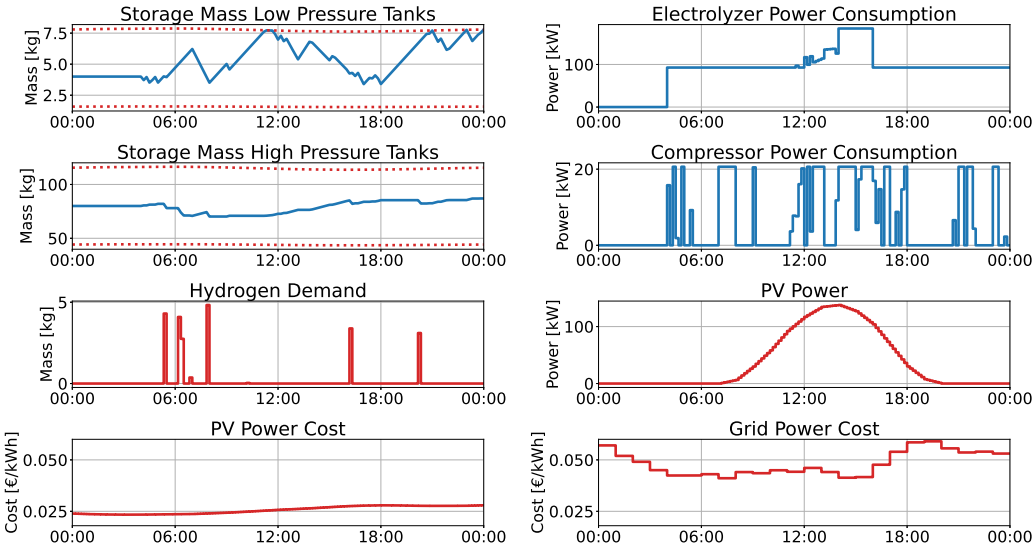

Figure 5.7: Simulation results for a deterministic two-storage-stage MPC with varying cost terms on the used grid power. It leads to the electrolyzer working at sub-optimal operation points during times when the prices for grid power are low, as for example it is the case at around 15:00.

The MPC has multiple incentives to turn the electrolyzer on at a particular moment in time. Either if the PV power is high enough or if the electricity prices are below the average. Moreover, the MPC makes the decision of running the electrolyzer at sub-optimal operation points during times when electricity is cheap.

Chapter 6

Conclusion

In the following we summarize the overall work done as a part of this thesis and give an outlook to possible future work that can be done.

The thesis presented a model-based control framework for the optimal management of a recharge station for fuel-cell vehicles, based on the real system which is placed at Empa. In comparison to the state of art, it investigated the impact of including compressor stages into the optimization problem and solved the induced increased complexity of the problem by evaluating different problem relaxation approaches against each other and choosing the one that performs the best.

The major parts of the work consisted of creating suitable models for different parts of the system. Firstly, the electrolyzer and compressor stages were modelled in a simplified fashion. If not enough quality data was available to fit the models, additional experiments were done on the real system. The models were evaluated by forward integrating them and comparing them to measurement data. The results show a really good tracking for the electrolyzer model and a reasonable tracking performance for the compressors.

Next to this, different pressure-temperature-density relationships for gases were examined in order to find one which can be used to calculate the mass contained in a storage tank with known pressure and temperature. By using experimental data from tables excellent results in performance were observed which were examined by comparing the mass changes inside the tank according to the pressure differences calculated through the respective relationship to the mass in- and outflow.

As the storage tanks are physically limited by pressures values and the relationship between pressure and mass depends on the temperature, temperature predictions for the storage tanks were necessary. As there was no research work done in a similar setting to ours, in a first step a detailed evaluations of relevant input features for possible models was conducted. Afterwards, inspired by building temperature predictions, two different models were implemented and tested against each other. The neural network was better suited in capturing the nonlinear relationship between input variables and the storage tank temperature.

Parallel to this, the predictions for the PV power were done where two different approaches were implemented and tested. The end-result consisted of a neural network which is pretrained on over 3 years worth of data and is consistently retrained based on recent data

with a rolling window approach in order to account for slowly varying external factors.

Next, the available hydrogen demand data was analyzed for its stochastic properties, the problem structure was broken down to simpler variables and suitable models were fitted to the data. The derived models were verified by performing a chi-squared test on them.

