Control of Thermal Management Systems for Electric Vehicles

Energy efficiency optimization for the Lightyear 0

Jeroen van der Knaap

MSc Thesis

TU Delft Delft University of Technology

Delft Center for Systems and Control

Control of Thermal Management Systems for Electric Vehicles

Energy efficiency optimization for the Lightyear 0

Master of Science Thesis

Jeroen van der Knaap

June 13, 2023

Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS) · Delft University of Technology

Lightyear

The work in this thesis was supported by Lightyear, that is on its mission to deliver clean mobility for everyone, everywhere. Their cooperation is hereby gratefully acknowledged.

TU Delft Delft University of Technology

DCSC

Acknowledgements

The time working on my thesis was a special one.

First of all, I had an unforgettable time at Lightyear. Working with so many smart and motivated people and celebrating many company milestones, made it truly special. I would like to thank everyone at Lightyear who helped me for their support, specifically, my supervisor, Simon, and group lead, Chris for their time and help whenever I asked for it.

I would also like to express my gratitude towards my thesis defense committee members, particularly my supervisor at the TU Delft, Sander Wahls, whose guidance and feedback have been invaluable. Finally, my gratitude goes out to Andreas and Prof. Bock to invite me to visit the university in Heidelberg and learn from true experts in the field.

Of course, the work could not have been possible without my girlfriend, friends, and family. They have supported me throughout the entire time and many memories are made, that will be remembered forever.

Jeroen van der Knaap
June 13, 2023

Abstract

Improving the energy efficiency of electric vehicles has various significant benefits, such as increasing the driving range. The Thermal Management System (TMS) plays a large role in optimizing the vehicle energy consumption and battery lifetime. In this thesis, a nonlinear Model Predictive control (MPC) strategy is presented to regulate the battery, motor and inverter temperatures to minimize vehicle energy consumption and maximize battery lifetime, two conflicting objectives traded off by a single tunable parameter, whilst staying within temperature limits to ensure safety. The control dynamics are nonlinear and discontinuous, due to valves in the system that can only be fully opened or fully closed, leading to a nonlinear mixed-integer optimization problem. A nonlinear model of the TMS including actuators and electric components is formulated that is validated by using simulation results over three drive cycles for moderate and hot ambient conditions. Mean temperature deviations between -0.22 and 0.25 °C are achieved for the battery, and between 0.14 and 1.48 °C for the motors, depending on the drive cycle. Using outer convexification, the optimization problem is reformulated as a continuous problem, which can be solved efficiently. The control strategy is tested in a simulation environment for both moderate and hot ambient temperatures and three different drive cycles. The strategy is compared to a benchmark strategy that uses a finite-state machine and PID-based control loops. A decrease in power consumption between 7% and 11% is achieved for 5 out of 6 use cases whilst additionally decreasing the ageing rate by 0% to 7%. For one use case, an increase in energy consumption is achieved by 2.5%, but the relative ageing rate is decreased by 44%. At hot temperatures, improvements are mostly achieved due to finding an energy-efficient battery cooling trajectory. At moderate temperatures, improvements are mostly achieved by increased motor cooling to take advantage of the temperature-dependent motor losses. The results obtained using a continuous solver are also compared to those obtained using a mixed-integer solver. Minimal loss in performance is seen compared to the mixed-integer solver, whilst requiring a significantly lower computation time that is within the sample time, making the MPC strategy fast enough for real-time control in the simulation environment. Finally, the effect of using noisy forecast information is used to test the robustness of the controller. Using noisy forecast information instead of perfect forecast information, the energy consumption and ageing rate increase by 0.0% to 1.2%.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The energy transition is a hot topic in the current world. People are using more and more energy which often contributes to the emission of greenhouse gases [8] and therefore goes at the cost of climate change [9]. Currently, the transportation sector is one of the largest contributors to the emissions of greenhouse gases. Additionally, since 1990, global transport emissions have grown at a constant rate of about 2% per year [10]. To reduce environmental impact, in some countries, people are encouraged to take their bikes for short-distance transport, which is a simple and free solution. Also, the combustion vehicle is getting replaced by the Electric Vehicle (EV). Global increases in passenger and freight travel activity levels have outpaced energy efficiency and fuel economy improvements, continuing a long-term trend for the transport sector [10].

Engineers can make a high impact on making the transport sector more sustainable in many ways. New vehicles are being made that use renewable energy sources, recyclable materials can be used in the design and the available energy can be handled more efficiently within the car. One of the clearest changes in the most recent years is the increase in EV's on the road. Globally, the amount of EV's has grown with at least 30% per year since 2016 [11]. Certain factors are still limiting people from buying an EV. The high price, limited driving range and time to charge are important reasons for many people not to buy an EV yet [12]. By making vehicles as energy efficient as possible, both of these problems can be addressed.

The company Lightyear is creating a solar-powered EV, designed to be grid-independent and drive anywhere. One of the main design goals is to maximize vehicle efficiency: to be able to drive as far as possible while consuming as little energy as possible. By minimizing energy losses within the car, vehicle efficiency is optimized. Figure 1-1a and Figure 1-1b show the first model: Lightyear 0. To make the car as energy efficient as possible, the entire vehicle should be designed to allow that. This means that not just individual components should be optimized but interactions between components should be considered as well. For example, even though an electrical motor might be very efficient, there will be energy losses. By considering these losses, the vehicle can be designed such that these losses can be used again to heat up other parts in the system that need heating, such as the cabin. Using this waste heat can increase overall energy efficiency.

(a) Lightyear 0 is designed to minimize aerodynamic drag

(b) Lightyear 0 uses solar panels to generate energy

Figure 1-1: Lightyear 0 is designed with the main focus on energy-efficiency

Both the energy efficiency and the lifetime of many electrical components depend on their operating temperatures. Additionally, the Thermal Management System (TMS) systems in EVs can result in a range reduction of up to 50% [13]. Therefore improving the energy efficiency of the TMS is an active area of research. The TMS has two functions:

- Regulate the temperature of the electrical components to ensure component efficiency and durability.
- Provide passenger comfort by controlling the cabin air temperature, humidity, and CO_2 concentration, and ensuring proper defogging/defrosting of the windscreen.

Thermal conditioning of components can cost a lot of energy. Passive cooling is not always sufficient and active cooling, turning on actuators in the TMS, is required. On the other hand, conditioning components can save energy: For example, the efficiency and durability of the electrical components can be temperature-dependent [6, 14] and therefore also the component efficiency and lifetime can be improved by controlling the operating temperature. Here the trade-off arises between thermally conditioning components very precisely at their optimal operating temperature and saving energy in the TMS.

A separate TMS can be developed for the cabin, motors and batteries to increase ease of adaptability to different vehicle architectures [14]. To increase efficiency, durability and safety while driving, subsystems such as a separate battery thermal management system and the cabin Heating, Ventilation and Air Conditioning (HVAC) should be integrated [15]. In the integrated case, less hardware is required and waste heat recovery is possible, where excessive heat from the high-voltage electric components is reused to heat the cabin [16]. Many examples of hardware implementations for the integrated TMS exist [16, 17] but little research on control solutions has been done. For the Lightyear 0, a fully integrated thermal management system is developed where heat can be transferred from any location in the car to any other location in the car. This offers an opportunity for energy savings but requires more complex control engineering.

Another development is that modern vehicles can make use of the internet, GPS and detailed maps. Relevant data such as the road topology, trip duration, weather forecast and traffic conditions can be extracted for proactive control of a thermal conditioning system

[18]. For example, when it is known that a hill is coming up, cooling of the powertrain can be done in advance to prevent peaks in power consumption. Component temperatures can be precisely controlled to minimize energy consumption throughout the entire trip.

The complexity in such integrated systems has multiple causes, including but not limited to:

- Generally, as there are multiple heat sources/sinks (cabin, electric components, ambient), the same temperature requirements can be fulfilled by replacing the heat in different locations.
- In order to reach all possible objectives, different actuators are available. Due to this, different control strategies are possible to reach the same objective.

An optimal control strategy for the TMS attempts to find control actions that optimize the desired objectives.

1-1 Objectives & requirements

Various research has been done on thermal management in the last few years. Two developments can be seen:

- The TMS is becoming more integrated [19]. Integrating the TMS allows for heat transfer between the different vehicle components, such as the motors, inverters and battery. An integrated TMS requires less hardware required to fulfil the same requirements
- Using forecast information, such as ambient temperature and vehicle speed, disturbances can be predicted which can be used for energy-saving control of the TMS [4].

The aim of this thesis is to develop a control system that minimizes the energy consumption in the integrated thermal conditioning system of electric vehicles. It has been shown in research that predictive control allows for high energy savings compared to traditional controllers. In the thermal management in the building sector [20, 21, 22] and automotive sector [4, 7], or energy management in the automotive sector [23, 18], predictive control using forecast information has been applied. The combination of using forecast information in an integrated system that includes the electric motors, inverters and High Voltage Battery (HVB), and multiple actuator combinations, in the case of a system with discrete valves, creates a more complex control problem that has not been fully explored yet.

The objectives set for this thesis are to:

- Develop an optimal supervisory controller that is able to plan the optimal component temperature trajectory and actuator configuration over time and compare the performance to the currently existing system.
- Analyse the sensitivities with respect to predictable disturbances in the performance of the control system.

- Explore the possibilities for real-time control using the presented methodology.

It is assumed that the controlled vehicle follows a route that is given by the navigation system and that the vehicle is driven according to the speed limits. The following control system requirements are defined:

- The component temperatures must always stay within predefined limits.
- The control strategy must be computed while driving.

Chapter 2

Thermal management setting

In the TMS of an EV, there are multiple components that are thermally conditioned. The conditioning process is physically executed by several actuators in the TMS. This chapter describes a general hardware setup after which the background is given on the controller setup and the specific subproblem of cooling is described.

2-1 Generalized setup of the reference vehicle

Within the EV, the internal temperature of different components has different effects. For the High Voltage Battery (HVB), temperature affects lifetime and energy efficiency [6], and for the motors, temperature affects energy efficiency [14]. Within the considered TMS, heat can be moved from any heat sink to another. In this thesis, control of the circuits that are able to remove heat from the powertrain components, which are the motors, inverters, and HVB is considered. The temperatures can be controlled using any of the currently relevant various actuator configurations: Via the radiator or the chiller. Figure 2-1 shows the the basic setup of an electric vehicle TMS. Various blocks are connected to each other, whereas each block consists of hardware and can target different components.

2-1-1 Conditioned components

For an electric car, the motors, inverters, HVB and passenger cabin generally all need to be thermally conditioned. This section elaborates on the function of the conditioned components.

Drivetrain: In an electric vehicle, the drivetrain consists of motors and inverters. In an electric car, one or multiple electric motors are used to provide vehicle propulsion. The motors are powered by energy coming from the HVB. The inverter connects the HVB and electric motor(s) by converting the direct current (DC) available in the HVB to three-phase

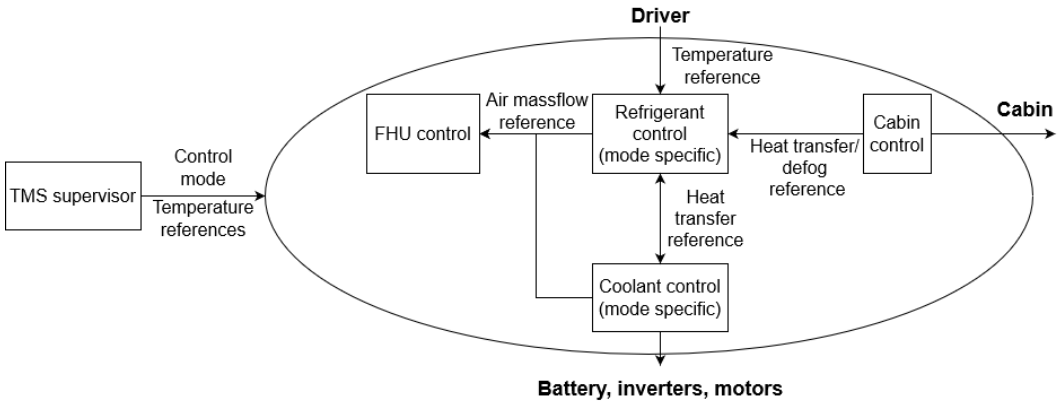


Figure 2-1: A top-level schematic of the TMS

alternating current (AC) which is required to drive the electric motors. The motors and inverters can be cooled by the coolant circuit.