All the previously identified models were combined into an deterministic control structure where we, in a first step, compared the difference in performance that we gain by including a compressor stage into the optimization problem, which were considerable and justified the added complexity to the model. This model was transformed into a stochastic one, by introducing multiple hydrogen demand scenarios and optimizing over all of them. As this problem scales increasingly poorly in its original form, different heuristics to relax the problem were examined in a downsized problem configuration. The best approach was determined to be a heuristic that chooses the input that a relative majority of scenarios deem to be optimal and then checking the feasibility of it. This resulting heuristic was applied to the original stochastic problem making it easily run in real-time for a prediction horizon length of one day and 10 scenarios. The performance of the resulting stochastic MPC is tested against the deterministic models, which shows that in cases where the storage tanks are not almost empty, the solution of the stochastic problem is essentially as good as the solution for the deterministic problem. In cases where we are close to the lower storage tank constraints, the stochastic MPC still shows a reasonable performance. Possible methods to improve the performance of the MPC in such situations were shortly mentioned and are going to be discussed in a little bit more detail in Section 6.1.4. Lastly, the MPC was tested for various different configurations in order to show its flexibility to being able to achieve different objectives.

6.1 Outlook

In the following an outlook on possible future work for the different sub-parts is given.

6.1.1 System modelling

The biggest weakness of the current model are the compressors. As they do not take pressure gradients into account, there will almost always be an offset in the predicted mass flow through them. A detailed identification of the compressors based on input and output pressures can be done, which might give more insight into the process of how the compressor work in detail.

The storage tank predictions can be done with an autoregressive neural network approach instead of the single-shot model which is only able to predict 24 hour prediction in one piece. This would allow for better online-correction of the prediction as it could be updated whenever new measurements for the storage tank temperature or new ambient temperature forecast data is available.

Lastly, the power taken from the grid needs to be rectified which involves certain losses. If the precise amount of power taken from the grid plays a large roll this effect should be included in the model. The relationship between input and output of a common rectifier

usually shows a linear behaviour such that the inclusion of it in the optimization problem should not be difficult.

6.1.2 PV Power Prediction

The current PV power prediction is currently split into two different models, one model to predict sunny days and one model to predict all other days. Instead of classifying days into 2 categories a better approach could be to assign a score between 0 and 1 to each day, 1 corresponding to perfectly sunny and 0 to completely cloudy. This classification could be done with a neural network instead of the currently implemented weather classification. The resulting score can then be used as an input to the PV power prediction model, such that only one model is necessary for all weather conditions. In a next step, the model structure should be changed to an autoregressive one, similarly to the one proposed for the storage tank temperature predictions. This can be combined with a weather classification which predicts sub-intervals during a day, such that a day can have times when it is cloudy and other times when it is sunny. The learning of these models would need to handle the forecasting errors contained in the forecasts in some way, as these errors substantially influence the training of such a model.

6.1.3 Hydrogen Demand

The main disadvantage of the current hydrogen model is the needed estimate of the number of active customers. As long as there are not too many customers, the modelling of the hydrogen demand through a probabilistic model can be problematic. In order to cope with this problem, a model can be constructed which takes the last time a customer refuelled their car and outputs a probability function of when they will come to refuel the car for the next time. This approach was already partially tested in this thesis, however, with the current amount of data such a model is incredibly hard to train and is generally problematic from a privacy point of view. Furthermore, there is no automated way to transfer refuelling data into the database, such that such an approach would not even be implementable in the real system at the moment.

6.1.4 Control Design

In a first step, detailed testing of the current MPC on the real system should be done. After this, one might want to go into more detail about the choice for the terminal weight in the MPC. As we have seen, the MPC behaviour highly depends on the choice of the terminal weight. A higher-level controller can be developed which provides a suitable terminal weight. The higher-level controller can either work on long-term predictions, e.g. over a week and adjust the weight according to the estimated PV power and estimated hydrogen demand. Another approach would be to solely look at the current storage mass level and adjust the terminal weight in such a way that the MPC results in more conservative action if we are close to the bounds and a more risk-taking actions if it is away from the constraints.

Additionally, depending on the objective that one wants to achieve with the controller, the objective function can be easily changed to incorporate more general settings. First steps of possible configurations were already introduced in Section 5.4.