Electric motors can generate a significant amount of heat which needs to be removed [3]. The temperature influences the efficiency, durability and safety of the motors [14, 24]. Next to the motors, the inverters generate significant heat that needs to be removed, whereas some of the losses are also temperature dependent [25].

In the reference vehicle, there are four motors and inverters. The components are cooled in series and parallel, as shown in Figure 2-2.

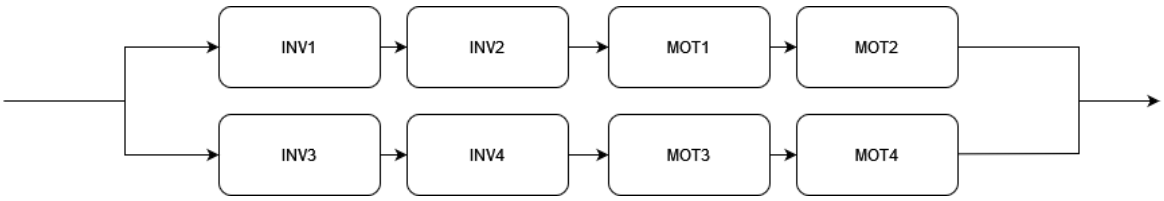


Figure 2-2: The inverter (INV) - motor (MOT) cooling arrangement.

For component safety and durability, temperature limits exist for the motors and inverters, which are shown in Table 2-1.

HVB: The High Voltage Battery (HVB) is the energy storage system for the motors, inverters and parts of the thermal management system that run on high voltage. Energy capacity, efficiency and lifetime are highly dependent on temperature. Battery lifetime is an important factor because of the price and environmental impact of replacing the battery. Additionally, safety must be ensured as extreme temperatures can cause self-ignition or thermal runaway [26]. For component safety and durability, temperature limits exist for the HVB, which are shown in Table 2-1. Next to temperature limits, there also charging and discharging power limits for the battery cells, depending on the temperature. The dependency on temperature for the power limits is shown in Figure 2-3. Between 20 and 35 °C charging and discharging are the least limited, whereas at extremer temperatures first only charging, but at < 10 °C and > 45 °C both charging and discharging are increasingly limited.

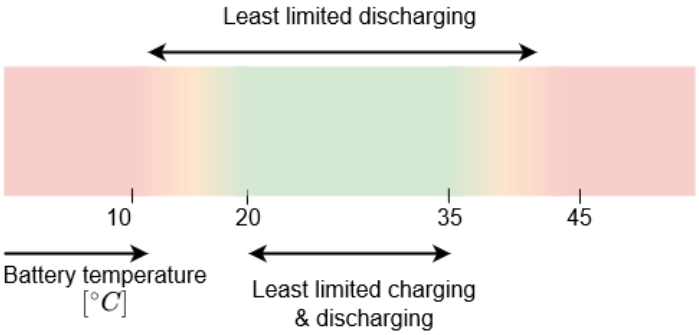


Figure 2-3: Battery power limits are related to temperature: In the green temperature zone, both charging and discharging are least limited, in the orange zone charging is increasingly limited and in the red zone both charging and discharging are highly limited

	Battery	**Inverters**	**Motors**	**Coolant fluid**
Lower temperature limit [°C]	0	0	0	0
Upper temperature limit [°C]	65	65	100	65

Table 2-1: Temperature limits for the different components in the vehicle

Passenger cabin: A Heating, Ventilation and Air Conditioning (HVAC) system has the function to optimise the level of thermal comfort for the passengers in the cabin. The passenger cabin is not considered in this thesis.

2-1-2 TMS hardware

As different actuator configurations can be used to reach the same objectives, understanding the limits and efficiencies of each configuration can provide useful information for further improvement of the control strategy. This section elaborates on the different hardware elements of the Thermal Management System (TMS).

Coolant system The coolant fluid used is a mixture of glycol and water. Coolant fluid is flowing along different parts of the vehicle through hoses. Within the coolant system, the mass flow and temperature of the coolant fluid are controlled. By control of pumps, heaters and heat exchangers, the heat transfer for and to the different components can be regulated. The coolant system is responsible for conditioning the motors, inverters and HVB. Heat can be exchanged with the refrigerant circuit through the chiller and the ambient air through the Front Hex Unit (FHU). For TMS safety and durability, temperature limits are set for the coolant fluid, as shown in Table 2-1.

Refrigerant system Due to the efficiency of cooling through vapour-compression cycle refrigeration, refrigerant is used in many electric vehicles. The refrigerant fluid is flowing past different parts of the vehicle through hoses. Within the refrigerant system, the mass flow and temperature of the refrigerant fluid are controlled. The refrigerant goes through a vapour-compression cycle, where heat transfer is controlled through the use of a compressor

and expansion valves. The refrigerant system is responsible for delivering the requested heat transfer to control the cabin temperature. Additionally, as mentioned previously, heat transfer can take place with the coolant system through the chiller and the ambient air in the FHU.

Front Hex Unit (FHU) The FHU provides the interface with the ambient air. The air flows through the grill and through the use of fans, the mass flow can be regulated. The FHU is responsible for providing the requested mass flow of ambient air for the refrigerant circuit and coolant circuit such that the desired heat transfer can take place.

Cabin HVAC system The Heating, Ventilation and Air Conditioning (HVAC) system is responsible for thermally conditioning the passenger cabin and is not considered in this thesis.

2-2 Simulation environment and reference controller

At Lightyear, a high-fidelity simulation environment has been developed in Simscape within the MATLAB Simulink environment. Simscape can be used to build and simulate physical component models based on physical connections. Components can be chosen from a large library or made custom, whereas many model parameters can be set based on which Simscape creates a differential-algebraic system of equations. The parameters within the simulation model are either based on component datasheets or verified by system identification experiments. The optimal control strategy developed in this thesis will be compared to an existing reference setup, where two control layers for the TMS exist. This is visualized in Figure 2-4. In the first layer, valve positions and target temperatures for different components are chosen using a finite-state machine, based on the component temperatures and ambient temperature. In the finite-state machine, a specific control mode needs to be chosen. There are six modes for cooling:

- Idle. All actuators are turned off.
- Battery cooling. The battery is cooled through the radiator for moderate cooling to ensure the battery temperature stays below 40 °C.
- Powertrain cooling. The motors and inverters are cooled through the radiator to ensure the component temperatures stay below their respective temperature limits, shown in Table 2-1.
- Battery chilling. The battery is cooled through the chiller for maximal cooling to ensure the battery temperature stays below 40 °C.
- Battery cooling + powertrain cooling. Both the battery, inverters and motors are cooled through the same radiator.
- Battery chilling + powertrain cooling. The battery is cooled through the chiller and the inverters and motors are cooled through the radiator. There is no significant interaction between the two circuits.

Then, a low-level PID-based controller tracks the set target temperatures using the chosen set of actuators.

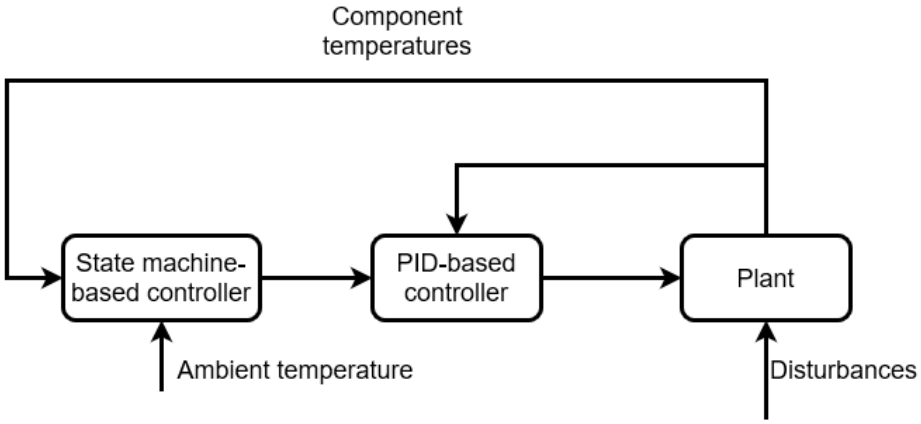


Figure 2-4: The benchmark control hierarchy in the TMS subsystem

Chapter 3

System modelling

This chapter describes the models used to describe the TMS. A simplified model of the cooling subsystem is described in Figure 3-1. First of all, black box, white box and grey box modelling methods are discussed. Background on the thermal, fluid, electrical and vehicle dynamics is given and models found in the literature are discussed. Finally, the models from the different domains are all integrated to create the state-space model.

3-1 White box, grey box and black box modelling

Three methods exist for modelling physical systems: Black box, grey box and white box modelling techniques. The advantages and disadvantages of all three methods are discussed.

Black box models: Black box models, sometimes called data-driven models, do not assume any physical correlation between modelled parameters. Examples of black box models are neural networks and random forests. Black box models have several advantages and disadvantages including:

- Low scalability with different hardware architectures.
- Low understandability because the model is not physics-based.
- High accuracy as black box models do not assume physical connections.
- High usability as an engineer does not need to understand the physics.

Black box models have been used for the thermal control of large buildings [20, 27]. As large buildings are often different (e.g. there are different materials and dimensions) and the influence of disturbances is large (e.g. there is a time-varying occupation) making individual physical models for each building can be time-consuming, inaccurate and expertise-demanding, data-driven approaches have a significant advantage.

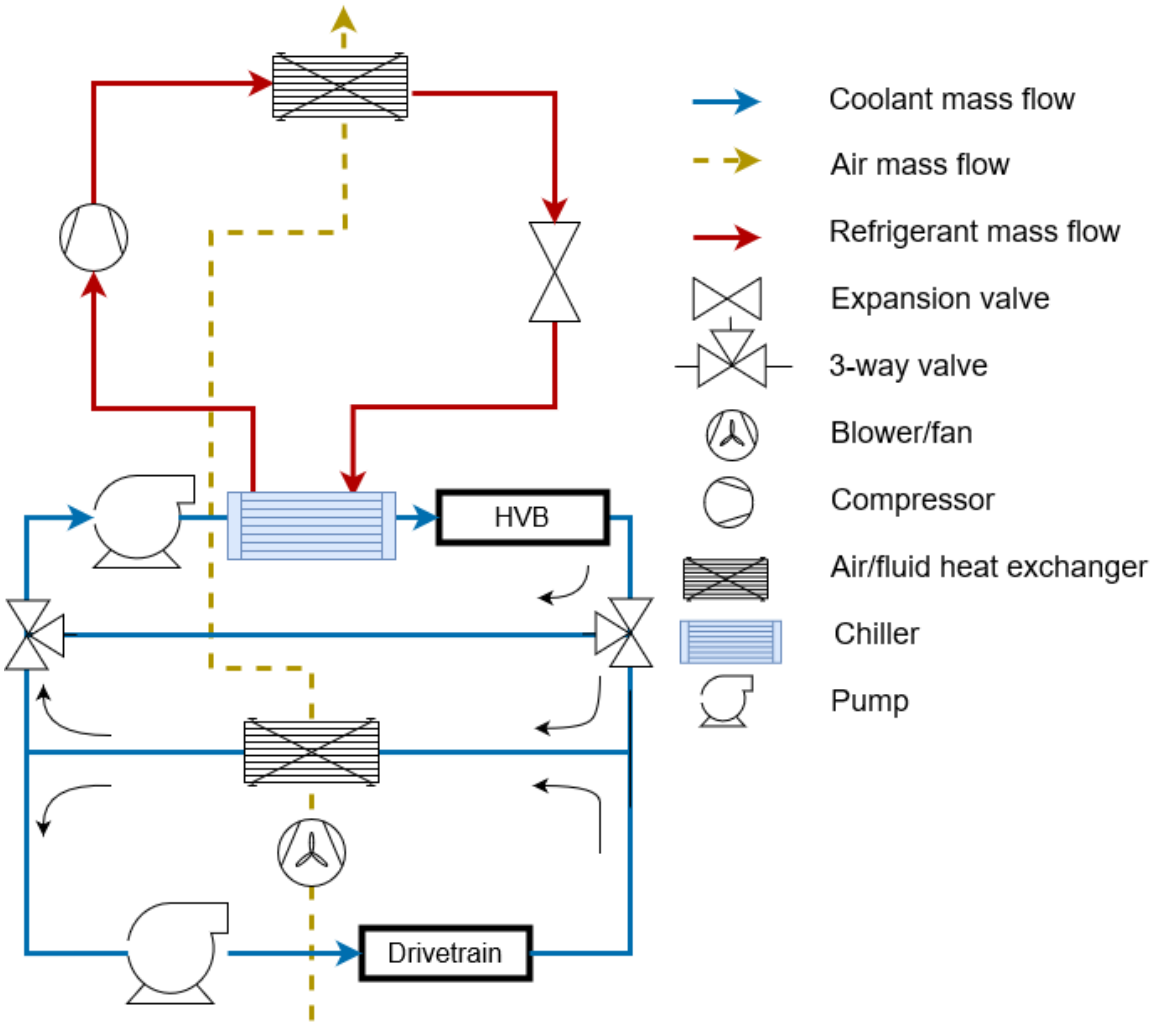


Figure 3-1: Model of the TMS cooling subsystem

White box models: White box models are based on physics. Differential equations have to be formulated that aim to describe the behaviour of the system over time and full knowledge about the system is assumed to be available to determine the model parameters. White box models have several advantages and disadvantages including:

- High scalability with different hardware architectures as components can be added or removed according to their physical connection to the system.
- High understandability because the model is physics-based.
- High accuracy is not assured as physical models may not always capture real-world behaviour.
- Low usability as an engineer does need to understand the physics.

Grey box models: Grey box models are based on physical equations but do use data to find model parameters. Differential equations have to be formulated that aim to describe the behaviour of the system over time and system identification tests are done to find model parameters. Grey box models have several advantages over white and black box models including:

- High scalability with different hardware architectures as components can be added or removed according to their physical connection to the system.
- High understandability because the model is physics-based.
- High accuracy as physical model parameters are fitted to data.
- Moderate usability as an engineer does need to understand the basic physical equations but the model parameters are fitted to data and not full system knowledge is required.

As the controller developed in this thesis is designed such that it can be used in many cars of the same model, the cost of engineering hours is relatively small. Additionally, as many different people may work with the controller, having easy-to-understand models is an important advantage. Finally, it is desired that the controller framework is scalable, and can be changed for different hardware configurations. Grey box models suit these requirements best.

3-2 Thermal dynamics

Thermal dynamics background is required to model the temperatures in the TMS and the heat transfer between different components. The first law of thermodynamics [28] states that energy cannot get lost, however, energy can be transformed from one form to another. Due to energy transformation, the internal energy of the system can change when energy is added or removed from the system. The internal energy is dependent on the temperature of the system which can change due to the addition or removal of heat, as expressed by

$$\Delta T = \frac{Q_{\text{added}} - Q_{\text{removed}}}{C}, \tag{3-1}$$

where T[K], Q_{added}[J], Q_{removed}[J] and C [J· K−1] are the temperature of the system, heat added to and removed from the system, and thermal capacity of the system. The following section applies the laws from thermal dynamics [28] to model the temperature dynamics of the electric components and coolant fluid, as well as the various heat exchangers.

3-2-1 Electrical component temperature

The temperature dynamics for a component with mass m and heat capacity C can be described by

$$\frac{dT_{\text{component}}}{dt} = \frac{\dot{Q}_{\text{losses}} - \dot{Q}_{\text{dissipated}}}{C}, \tag{3-2}$$

where $\dot{Q}_{\text{losses}}$ [W] is the rate of generated heat due to losses and $\dot{Q}_{\text{dissipated}}$[W] is the rate of heat dissipation through cooling, which can be to the coolant or ambient air. This concept is visualized by Figure 3-2. The heat capacity C [J·K^{-1}] is defined by

$$C = mc, \tag{3-3}$$

where m[kg] is the component mass and c [J·K^{-1}·kg^{-1}] is the specific heat capacity of the component.

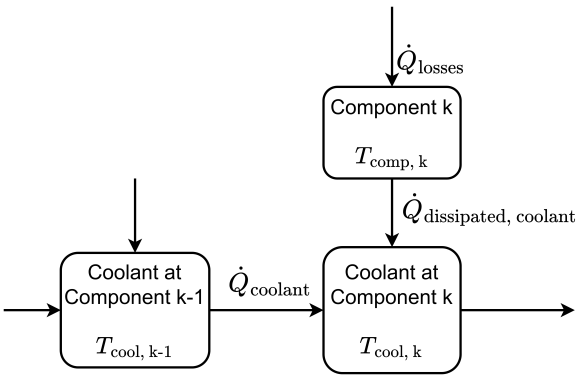


Figure 3-2: Component and coolant temperature dynamics

When a component is being cooled down using a cooling plate mounted to the component, practically, the component temperature will be different at different distances from the cooling plate. The component can be modelled using a multi-node model, where the temperature of different sub-components is modelled or a single-node model, where the component temperature is represented by a single element. In the following paragraphs, the application of multi-node and single-node models for the High Voltage Battery (HVB), motors and inverters are discussed.

HVB

As the battery comprises many cells, the battery parameters, such as temperature and voltage, can be determined for each cell. In previous research, it has been shown that it can be assumed that the temperature can be treated as a homogeneous and uniform solid, meaning that the electrical and thermal parameters of the cells inside the HVB are the same. This is valid as the maximum temperature difference between cells in the battery pack is only small [6]. Such a lumped thermal mass model was used in [29, 30].

Motors and inverters

In [3], a motor thermal controller is developed that uses a two-node model and models both the rotor and stator temperature separately. The controller is designed with the aim to validate the novel hardware architecture and is not compared to a benchmark controller. As with the battery, lumped parameters models with uniform temperature can be used to model the electric motor [14]. A Model Predictive control (MPC)-based control strategy is

developed where up to 86 % energy saving is achieved compared to the benchmark gain-scheduling controller. In the same paper, the dynamics of the power electronics are neglected in the control model, as the dominating thermal inertia is given by the electric motors. This means that they assume that the power electronics directly transfer heat to the coolant. This allows for model reduction while their simulations reveal only small deviations. This however might not always be possible, as the inverters have a maximum operating temperature. If the temperature is not tracked, this might cause the inverter to overheat.

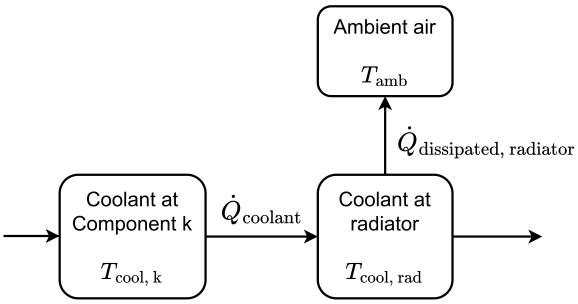


Figure 3-3: Coolant temperature dynamics at the radiator

3-2-2 Coolant temperature

The coolant in the vehicle is flowing through tubes and is heated up or cooled down by the electrical components, refrigerant or ambient air. In between these components, the temperature change due to losses in the tubes is assumed negligible.

The dynamics of the coolant outlet temperature at a component, as visualized in Figure 3-2, T_{coolant}[K] can be described as

$$\frac{dT_{\text{coolant}}}{dt} = \frac{\dot{Q}_{\text{coolant}} + \dot{Q}_{\text{dissipated}}}{mc}, \tag{3-4}$$

where $\dot{Q}_{\text{coolant}}$[W] is defined using the heat balance equation as

$$\dot{Q}_{\text{coolant}} = (T_{\text{coolant, inlet}} - T_{\text{coolant, outlet}})\dot{m}c \tag{3-5}$$

and depends on the inlet temperature $T_{\text{coolant, inlet}}$ [K], the outlet temperature $T_{\text{coolant, outlet}}$ [K] of the fluid and the mass flow rate of the fluid $\dot{m}$ [kg·s^{-1}]. When a valve determines the inlet temperature of a component, $\dot{Q}_{\text{coolant}}$ is described by

$$\begin{aligned}\dot{Q}_{\text{coolant}} =& w_1(T_{\text{coolant, inlet, 1}} - T_{\text{coolant, outlet}})\dot{m}c \quad + \\ & w_2(T_{\text{coolant, inlet, 2}} - T_{\text{coolant, outlet}})\dot{m}c,\end{aligned} \tag{3-6}$$

where $T_{\text{in, 1}}$ [K], $T_{\text{in, 2}}$ [K] are the two inlet temperatures.

When the heat is being exchanged with another fluid, for example, at the radiator as visualized in Figure 3-3, the coolant temperature change is described by

$$\frac{dT_{\text{coolant}}}{dt} = \frac{\dot{Q}_{\text{coolant}} - \dot{Q}_{\text{dissipated}}}{mc}. \tag{3-7}$$

In both cases, $\dot{Q}_{\text{dissipated}}$ can be described by modelling the specific heat exchanger, as described in subsection 3-2-3.

The temperature at the outlet of a junction $T_{\text{coolant, out}}$ [K] can be described by

$$T_{\text{coolant, out}} = \frac{T_{\text{in, 1}}\dot{m}_1 + T_{\text{in, 2}}\dot{m}_2}{\dot{m}_1 + \dot{m}_2}, \tag{3-8}$$

where $T_{\text{in, 1}}$ and $T_{\text{in, 2}}$ are the temperatures of the entering mass flows $\dot{m}_1$ [kg $\cdot$ s^{-1}] and $\dot{m}_2$ [kg $\cdot$ s^{-1}] respectively.

3-2-3 Heat exchangers

In vehicle thermal management, components are cooled using passive cooling, active cooling or both. In active cooling, actuators such as pumps and fans are used to accelerate the mass flow rate of the fluid, which can be the coolant or ambient air. The heat transfer $\dot{Q}$ [W] is described by

$$\dot{Q} = UA(T_1 - T_2) \tag{3-9}$$

and depends on the temperatures T_1[K] and T_2[K], for example, the coolant temperature and ambient air temperature in the radiator. U[W$\cdot$m$^{-2}\cdot$K^{-1}] is the overall heat exchange coefficient and A[m^2] is the heat exchange surface. This equation cannot be directly applied as the temperature difference at the inlet and outlet of a heat exchanger is generally not the same when a mass flow rate is used to cool down a component or another fluid, as visualized in Figure 3-4.

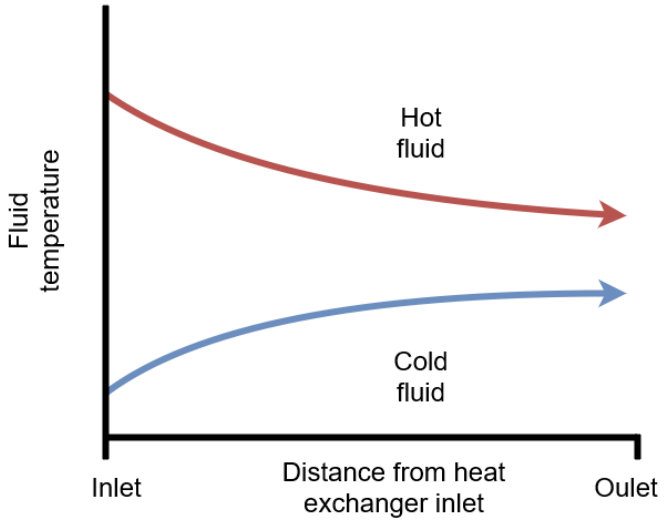


Figure 3-4: Fluid temperatures in a heat exchanger generally change between the inlet to outlet

A method to estimate the heat exchange is the Log-Mean Temperature Difference (LMTD) method, which uses the measured inlet and outlet fluid temperature to estimate the average temperature difference between the hot and the cold fluid. This can be used for predicting performance in thermal management, as was done in [31]. In a real car, we do not have access to all the inlet and outlet temperature measurements but we would still like to

estimate the heat transfer over time. The Number of Transfer Units (NTU) method can be used for this purpose as the method only requires the component and coolant inlet temperature, or two inlet temperatures in the case of the radiator. This method was also used for motor cooling [14] and battery cooling [6] through a coolant.