Bibliography

[1] Jock Gilchrist Isabella Burch. Survey of Global Activity to Phase Out Internal Combustion Engine Vehicles. https://theclimatecenter.org/wp-content/uploads/2020/03/Survey-on-Global-Activities-to-Phase-Out-ICE-Vehicles-update-3.18.20-1.pdf. Accessed: 2021-01-23.

[2] Auto AG Truck collaborating with Hyundai Hydrogen Mobility. https://www.luzern-business.ch/en/news/detail/?nid=14364. Accessed: 2021-01-23.

[3] Hydrogen basics. https://afdc.energy.gov/fuels/hydrogen_basics.html. Accessed: 2021-01-23.

[4] Aristeidis Tsakiris. Analysis of hydrogen fuel cell and battery efficiency. https://c2e2.unepdtu.org/wp-content/uploads/sites/3/2019/09/analysis-of-hydrogen-fuel-cell-and-battery.pdf. Accessed: 2021-01-23.

[5] Hydrogen energy storage. https://energystorage.org/why-energy-storage/technologies/hydrogen-energy-storage/. Accessed: 2021-01-23.

[6] move - Mobility of the Future. https://www.empa.ch/web/move. Accessed: 2021-01-23.

[7] Teesing develops 700 bar filling technique. https://www.teesing.com/en/page/news-items/sustainable-700-bar-filling-technique-for-furure-hydrogen-cars. Accessed: 2021-01-23.

[8] COSMO forecasting system. https://www.meteoswiss.admin.ch/home/measurement-and-forecasting-systems/warning-and-forecasting-systems/cosmo-forecasting-system.html. Accessed: 2021-01-23.

[9] P. Gabrielli, B. Flamm, A. Eichler, M. Gazzani, J. Lygeros, and M. Mazzotti. Modeling for optimal operation of pem fuel cells and electrolyzers. In *2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, pages 1–7, 2016.

[10] Benjamin Flamm, Christian Peter, Felix N. Büchi, and John Lygeros. Electrolyzer modeling and real-time control for optimized production of hydrogen gas. *Applied Energy*, 281:116031, 2021.

[11] National Institute of Standards and Technology. Thermophysical Properties. https://webbook.nist.gov/cgi/fluid.cgi?ID=C1333740&Action=Page. Accessed: 2021-01-23.

[12] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[13] François Chollet et al. Keras. `https://keras.io`, 2015.

[14] Matthew J. Reno and Clifford W. Hansen. Identification of periods of clear sky irradiance in time series of ghi measurements. *Renewable Energy*, 90:520 – 531, 2016.

[15] R N Dows and E J Gough. Pvusa procurement, acceptance, and rating practices for photovoltaic power plants. 9 1995.

[16] D R Myers. Evaluation of the performance of the pvusa rating methodology applied to dual junction pv technology: Preprint (revised). 7 2009.

[17] Karl Pearson. *On the Criterion that a Given System of Deviations from the Probable in the Case of a Correlated System of Variables is Such that it Can be Reasonably Supposed to have Arisen from Random Sampling*, pages 11–28. Springer New York, New York, NY, 1992.

[18] Valentin Kaisermayer, Daniel Muschick, Martin Horn, and Markus Gölles. Progressive hedging for stochastic energy management systems. *Energy Syst.*, pages 1–29, Aug 2020.

[19] Swiss spot market price. `https://www.epexspot.com/en`, Jan. 2017.

[20] MOSEK - Modeling Cookbook - Mixed integer optimization. `https://docs.mosek.com/modeling-cookbook/mio.html`. Accessed: 2021-01-23.

Appendix A

Compressor model in a mixed-integer linear program

In the following the constraints used to model the compressor dynamics are introduced. In a first step we are going to introduce

$$\alpha_i := \begin{cases} 1 & \text{if } m_i^{low} \geq m_i^{low,15bar} \\ 0 & \text{otherwise} \end{cases}, \tag{A.1}$$

where $m_i^{low,15bar}$ indicates the mass inside the low pressure tanks corresponding to a pressure value of 15bar. α_i is used to indicate at which compressor speed the 390bar compressor works. (A.1) can be implemented by using the 'big-M' trick as described in [20]:

$$m_i^{low} \leq m_i^{low,15bar} + \alpha_i \cdot 1.2 \cdot m_i^{low,max}, \tag{A.2}$$
$$m_i^{low} \geq m_i^{low,15bar} - (1 - \alpha_i) \cdot 1.2 \cdot m_i^{low,max}, \tag{A.3}$$

where $1.2 \cdot m_i^{low,max}$ has to be sufficiently large such that the inequality constraints can always be satisfied with either α_i being zero or one. Due to the construction of $m_i^{low,max}$ this is always guaranteed. α_i can then be used in order to assign the right values to p_i^{comp} and Δm_i^{comp}:

$$p_i^{comp,1} \leq p^{comp,max,1} \cdot (1 - \alpha_i) \tag{A.4}$$
$$p_i^{comp,2} \leq p^{comp,max,2} \cdot \alpha_i \tag{A.5}$$
$$p_i^{comp} = p_i^{comp,1} + p_i^{comp,2} \tag{A.6}$$
$$\dot{m}_i^{comp} = \frac{\dot{m}^{comp,max,1}}{p^{comp,max,1}} \cdot p_i^{comp,1} + \frac{\dot{m}^{comp,max,2}}{p^{comp,max,2}} \cdot p_i^{comp,2} \tag{A.7}$$
$$\Delta m_i^{comp} = \dot{m}_i^{comp} \cdot \Delta t_i, \tag{A.8}$$

where $p^{comp,max,1}$ and $p^{comp,max,2}$ are the corresponding power consumption for the two different compressor speeds and $\dot{m}^{comp,max,1}$ and $\dot{m}^{comp,max,2}$ the corresponding mass-flows, respectively.

Appendix B

Terminal weight heuristic

As previously discussed for the calculation of the terminal weight it makes sense to incorporate a certain amount of PV power that can be used to generate excess hydrogen after the end of the prediction horizon. In order to estimate this amount, we are first going to estimate the amount of PV power that will be used to generate the necessary hydrogen to compensate the hydrogen demand. For this we are going to approximate the time duration $\Delta\hat{t}_{day}^{run}$ during which the electrolyzer will be turned on during one day in order to produce as much hydrogen as there is demand. This can be done by

$$\Delta\hat{t}_{day}^{run} = \frac{\Delta\hat{m}_{day}^{dem}}{\Delta\hat{m}_{day}^{prod}}, \tag{B.1}$$

where

$$\Delta\hat{m}_{day}^{dem} = \sum_{i=0}^{N-1} \Delta m_i^{dem} \cdot \frac{1}{24} \sum_{i=0}^{N-1} \Delta t_i, \tag{B.2}$$

and

$$\Delta\hat{m}_{day}^{prod} = f^1(p^{el,nom}) \cdot 24, \tag{B.3}$$

where $\Delta\hat{m}_{day}^{dem}$ denotes the hydrogen demand during a day and $\Delta\hat{m}_{day}^{prod}$ the amount of hydrogen the electrolyzer can produce during one day. Assuming that the time during which the electrolyzer is run is split up to the maximum values of the available PV power in the prediction horizon, we can estimate the amount of PV power that is left over to produce excess hydrogen. Depending on how long one wants to run the electrolyzer there will be less or more PV power available. In the case where one wants to fill up the tank as fast as possible, it would run the whole day, resulting in the average being taken over the whole remaining day. In the case where one does not need to fill up the tank as fast, the time duration can be chosen shorter, and therefore the average will be taken over a smaller time-interval resulting in assuming that there is more available PV for excess PV production. All of this does not take into account that the PV power that is not used by the electrolyzer can be used by the compressor. In the end, this whole calculation remains a heuristic which can be developed further in order to also include the compressor in the calculation. Either way, once the excess PV power $\hat{p}^{PV}$ is approximated the terminal weight can be calculated by

$$c_{end} = -\frac{\hat{c}^{grid,avg} \cdot (p^{el,nom} - \hat{p}^{PV}) + \hat{p}^{PV,avg} \cdot \hat{c}^{PV}}{f^1(p^{el,nom})}. \tag{B.4}$$

The whole approach can be easily extended to the stochastic setting by taking the mean value of the hydrogen demand over all scenarios.

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Modelling, Identification and Control of a Renewable Hydrogen Production System for Mobility Applications

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Laaksonlaita	Timo

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
Zurich, 30.01.2021	T. Laaksonlaita

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.