Number of Transfer Units (NTU) method

The heat transfer between the fluid mass flow and a component, or between different fluid mass flows can be modelled using the NTU method, also called the effectiveness method. Using this method, the heat transfer is described by

$$\dot{Q} = \epsilon C_{\mathrm{r}}(T_1 - T_2), \tag{3-10}$$

using the unitless heat transfer efficiency coefficient $\epsilon[\ ]$, the temperatures at each side of the heat exchanger T_1 and T_2 and the minimum heat capacity rate $C_{\mathrm{r}}[\mathrm{W\ k^{-1}}]$, which is described by

$$C_{\mathrm{r}} = \min(c_1\dot{m}_1, c_2\dot{m}_2). \tag{3-11}$$

ϵ is a function of the unitless NTU$[\ ]$, which is defined as

$$\mathrm{NTU} = \frac{UA}{C_{\mathrm{r}}}. \tag{3-12}$$

When considering the cooling of the electrical components, a single-stream heat exchanger is used, as the component with an assumed uniform temperature, is cooled by a single stream of air and/or coolant fluid. The minimum heat capacity rate is based on the coolant mass flow rate and component thermal capacity and expressed by

$$C_{\mathrm{r}} = c_{\mathrm{coolant}}\dot{m}_1. \tag{3-13}$$

Now, ϵ is described as

$$\epsilon(\dot{m}_{\mathrm{cool}}) = 1 - \exp\left(-\mathrm{NTU}\right). \tag{3-14}$$

As this method is valid for the case of a single-stream heat exchanger, this method can be used when a mass flow is flowing past a component or when one of the two fluids is held at a fixed temperature, for example during evaporation or condensation.

For a double-stream heat exchanger, with two different fluids flowing past each other, the single-stream method cannot be used. The radiator, which allows cooling for the coolant through the ambient air, has a cross-flow configuration. For this case, ϵ is described by [32]

$$\epsilon(\dot{m}_{\mathrm{cool}}) = 1 - \exp\frac{\exp -C_{\mathrm{r}}\mathrm{NTU}^{0.78} - 1}{C_{\mathrm{r}}\mathrm{NTU}^{-0.22}}. \tag{3-15}$$

When several components are cooled in series, the heat dissipation can be calculated for each component based on the component's temperatures and the initial coolant temperature only. The coolant temperature at the outlet of the first component in a steady state is calculated by

$$T_{\mathrm{coolant,\ out,\ 1}} = T_{\mathrm{coolant,\ in,\ 1}} + \epsilon(T_{\mathrm{comp}} - T_{\mathrm{coolant,\ in,\ 1}}). \tag{3-16}$$

By substitution, the outlet of the coolant at the following components can also be calculated, when knowing the series component temperatures and inlet temperature of the first component only.

Heat transfer coefficient

When convective cooling happens due to a mass flow rate, the UA depends on this mass flow rate and can be calculated using empirical Nusselt correlations [1, 33]

$$U \propto \mathrm{Nu} = \frac{\dot{Q}_{\mathrm{conduction}}}{\dot{Q}_{\mathrm{convection}}}, \tag{3-17}$$

where the Nusselt number Nu[ ] is a non-dimensional number defined as the ratio of conductive heat transfer to convective heat transfer and is related to the flow rate via Reynolds number by

$$\mathrm{Nu} \propto Re^{b}, \tag{3-18}$$

where b has a value between 0.5 and 0.8, depending on whether the flow is laminar or turbulent. Reynolds number is defined as

$$Re = \frac{\rho Q D_H}{\mu A}, \tag{3-19}$$

where ρ is the fluid density, Q is the volumetric flow rate $[m^3 \cdot \mathrm{s}^{-1}]$, D_H is the hydraulic diameter [m], μ is the dynamic viscosity $[Pa \cdot s]$ and A is the pipe cross-sectional area, such that the UA can be described by

$$UA = a\dot{m}^{b}_{\mathrm{fluid}}\ , \tag{3-20}$$

where a and b have to be determined experimentally. When a thermal mass is present within the heat exchanger, heat transfer might be different. In practice, the temperature of the component is connected to the fluid through for example a coolant plate, and the targeted component is not directly cooled by the coolant.

For the cross-flow heat exchanger, both air and coolant mass flow rates influence the heat exchange. The thermal resistances are in series, such that the heat transfer coefficients are added by

$$UA = \frac{1}{\frac{1}{a_1 \dot{m}^{b_1}_{\mathrm{fluid},1}} + \frac{1}{a_2 \dot{m}^{b_2}_{\mathrm{fluid},\ 2}}}, \tag{3-21}$$

where a_1 and b_1 are fitted parameters for the heat transfer due to the mass flow rate of the first fluid and a_2 and b_2 are fitted parameters for the heat transfer due to the mass flow rate of the second fluid.

3-3 Fluid dynamics

Different fluids are used within the TMS. Fluid parameters such as the mass flow rate, temperature and specific heat capacity have an effect on thermal dynamics. The following section presents the relevant background on fluid dynamics from [28] to model the TMS. The mass flow rate of a fluid is related to the volumetric flow rate $\dot{V}_{\mathrm{fluid}}[\mathrm{m}^3 \cdot \mathrm{s}^{-1}]$ by

$$\dot{m}_{\mathrm{fluid}} = \dot{V}_{\mathrm{fluid}} \rho_{\mathrm{fluid}}, \tag{3-22}$$

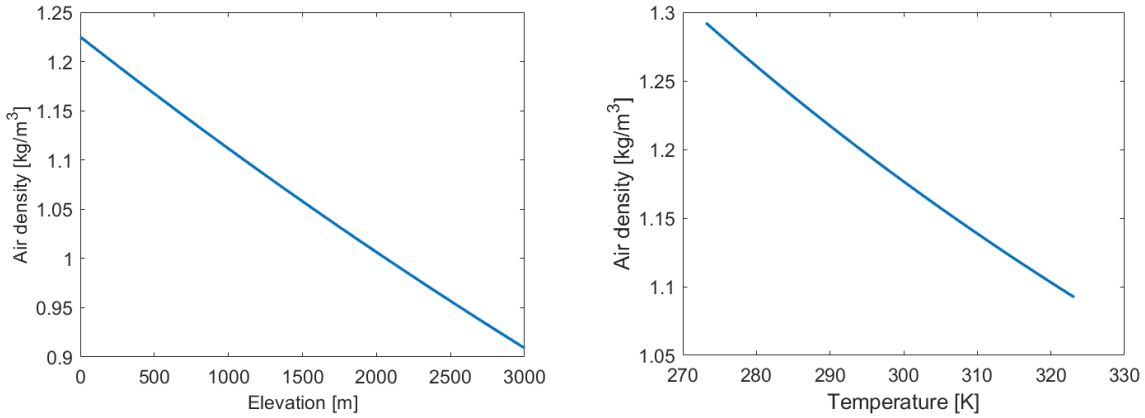
where $\rho[\text{kg}\cdot\text{m}^{-3}]$ is the density of the fluid. According to the affinity laws, the volumetric flow rate is proportional to the fan speed $n_{\text{fan}}[\text{s}^{-1}]$ or pump speed $n_{\text{pump}}[\text{s}^{-1}]$:

$$\frac{n_1}{n_2} = \frac{\dot{V}_1}{\dot{V}_2}. \tag{3-23}$$

The density of gases ρ is described by

$$\rho = \frac{p}{R_{\text{gas}}T}, \tag{3-24}$$

where $p[\text{Pa}]$ is the absolute pressure, $T[\text{K}]$ is the gas temperature and $R_{\text{gas}}[\text{J}\cdot\text{K}^{-1}\cdot\text{mol}^{-1}]$ is specific gas constant. The air density therefore also depends on the elevation via the absolute pressure. The infleunce of elevation and temperature on the air density is visualized in Figure 3-5.



(a) Absolute pressure at different heights **(b)** Air density at different temperatures

Figure 3-5: Effect of height and temperature on air density

3-3-1 Ambient air

The volumetric flow rate of the ambient air $\dot{V}_{\text{amb}}[m^3\cdot\text{s}^{-1}]$ is a function of the speed of the car and the fan speed.

$$\dot{V}_{\text{amb}} = f(v_{\text{car}}, n_{\text{fan}}), \tag{3-25}$$

where $v_{\text{car}}[\text{m}\cdot\text{s}^{-1}]$ is the vehicle speed and $n_{\text{fan}}[\text{s}^{-1}]$ is the fan speed. The ambient volumetric flow rate is defined as [5]

$$\dot{V}_{\text{amb}} = \alpha v_{\text{car}} + \dot{V}_{\text{added,fan}}, \tag{3-26}$$

where the flow rates caused by the vehicle speed through the parameter α and fan speed $\dot{V}_{\text{added,fan}}[m^3\cdot\text{s}^{-1}]$ are added.

3-3-2 Coolant

The volumetric flow rate of the coolant fluid is a function of the pump speeds and pressure drops caused by the components in the circuit. The pressure drop depends on the current flow configuration through the valve positions w [1]. The volumetric flow rate around a pump is then expressed by

$$\dot{V}_{\text{coolant, pump}} = f(w, n_{\text{pump}}). \tag{3-27}$$

Additionally, the volumetric flow rate at the outlet of a valve w with two inputs is defined as follows

$$\dot{V}_{\text{coolant, valve}} = w\dot{V}_{\text{in, 1}} + (1 - w)\dot{V}_{\text{in, 2}}, \tag{3-28}$$

where w represents the valve position, being 1 when open and 0 when closed. Depending on the accuracy required, the affinity laws can also be used for the coolant pump, which will be validated using system identification.

3-3-3 Refrigerant

In the vapour compression cycle, the refrigerant undergoes phase changes when exchanging heat with the coolant and ambient. In a phase change, the temperature does not change but the enthalpy, or heat energy, does. The enthalpy H[J] is described by

$$H = E + pV, \tag{3-29}$$

where E[J] is the refrigerant internal energy, p[Pa] is the pressure and V[m^3] is the volume. The mass flow rate and pressures are controlled through a compressor and one or multiple expansion valves. For a single expansion valve, the pressure-enthalpy diagram might look as shown in Figure 3-6. The vapour compression cycle comprises four phases, as visually explained in Figure 3-6.

1. Compression. Work is added to the system by the compressor. The internal energy is increased, as well as the pressure and temperature. The refrigerant is still vapour.
2. Condensation. Heat is transferred from the system to the ambient, and the temperature and pressure are kept constant. The refrigerant goes through a phase change and is now in a fluid state.
3. Expansion. The refrigerant is expanded through the expansion valve. The pressure and temperature are decreased. The refrigerant is now partly vaporized.
4. Evaporation. Heat is removed from the to-be-cooled component and energy is added to the system. The refrigerant goes through a phase change and is now vapour.

The refrigerant circuit is controlled by two inputs, the compressor and the expansion valve. When cooling by the refrigerant circuit is required, the goal is to go through the vapour compression cycle as efficiently as possible, meaning that only the minimum amount of work is added to achieve the desired cooling rate.

To model the refrigerant circuit, many additional states need to be modelled, to account for each of the four phases. Such an approach was presented in [34]. This also requires

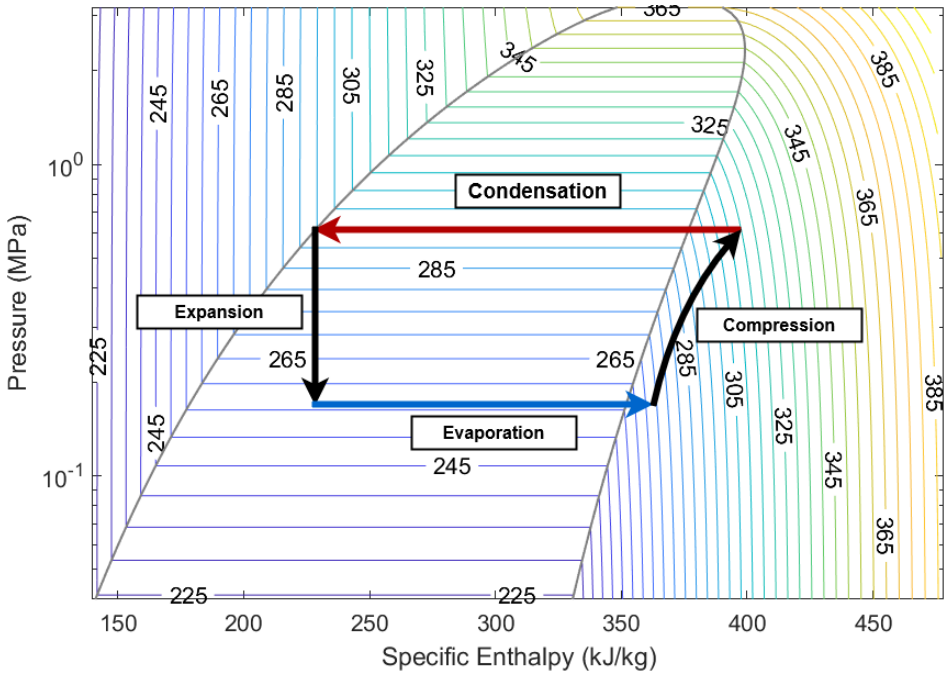


Figure 3-6: A possible layout of a vapour compression cycle

tabulated data to model both the relationship between pressure and internal energies. The compressor and expansion valve need to be controlled at a high frequency to maintain efficiency and stability.

When it is assumed that the temperature and pressure dynamics in the refrigerant are faster than the target heating/cooling dynamics, static mapping allows for more convenient modelling [35].

The compressor is the main power consumer in the refrigerant system. The power consumption of the compressor is often modelled using the unitless Coefficient Of Performance (COP) [5, 36, 37]. COP is defined by

$$\mathrm{COP} = \frac{\dot{Q}_{\mathrm{dissipated}}}{P_{\mathrm{compressor}}}. \tag{3-30}$$

In some papers [36], a fixed COP is assumed, but as shown in many other papers [5, 37], this assumption is not valid in many circumstances, as the COP can vary significantly.

Part of this variation depends on the dimensionless Partial Load Ratio (PLR) [37], which is defined as

$$\mathrm{PLR} = \frac{\dot{Q}_{\mathrm{desired}}}{\dot{Q}_{\mathrm{nominal}}}, \tag{3-31}$$

where $\dot{Q}_{\mathrm{desired}}$[W] is the heat dissipation setpoint and $\dot{Q}_{\mathrm{nominal}}$[W] is the nominal heat dissipation of the system. The PLR should be taken into account as the COP is not equal under different desired cooling capacities.

A COP static map was used in [5] for Heating, Ventilation and Air Conditioning (HVAC) control and COP was modelled as a fixed value in an MPC strategy in[36, 38], different for

heating and for cooling [39]. For MPC, a static map can be used, for example, taking into account the dependency on ambient temperature, cabin temperature and partial load ratio [37]. As mentioned in [40], weighting matrices in MPC can be adjusted according to current operating conditions, such as the COP, at the start of a new function evaluation. This allows for efficient optimization while taking into account the current operating conditions.

In the integrated TMS, the compressor can be used to condition more than one component, for example, the battery and the cabin. In this case, multiple evaporators are used and a COP will exist for the different cooling demands [41].

3-4 Electrical dynamics

First, the models regarding the power consumption of TMS actuators are described, after which models for the losses from the battery, motor and inverters are described. These losses depend on the current torque and speed requirements, which are modelled in the final part of this section.

3-4-1 Compressor

As mentioned in the subsection 3-3-3, the power consumption of the compressor can be modelled through the COP. The parameter identification of the COP is discussed in chapter 6.

3-4-2 Pumps and fans

To minimize power consumption in the vehicle, we need to know the energy required to reach the desired mass flow rate to control heat exchange. According to the affinity laws [28], the power consumption of the fan or pump is related to the volumetric flow rate as follows:

$$\frac{P_1}{P_2} = (\frac{\dot{V}_1}{\dot{V}_2})^3, \tag{3-32}$$

showing a cubic relationship between the volumetric flow rate $\dot{V}[\mathrm{m^3 \cdot s^{-1}}]$ and the power consumption $P[\mathrm{W}]$. This law does assume a fixed electric efficiency.

The pump power consumption $P_{\mathrm{pump}}[\mathrm{W}]$ is then described as

$$P_{\mathrm{pump}} = a\dot{V}^3_{\mathrm{coolant}} \tag{3-33}$$

where $\dot{V}_{\mathrm{coolant}}[\mathrm{m^3 \cdot s^{-1}}]$ is the volumetric flow rate and fitting parameter a has to be found through experiments and can differ per valve position [1]. Additional second and first-order terms [6] can be used to achieve the desired model accuracy, such that

$$P_{\mathrm{pump}} = a_1\dot{V}^3_{\mathrm{coolant}} + a_2\dot{V}^2_{\mathrm{coolant}} + a_3\dot{V}_{\mathrm{coolant}}. \tag{3-34}$$

If multiple pumps are used within the same circuit, there is interaction and separate models per pump as in Equation 3-34 may not be accurate. In [1], the relation was captured by Look-Up Table (LUT)'s. In this thesis, the following relation is assumed:

$$P_{\mathrm{pump},1} + P_{\mathrm{pump},2} = f(\dot{V}_1) + f(\dot{V}_2) + b\dot{V}_1\dot{V}_2, \tag{3-35}$$

where $f(\dot{V}_1)$ and $f(\dot{V}_2)$ are described by Equation 3-34 and parameter b needs to be found through experiments. For the fan, it should be taken into account that there is an external force, due to the vehicle speed, adding mass flow to the system. The power consumption of the fan P_{fan}[W], based on Equation 6-11, is expressed by

$$P_{\text{fan}} = a_1 \dot{V}_{\text{air}}^3 - a_2(\alpha v_{\text{car}})^3 \tag{3-36}$$

where $\dot{V}_{\text{air}}[\text{m}^3 \cdot \text{s}^{-1}]$ is the desired flow rate and $v_{\text{car}}[\text{m} \cdot \text{s}^{-1}]$ is the vehicle speed.

3-4-3 HVB

A general model to calculate the battery losses of Lithium-ion cells $\dot{Q}_{\text{losses}}$ comprises two main terms [29] such that the losses are expressed by

$$\dot{Q}_{\text{losses, battery}} = I_{\text{battery}}^2 R_{\text{battery}} - I_{\text{battery}} T_{\text{battery}} \frac{dV_{\text{oc}}}{dt}, \tag{3-37}$$

where the first term comprises losses due to the internal resistance $R_{\text{battery}}[\Omega]$ and battery current I_{battery}[A], and the second term is due to entropy change. The entropy change can be ignored as the internal resistance losses are dominant, it comprises more than 95% of the total losses in [42]. The losses are then described by

$$\dot{Q}_{\text{losses, battery}} = I_{\text{battery}}^2 R_{\text{battery}}. \tag{3-38}$$

Different models for the cell internal resistance $R_{\text{cell}}[\Omega]$ exist. In previous work, the resistance is modelled as a function of cell temperature and State of Charge (SoC) [42, 6, 30, 1]. The values for the cell resistance can be obtained through offline tests and stored for lookup [42, 43], such that the cell resistance can be described as

$$R_{\text{cell}} = f(T_{\text{cell}}, \text{SoC}), \tag{3-39}$$

where T_{cell}[K] is the cell resistance. The SoC is generally kept above 20% to maintain normal operation. In the region where SoC $> 20\%$, the cell resistance is almost solely dependent on the temperature [30]. This allows for significant model simplification. The relation between the cell resistance with temperature can be determined offline and stored for lookup [42] or a linear relation can be drawn in the region above 25 °C [30], such that

$$R_{\text{cell}} = a - bT_{\text{battery}}, \tag{3-40}$$

where a and b need to be identified through system identification experiments. When it is assumed that each cell in the battery has the same temperature, the losses are expressed as

$$\dot{Q}_{\text{losses, battery}} = I_{\text{battery}}^2 R_{\text{battery}}(T_{\text{battery}}). \tag{3-41}$$

Ageing

Estimating the lifetime of Li-ion batteries is a critical challenge that limits the integration of EV's into the automotive market [44]. It is also necessary because the degradation of the batteries largely determines the cost, performance and environmental impact of EV's [45].

The topic of ageing is also relevant for thermal management, as the temperature of the battery has an influence on the battery State of Health (SoH), or lifespan [6]. The term ageing comprises different effects, namely capacity fade and power fade, caused by the increase of internal resistance. Capacity fade $C_{\text{lost}}[\%]$is defined as the loss in battery capacity, or available energy, over time and is described by

$$C_{\text{lost}} = 100 - \frac{C_{\text{used}}}{C_{\text{original}}} \times 100\%, \tag{3-42}$$

depending on the battery capacity after usage and the orignal capacity $C_{\text{used}}[\text{Wh}]$ and $C_{\text{original}}[\text{Wh}]$. Power fade is caused by the increase of the battery resistance $R_{\text{battery}}[\Omega]$ and will cause decreased efficiency over time. The variables influencing ageing can be different for batteries with different designs and chemistries [46].

If a battery cooling strategy would be based only on minimizing losses, and not ageing, cooling would never be required except when reaching the upper-temperature limit. This would not be optimal for ageing. Tests required to identify ageing models often take a long time, for example, a dataset involving many cells, which were tested for more than three years, was used in [47]. Accelerated ageing tests have been developed [48], but due to the length of this thesis, ageing model identification is not feasible within the available timeline and model parameters are chosen based on literature.

During the battery lifetime, the performance degrades because of various ageing mechanisms. This problem of ageing can be divided into two subproblems: calendar ageing and cyclic ageing. Calendar ageing occurs independently from charging or discharging. Cycling ageing occurs when the battery is charged or discharged.

Calendar ageing The main calendar ageing mechanism, growth of the Solid Electrolyte Interface (SEI) layer, is accelerated by high temperatures and SoC [47, 48, 45]. Most electric vehicles spent more than 90% of their time being parked and current rates are relatively low compared to hybrid electric vehicles, except when accelerating or fast charging. Therefore, it is recognised that the main ageing mechanism is the SEI growth and calendar ageing models are able to make reliable lifetime predictions [45]. Calendar ageing contributes more to increased internal resistance than cycling according to [49]. The temperature-ageing relation can be described by the Arrhenius equation [47, 48, 46, 49]. This equation describes the rate of chemical reaction rates by

$$\text{reaction rate} \propto \exp\left(-\frac{E_a}{k_b T}\right). \tag{3-43}$$

Based on the activation energy $E_a[\text{J}]$, temperature $T[\text{K}]$ and Boltzmann constant $k_b[\text{J}\cdot\text{K}^{-1}]$, such that a temperature-dependent ageing factor α can be expressed as

$$\alpha = a \exp(-\frac{b}{T}), \tag{3-44}$$

where a and b are fitted to measurement data and α can be expressed differently for capacitive and resistive ageing. Assuming a time dependency of $t^{0.75}$ [48], The remaining capacity after ageing $C_{\text{aged}}[\%]$ can be then described by

$$C_{\text{aged}} = (1 - \alpha_{\text{cap}} t^{0.75}) \cdot 100\%, \tag{3-45}$$

using a fitted capacity ageing factor α_{cap}, which is expressed by Equation 3-44, and the time t, here in days. This same can be done for battery cell resistance after ageing $R_{\text{aged}}[\%]$, which can be expressed as

$$R_{\text{aged}} = (1 + \alpha_{\text{res}} t^{0.75}) \cdot 100\%, \tag{3-46}$$

where the resistance ageing factor α_{res} is also expressed by Equation 3-44 and fitted to resistance measurement data.

Cycling ageing The main cycling ageing mechanism, lithium plating, is increased at high charge rates and at low temperatures [45]. Similarly, with calendar ageing, cycling ageing has been modelled using the Arrhenius equation [50] [46].

Several approaches to modelling ageing for control are possible. Ageing models have been used for battery thermal control [6]. The used method [50] does include discharge rates and temperature, but no charge rates. Here they assume that high temperatures increase the rate of ageing during discharging, not taking into account the increased cycling ageing at low temperatures. In [7], a desirable range is determined offline. They mention that the overall ageing is slowest and the internal resistance is minimal in a temperature range from approximately 15 to 40 °C, based on [46]. A practical temperature reference is then chosen with consideration of ambient temperature. In [42, 1], curve fitting was done to assign a cost to both higher and lower battery temperatures compared to their minimal optimal temperature of 27 °C. In [29], simply a fixed desired temperature was defined and deviation was penalized in the MPC cost function.

The results of ageing tests in [46] show that ageing happens at a minimal rate at 25°C, as at both lower and higher temperatures, different ageing mechanisms come up and can be modelled using the Arrhenius equation. Using this method, both effects happening at low and high temperatures are separately modelled. Therefore, the model is scalable and can be adjusted easily when ageing tests are done.

3-4-4 Motors and inverters

Electric motors can generate a significant amount of heat which needs to be removed [3]. The temperature influences the efficiency, durability and safety of the motors [14, 24].

Next to the motors, the inverters generate significant heat whereas the losses are temperature dependent [25]. The inverter switching losses as well as conduction losses are most significant at high temperatures. In [14], however, the inverter temperatures are modelled as non-temperature-dependent. The reason might be that the losses depend very much on the specific vehicle architecture and inverter design. The motor losses can be divided into copper losses, iron losses, magnet losses and mechanical losses, whereas the copper losses $\dot{Q}_{\text{losses, copper}}[\text{W}]$ are most significant at high motor torques, as the currents through the copper windings, and the losses, increase with respect to the phase resistance, as expressed by

$$\dot{Q}_{\text{losses, copper}} = I^2 R_0 (1 + \alpha (T_{\text{motor}} - T_0)), \tag{3-47}$$

where $I[\text{A}]$ is the motor winding current, $R_0[\Omega]$ is a reference resistance measured at reference temperature $T_0[\text{K}]$ and $T_{\text{motor}}[\text{K}]$ is the current temperature and $\alpha[\text{K}^{-1}]$ is the

temperature coefficient. This shows that the copper losses are proportional to temperature. The iron and magnet losses are caused by the frequent change of magnetic flux density, inducing a voltage in the stator, rotor and magnets, leading to eddy currents. The remaining mechanical losses are caused by the bearings and depend mainly on the rotational speed of the motor [25]. Due to the complex thermal condition of the motor, the mathematical modelling of the losses is complicated. The iron and magnet losses are determined by the rotor speed ω and electric current and can be stored in a lookup table whose inputs are the rotor speed and current amplitude [24]. Combining the map and knowledge on copper resistance, one can calculate the motor losses $\dot{Q}_{\text{losses, motor}}$[W] using [14]

$$\dot{Q}_{\text{losses, motor}} = f(\omega, \tau) + a\tau^2(1 + \alpha(T_{\text{motor}} - T_0)), \tag{3-48}$$

where ω[rad·s^{-1}] is the motor speed, τ[N·m] is the torque setpoint, proportional to the winding current and a needs to be identified through experiments. As no inverter models were available that include temperature, the inverter losses are assumed non-temperature-dependent and are expressed by

$$\dot{Q}_{\text{losses, inv}} = f(\omega, \tau). \tag{3-49}$$

Simplification of the multi-motor and multi-inverter model

As the torque requests and speed for all the motors and inverters will always be the same, it might not be necessary to model all four motors and inverters.

The motors and inverters in parallel are always cooled at the same coolant mass flow rate, such that

$$\dot{m}_{\text{motors, inverters}} = 0.5\dot{m}_{\text{pump}}, \tag{3-50}$$

where $\dot{m}_{\text{motors, inverters}}$[kg·s^{-1}] is the mass flow rate at the motors and inverters and $\dot{m}_{\text{pump}}$[kg·s^{-1}] is the mass flow rate at the pump, before the split. Therefore only one of the two sets of motors and inverters needs to be modelled.

When two components are cooled in series, both components are not cooled by a coolant with the same temperature, so the heat dissipation is not equal. Depending on the heat transfer coefficient, the heat dissipation might still be very similar. The two motors can be modelled as one when some model adaptions are made. The heat capacity is doubled, as described by

$$C_{\text{motors, simpl}} = 2C_{\text{motor}}, \tag{3-51}$$

where $C_{\text{motors, simpl}}$[J·K^{-1}] is the heat capacity used in the simplified model. The induced heat is adapted in the same way and defined as

$$\dot{Q}_{\text{losses, motors, simpl}} = 2\dot{Q}_{\text{losses, motor}}, \tag{3-52}$$

where $\dot{Q}_{\text{losses, motors, simpl}}$[W] is the induced heat due to losses in the simplified model. Assuming now that both series components have the same temperature. The heat transfer efficiency coefficient can be rewritten using the rule of series components described in Equation 3-16, such that

$$\epsilon_{\text{motors, simpl}} = 2\epsilon_{\text{motor}} - \epsilon^2_{\text{motor}}, \tag{3-53}$$

where $\epsilon_{\text{motors, simpl}}[\ ]$ is the heat transfer efficiency coefficient for the simplified model, assuming that the temperature of the two motors is the same, which has to be validated through experiments. This same approach can be taken for the inverters.

3-5 Vehicle dynamics

A simple vehicle model can be established by using basic physical equations. Based on the traffic information, road conditions and weather forecast, that are assumed to be available, the future propulsion power P_{pr} can be estimated as shown in Equation 3-54 [36]. If there is no traffic flow or road information available, the future speed and acceleration can also be estimated based on the current speed and acceleration using a Markov chain, developed using statistical data [37] [51] or a Neural Network (NN), developed using statistical data [52, 6]. As the vehicle speed and acceleration are related to the losses in the battery, the heat generation can also be directly predicted using Markov chains based on the current speed and acceleration [29] or using traffic flow information [42] In this thesis, it is assumed that a speed profile and an elevation profile are available.

The battery power consumption P_{pr}[W] used for vehicle propulsion is expressed by

$$P_{\text{pr}} = \begin{cases} \frac{F_{\text{tr}} v_{\text{car}}}{\eta_{\text{dt}}}, & \text{if } F_{\text{tr}} \geq 0 \\ F_{\text{tr}} v_{\text{car}} \eta_{\text{dt}} & F_{\text{tr}} < 0, \end{cases} \tag{3-54a}$$

where F_{tr}[N] is the total traction force and η_{dt}[] is the efficiency of the drivetrain. For discharging ($F_{\text{tr}} > 0$), the battery power is divided by the drivetrain efficiency. For charging, whilst regenerative braking, ($F_{\text{tr}} > 0$), we multiply by the drivetrain efficiency,

The traction force F_{tr}[N] is described by

$$F_{\text{tr}} = F_{\text{roll}} + F_{\text{aero}} + F_{\text{grav}} + F_{\text{accel}}, \tag{3-54b}$$

where F_{roll}[N] is the rolling resistance, F_{aero}[N] is the aerodynamic drag, F_{grav}[N] is the gravitational force and F_{accel}[N] is the inertial force. The rolling resistance is defined as

$$F_{roll} = m_{\text{car}} g c_{rr}, \tag{3-54c}$$

depending on the gravitational acceleration g [m·s^{-2}] and the dimensionless rolling resistance coefficient c_{rr}[ ]. The aerodynamic drag is described by

$$F_{\text{aero}} = \frac{1}{2} \rho_{\text{air}} C_{\text{d}} A_{\text{fr}} (v_{\text{wind}} + v_{\text{car}})^2, \tag{3-54d}$$

where ρ_{air}[kg·m^{-3}] is the air density, C_{d}[ ] is the dimensionless drag coefficient, A_{fr}[m^2] is the frontal area and v_{wind}[m^{-1}] wind speed. The gravitational force is expressed by

$$F_{\text{grav}} = m_{\text{car}} g sin(\alpha_{\text{road}}), \tag{3-54e}$$

where α_{road}[°] is the slope of the road. The inertial force,

$$F_{accel} = m a_{\text{car}}, \tag{3-54f}$$

depends on the acceleration of the vehicle a_{car}[m·s^{-2}]. The torque on a wheel can be described as a function of the rotational speed ω[s^{-1}] and propulsion power by

$$\tau_{=} P_{\text{pr}} / 4 / \omega. \tag{3-54g}$$

The rotational speed of the motor is proportional to the vehicle speed and expressed by

$$\omega \propto v_{\text{car}}. \tag{3-54h}$$

3-6 Resulting state-space model of the TMS

This section summarizes the model used to describe the Thermal Management System (TMS). In the following equations, the parameters a_i, b_i, c_i, and d_i need to be identified through system identification experiments. The following states x are used to model the TMS dynamics:

$$x = \begin{pmatrix} T_{\text{battery}} \\ T_{\text{motor}} \\ T_{\text{inverter}} \\ T_{\text{cool, chiller}} \\ T_{\text{cool, battery}} \\ T_{\text{cool, rad}} \\ T_{\text{cool, motor}} \\ T_{\text{cool, inverter}} \end{pmatrix}. \tag{3-55}$$

The differential state equations $\dot{x}$ describing the system dynamics are expressed by Equation 3-56 on the following page.

$$\dot{x} = \begin{pmatrix} \frac{(a_1 - a_2 T_{\text{battery}}) I_{\text{battery}} - \epsilon_{\text{battery}}(\dot{m}_1)(T_{\text{battery}} - T_{\text{cool, chiller}})\dot{m}_1 c_{\text{cool}}}{C_{\text{cat}}} \\ \frac{2f_{\text{motor}}(\tau,\omega) + 2a_3\tau(1+\alpha(T_{\text{motor}} - T_0) - (2\epsilon_{\text{motor}}(0.5\dot{m}_2) - \epsilon_{\text{motor}}(0.5\dot{m}_2)^2)((T_{\text{motor}} - T_{\text{cool, inverter}})0.5\dot{m}_2 c_{\text{cool}}}{2C_{\text{motor}}} \\ \frac{2f_{\text{inverter}}(\tau,\omega) - (2\epsilon_{\text{inverter}}(0.5\dot{m}_2) - \epsilon_{\text{Inverter}}(0.5\dot{m}_2)^2)(T_{\text{inverters}} - T_{\text{cool, rad}})0.5\dot{m}_2 c_{\text{cool}}}{2C_{\text{inverter}}} \\ \frac{w_1(T_{\text{cool, battery}} - T_{\text{cool, chiller}})\dot{m}_1 c_{\text{cool}} + (1-w_1)(T_{\text{cool, rad}} - T_{\text{cool, chiller}})\dot{m}_1 c_{\text{cool}} - \dot{Q}_{\text{chiller}}}{m_{\text{cool, chiller}} c_{\text{cool}}} \\ \frac{(T_{\text{cool, chiller}} - T_{\text{cool, battery}})\dot{m}_1 c_{\text{cool}} + \epsilon_{\text{cat}}(T_{\text{battery}} - T_{\text{cool, chiller}})\dot{m}_1 c_{\text{cool}}}{m_{\text{cool, battery}} c_{\text{cool}}} \\ \frac{w_1((T_{\text{cool, mot}} - T_{\text{cool, rad}})\dot{m}_2 c_{\text{cool}} - \text{UA}_{\text{rad}}(\dot{m}_2)(T_{\text{cool, mot}} - T_{\text{amb}})) + (1-w_1)((T_{\text{cool, rad, in}} - T_{\text{cool, rad}})(\dot{m}_1 + \dot{m}_2)c_{\text{cool}} - \text{UA}_{\text{rad}}(\dot{m}_1 + \dot{m}_2)(T_{\text{cool, rad, in}} - T_{\text{amb}})}{m_{\text{cool, rad}} c_{\text{cool}}} \\ \frac{(T_{\text{cool, inverter}} - T_{\text{cool, motor}})0.5\dot{m}_2 c_{\text{cool}} + (2\epsilon_{\text{motor}} - \epsilon_{\text{motor}}^2)(T_{\text{motor}} - T_{\text{cool, inverter}})0.5\dot{m}_2 c_{\text{cool}}}{2m_{\text{cool, motor}} c_{\text{cool}}} \\ \frac{(T_{\text{cool, rad}} - T_{\text{cool, inverters}})0.5\dot{m}_2 c_{\text{cool}} + (2\epsilon_{\text{inverter}} - \epsilon_{\text{Inverter}}^2)(T_{\text{Inverter}} - T_{\text{cool, rad}})0.5\dot{m}_2 c_{\text{cool}}}{2m_{\text{cool, inverter}} c_{\text{cool}}} \end{pmatrix}, \tag{3-56}$$

where the mode w_1 describes the valve positions and is placed as a multiplier in front of the differential equation describing the dynamics for that valve configuration, as required by the outer convexification method which is described in more detail in section 4-4. The coolant temperature at the inlet of the radiator $T_{\text{cool, rad, in}}$ is defined as

$$T_{\text{cool, rad, in}} = \frac{T_{\text{cool, battery}}\dot{m}_1 + T_{\text{cool, mot}}\dot{m}_2}{\dot{m}_1 + \dot{m}_2}, \tag{3-57}$$

the heat transfer efficiency coefficient $\epsilon(\dot{m})$ for each of the components is defined as

$$\epsilon(\dot{m}) = 1 - \exp\left(-\frac{a_1 \dot{m}^{a_2}}{c_{\text{cool}}\dot{m}}\right), \tag{3-58}$$

and the heat transfer coefficient multiplied by the heat transfer surface UA_{rad}, is expressed by

$$\text{UA}_{\text{rad}}(\dot{m}_{\text{cool}}) = \frac{1}{\frac{1}{a_1 \dot{m}_{\text{cool}}^{b_1}} + \frac{1}{a_2 \dot{m}_{\text{amb}}^{b_2}}}. \tag{3-59}$$

The $N = 8$ temperature states are linearly constrained to their safety limits.

$$\sum_{i=1}^{N} T_{\text{min, i}} \leq T_{\text{i}} \leq T_{\text{max, i}}, \tag{3-60}$$

The model is nonlinear and discontinuous, due to the fact that the mode $w_1 \in 0, 1$. The control inputs, the mode w_1 describing the valve positions, coolant and ambient air mass flow rates $\dot{m}_1$, $\dot{m}_2$ and $\dot{m}_{\text{amb}}$, and the chiller heat dissipation $\dot{Q}_{\text{chiller}}$ are denoted by u.

$$u = \begin{pmatrix} w_1 \\ \dot{m}_1 \\ \dot{m}_2 \\ \dot{m}_{\text{amb}} \\ \dot{Q}_{\text{chiller}}. \end{pmatrix} \tag{3-61}$$

The $L = 5$ control inputs are also linearly constrained.

$$\sum_{i=1}^{L} u_{\text{min, i}} \leq u_{\text{i}} \leq u_{\text{max, i}}, \tag{3-62}$$

The time-varying disturbances, denoted by d, are the vehicle speed v_{vehicle}, ambient temperature T_{ambient}, motor torque τ, motor rotational speed ω and battery current I_{battery}.

$$d = \begin{pmatrix} v_{\text{vehicle}} \\ T_{\text{ambient}} \\ \tau \\ \omega \\ I_{\text{battery}}. \end{pmatrix} \tag{3-63}$$

The objective J is to minimize the total induced heat $\dot{Q}_{\text{induced, total}}$, TMS power consumption $P_{\text{consumed, total}}$ and the ageing rate $\alpha_{\text{ageing}}(T)$ and is expressed by

$$J = \int_{t=0}^{t_N} C(\alpha_{\text{ageing}}) + \dot{Q}_{\text{losses, total}} + P_{\text{consumed, total}}, \tag{3-64}$$

where C is the ageing weight used to trade off ageing and energy consumption and t_N is the time of the trip. The ageing rate α_{ageing} is defined by

$$\alpha_{\text{ageing}} = a_1 T_{\text{battery}}^4 + a_2 T_{\text{battery}}^3 + a_3 T_{\text{battery}}^2 + a_4 T_{\text{battery}} + a_5. \tag{3-65}$$

The total induced heat $\dot{Q}_{\text{losses, total}}$ is the sum of motor and battery temperature-dependent losses $\dot{Q}_{\text{losses, motor}}$ and $\dot{Q}_{\text{losses, battery}}$.

$$\begin{aligned} \dot{Q}_{\text{losses, total}} = &\underbrace{4a\tau^2(1 + \alpha(T_{\text{motor}} - T_0))}_{\dot{Q}_{\text{losses, motor}}} + \\ &\underbrace{(a - bT_{\text{battery}})I_{\text{battery}}^2}_{\dot{Q}_{\text{losses, battery}}}. \end{aligned} \tag{3-66}$$

The total power consumption $P_{\text{consumed, total}}$ is the sum of the power consumption of the fan P_{fan} and compressor $P_{\text{compressor}}$. The power consumption of the pumps is described by $P_{\text{pump,1}}$, $P_{\text{pump,2}}$ and an additional term $P_{\text{pump, 1,2}}$ to capture the nonlinearity when both pumps are turned on and $w_1 = 0$.

$$\begin{aligned} P_{\text{consumed, total}} = & \underbrace{a_1 \dot{m}_{\text{amb}}}_{P_{\text{fan}}} + \\ & \underbrace{\frac{\dot{Q}_{\text{chiller}}}{\text{COP}}}_{P_{\text{compressor}}} + \\ & w_1 (\underbrace{b_1 m_1^3 + b_2 m_1^2 + b_3 m_1}_{P_{\text{pump,1}}} + \underbrace{c_1 m_2^3 + c_2 m_2^2 + c_3 m_2}_{P_{\text{pump,2}}}) + \\ & (1 - w_1)(\underbrace{d_1 m_1^3 + d_2 m_1^2 + d_3 m_1}_{P_{\text{pump,1}}} + \underbrace{e_1 m_2^3 + e_2 m_2^2 + e_3 m_2}_{P_{\text{pump,2}}} + \underbrace{f_1 m_1 m_2}_{P_{\text{pump, 1,2}}}), \end{aligned} \tag{3-67}$$

where the remaining coefficients are to be identified.

Chapter 4

Control methods for the TMS

The goal of energy-efficient thermal management is to find the optimal control scheme that trades off component efficiency, actuator power consumption and battery ageing within the Thermal Management System (TMS). First, the problem of optimal control is explained and optimal control methods are discussed. Then, the concept of Model Predictive control (MPC) is explained and examples found in literature in the context of thermal management are discussed. Then, discretization methods, approaches to the mixed-integer problem, integration methods and optimization methods are mentioned. Finally, the concept of hierarchical control is explained and the possibilities for hierarchical control in the TMS is discussed

4-1 Optimal control methods

Optimal control strategies are used to find optimal methods to control dynamic processes. The following section explains the concept of optimal control based on [53] and elaborates on several methods in more detail. Let $x(t)$ be the state variable at time t, $u(t)$ be the control variable at time t, and $d(t)$ be disturbances to the process, the state dynamics can be described by differential equations, as discussed in chapter 3. Given a certain initial value of the state x_0 and chosen control inputs over a time interval $0 \leq t \leq T$, these differential equations can be integrated to get the state trajectory over time. The goal is to choose the control inputs such that the objective function is minimized,

$$J = \int_{t=0}^{T} F(x(t), u(t), d(t), t)dt. \tag{4-1}$$

The objective could be to minimize energy consumption over time. The Optimal Control Problem (OCP) is to find a control action that maximizes the objective, subject to the state

continuation constraint, and possible other inequality constraints c or equality constraints r:

$$\begin{aligned} \min_{x,u} \quad & \int_{t=0}^{T} F(x,u,d,t)dt, \\ s.t. \quad & \frac{dx}{dt} = f(x,u,d,t), \\ & x(0) = x_0, \\ & 0 \leq c(x,u,d,t), \\ & 0 = r(x,u,d,t). \end{aligned} \tag{4-2}$$

There are basically two different ways of solving an OCP: Direct and indirect methods. Both methods and the concept of dynamic programming are discussed in this section.

4-1-1 Dynamic programming

Dynamic Programming (DP) uses Bellman's principle of optimality which states that any intermediate step in an optimal trajectory is optimal. Iterating backwards from the end of the control trajectory, a minimization problem is solved for all feasible initial states of that intermediate step, and all possible control inputs. The feasible trajectory with the lowest cost is the optimal trajectory [2].

DP can be used to get a benchmark solution to the optimal control problem, as was done in [5] for cabin conditioning. Continuous signals need to be discretized at a certain resolution which always will give a trade-off between accuracy and computation time. As DP searches the entire space, the global optimum is always found but it is slow in computation due to the curse of dimensionality and requires large processor memory [54]. Therefore it is not well suited for the real-time control problem.

4-1-2 Indirect methods

Indirect methods optimize in an infinite-dimensional function space. These methods are referred to as "first optimize, then discretize". The necessary conditions of optimality have to be derived for each problem, and again for each small change of initial conditions or constraints to transform the OCP into a Boundary Value Problem (BVP). As human insight into the problem is required, the numerical solution of an optimal control problem using indirect methods cannot be fully automated and therefore indirect methods have not appeared as a suitable method for real-time control [2].

4-1-3 Direct methods

Direct methods on the other hand are based on the discretization of the continuous OCP into a finite-dimensional Nonlinear Programming (NLP) problem. Direct methods can be sketched as "first discretize, then optimize". The formulated NLP problem can then be solved by an optimization algorithm and is therefore suited for real-time use.

4-2 Model Predictive Control

When aiming to solve an optimal control problem, the goal is to find a solution to the Ordinary Differential Equation (ODE)'s that maximizes or minimizes the objective. The concept of MPC is a direct method that uses optimization to find a control action taking into account the future behaviour of the system. This opens up the possibility of reacting to time-varying disturbances. [2].

The basic MPC algorithm is explained based on [55]:

1. The state $x(i)$ is measured at time i. The initial value for the OCP is $x_0 = x(i)$.

2. Solve the following optimization problem:

$$\begin{aligned} \min_{x,u} \quad & \sum_{k=0}^{N_\mathrm{p}-1} \ell(x(k), u(k), d(k)), \\ \text{s.t.} \quad & x(k+1) = f(x(k), u(k), d(k)), \\ & x(0) = x_0, \\ & 0 \leq c(x(k), u(k), d(k)), \\ & 0 = r(x(k), u(k), d(k)), \end{aligned} \tag{4-3}$$

   where N_p denotes the prediction horizon and $\ell(x(k), u(k), d(k))$ is the value of the objective function at timestep k. The found optimal control trajectory is denoted by u^*.

3. Apply the first input $u^*(0)$ to the system in the next sampling period T_s, after which the process is repeated.

The concept of MPC is visualized in Figure 4-1. Note that the original OCP is now discretized, as the system is controlled at a discrete sampling time T_s. MPC always consists of several parts:

- Prediction model. The dynamical model and the system constraints are presented in the chapter 3.
- Objective function. The objective of minimizing energy consumption and maximizing the High Voltage Battery (HVB) lifetime was also presented in chapter 3.
- Integrator. The integrator is required to solve the model ODE's over time.
- Optimizer. The optimizer is required to find the control inputs that fulfil the constraints and minimize the objective.

In the following sections, several examples will be given in TMS application and the length of the prediction horizon is discussed.

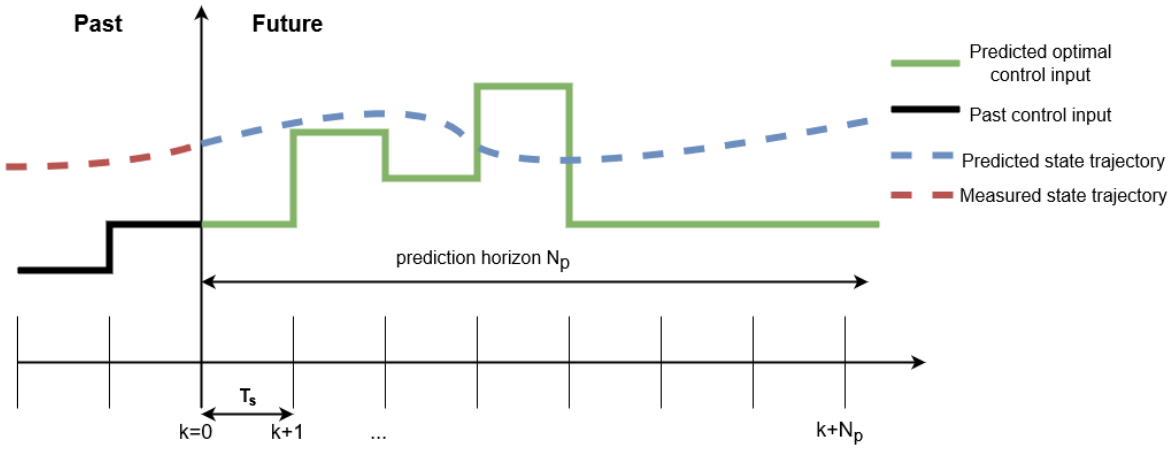


Figure 4-1: Visualization of the MPC principle

4-2-1 MPC in general thermal management applications

MPC has been used before in the context of thermal management. It was used in the application of building thermal management, where predictable forecasts such as weather information [21] [56] and occupancy grids [20] [22] were used to improve the control performance. [21] compared MPC using weather forecast to traditional rule-based control and 17% energy was saved and occupants comfort was improved. In [20], adding occupancy forecast information in MPC allowed for up to 40 % lower energy consumption annually for cooling compared to conventional MPC.

In the vehicle context, MPC was also used for battery conditioning, powertrain conditioning and vehicle cabin conditioning. For cabin conditioning, MPC was compared to an on-off controller. Weather and traffic forecast information was used to predict the future thermal load, causing an estimated battery lifetime improvement of up to 19% [36]. In [57], passenger amount variation was predicted in buses and up to 6% of energy was saved compared to a rule-based controller, respectively. In [37], both velocity predictions and weather forecasts were used to improve the energy consumption both energy consumption and passenger comfort were improved by 12 and 50% compared to a rule-based controller. In [38], a nonlinear MPC strategy was developed. A linear quadratic approximation was applied which allowed for between 20 and 40% energy saved compared to the traditional controller and at maximum consumed only 4 % more compared to the nonlinear strategy, depending on the driving scenario.

For battery conditioning for electric vehicles, MPC using speed prediction and a variable target temperature based on the ambient temperature was used in [6], where improved battery lifetime and between 25 % energy was saved compared to an on-off controller and 14% compared to conventional MPC. In [30], MPC was compared to PID control and 10 - 17% energy consumption was achieved depending on the drive cycle.

For conditioning of the motors and inverters, MPC was also proposed in [14]. Here between 7 and 86% of energy was saved compared to the benchmark over different drive cycles. MPC has also been tested successfully focusing purely on reducing thermal stress in [24]

4-2-2 MPC for integrated vehicle thermal management systems

The control problem for a hybrid system can be solved by various techniques, the most simple will employ heuristic knowledge in the form of a state machine, combined with PID controllers. MPC can also be used to control a hybrid system, but the integer requirements must be taken into account to achieve usable results.

As solving a mixed-integer problem is inherently complex, a linear formulation of the TMS dynamics is proposed in [58]. The system is described as a linear Mixed Logical Dynamical (MLD) system, where some of the inputs are binary (valves) and some are continuous (actuators). These inputs are then used to choose an energy optimal mode and actuator input in an integrated thermal management system. The concept is promising, but no real nonlinear vehicle setup is presented, and no approach is shown to linearize such nonlinear dynamics.

Cabin and battery thermal management were integrated in [4, 7], and cabin and powertrain thermal management was integrated in [18]. However, the used setups do not include valves, which significantly simplifies the respective control problems.

A nonlinear hybrid MPC was developed in [1] to thermally control the battery and power electronics, excluding the motors. The setup considered is visualized in Figure 4-2.

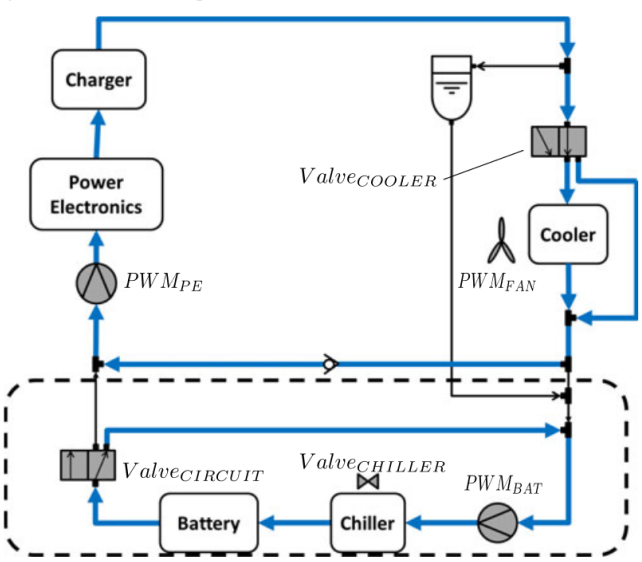


Figure 4-2: The hardware setup considered in [1]

The possible positions for the used valve are binary values as they can only be open or closed, but the approach of rounding was investigated to decrease model complexity. The problem was simplified by assuming that the coolant reaches the same temperature as the battery at the outlet, no equations were used to describe the heat transfer efficiency, which is generally not a valid assumption. The performance was validated and they have shown that compared to the baseline, for rounding and non-rounding, an almost equal decrease in energy consumption of 5.6% and 5.2% was achieved, with significantly decreased computation time for the rounding case. The rounding-based solution was solved in 1.5

seconds. The cost function by a part that represented the power consumption of the TMS and a part penalizing the battery temperature, taking into account both performance and lifetime. Continuation of the work was discussed [59], where this was taken a step further and implemented on a rapid prototyping module and testing the actual car instead of in simulation. Decreases of 8-50% were achieved in the cost function, depending on the drive cycle.

4-2-3 Prediction horizon

As the thermal dynamics are slow, a prediction horizon of multiple minutes adds benefits. This can be seen based on previously done research that used MPC for thermal management, as summarized in Table 4-1. Different horizon lengths are used depending on the required computation time and accuracy, but the length is at least 30 seconds, and up to several minutes. The research with shorter lengths assume a fixed target temperature [60] [30] and are not considering multiple modes and valves position [29]. None of these systems includes other components than the HVB. The research with an extremely long horizon does note that it is hard to use long previews that are accurate [7].

Publication	**Cabin**	**Battery**	**Inverters**	**Motors**	**Sample time [s]**	**Prediction Horizon [s]**
[1]		X	X		2.5	200
[4]	X	X			5	60 - 180
[7]		X	X	X	5	1375 (entire cycle)
[14]			X	X	15	900
[6]		X			?	60
[29]		X			?	5
[60]		X			5	20
[61]		X			1	5-60
[30]		X			1	10
[62]		X			1	30-180
[33]		X			?	10-50

Table 4-1: Prediction horizon in previous TMS research. an "X" denotes wich components are controlled. A "?" indicates that the sample time was not given.

4-3 Discretization methods

Direct methods are based on a discretization of the OCP such that it can be solved by an optimization algorithm. The methods of direct single shooting, direct collocation and direct multiple shooting are explained in detail in [63, 2] and summarized in the following section.

4-3-1 Direct single shooting

Direct single shooting is a so-called direct sequential method, where the steps of system simulation and optimization are performed sequentially. The state trajectory is regarded as

an implicit function of the controls and the initial state. An ODE solver is used to integrate the ODE's and simulate the system behaviour. Only, the control inputs are discretized which means the state trajectory cannot be directly constrained. Additionally, initialization of the state trajectory using prior information is not possible [2]. This means disturbance predictions cannot be used within the prediction horizon and additionally, the state trajectory can be constrained, which is required in the thermal management problem.

4-3-2 Direct collocation

Direct collocation methods do discretize both the states and controls into a fine grid. Opposed to direct single shooting, state trajectory variables can now be initialized, at the cost of a larger resulting optimization problem. As the discretization can be done at any step size, in theory, the NLP problem can be solved efficiently. The issue is that often, it is not known at what point in time, small timesteps are required. Adaptive stepsize solvers cannot be used for direct collocation and optimizing the entire problem using very small timesteps is not efficient, as computation would take a lot of time [2].

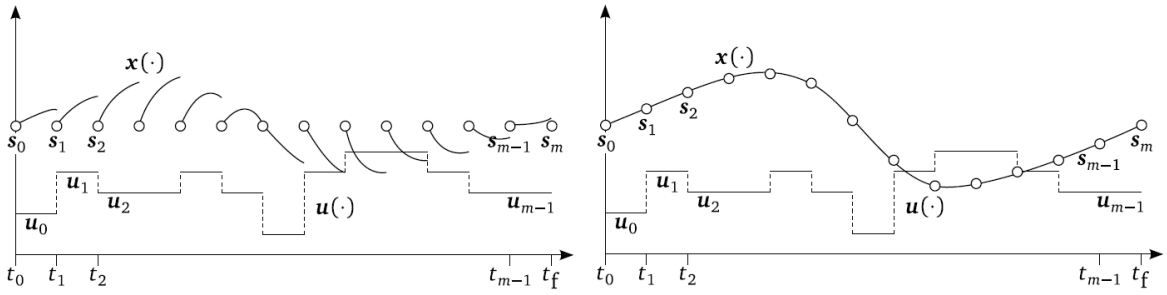


Figure 4-3: In direct multiple shooting, initially the state trajectory is not continuous, as can be seen on the left image. When a feasible solution is found, the state trajectory is continuous, as can be seen on the right image [2].

4-3-3 Direct multiple shooting

Direct multiple shooting inherits advantages from both collocation and single shooting approaches: The state trajectory is discretized but finding a solution still relies on solving initial value problems. The control trajectory is discretized into m shooting intervals

$$a = t_0 < t_1 < ...t_{m-1} < t_m = b. \tag{4-4}$$

On each interval $[t_i, t_{i+1}]$, $0 \leq j \leq m-1$, an Initial Value Problem (IVP) is defined:

$$\frac{dx}{dt} = f(t, x), \quad x(t_i) = s_i. \tag{4-5}$$

By a suitable choice of unknown vector s_i, the solutions on the shooting intervals are constrained such that a continuous function results that is continuous on the entire interval $[a, b]$:

$$x(t_{i+1}; s_i) = s_{i+1}, \quad i = 0, m-2. \tag{4-6}$$

For all intervals, simulation and optimization can be done simultaneously until the objective is reached and the constraints are fulfilled. To clarify this, the concept of direct multiple shooting is shown in Figure 4-3. Initially, the state trajectory is physically not possible, as it is not continuous. After several iterations, a continuous state trajectory is achieved. If N is the number of shooting intervals, there are N times many more variables in the optimization problem compared to direct single shooting, but also more structure in the optimization problem. By condensing the matrices, this can be reduced to the same size as for direct single shooting. Additionally, there is the advantage that initialization and constraining of state trajectories can be done. Due to the division of the problem into smaller shooting intervals, there are also better feasibility properties compared to direct single shooting.

Compared to direct collocation, there is the advantage that they easily allow using efficient adaptive solvers. This is due to the fact that the state discretization is decoupled from the control discretization. Therefore, the direct multiple shooting method is most suitable for our problem.

4-4 Mixed-integer programming

Solving mixed-integer optimization problems is computationally complex compared to solving continuous optimization problems due to the fact that some variables are constrained to have an integer value. In the TMS, the integer inputs are valves. Whether a valve is in one position or another, determines the flow of coolant. One approach would be

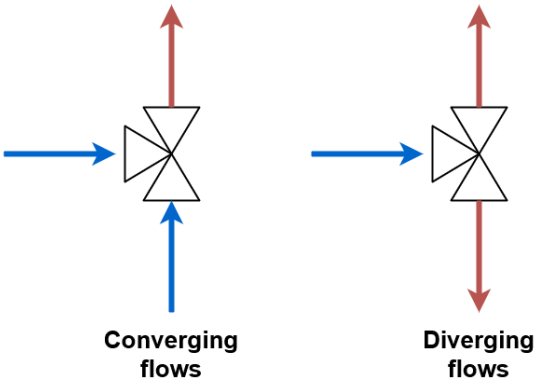


Figure 4-4: Valves determine the flow of coolant.

to fix the integer control variable for every possible choice and then solve a continuous optimal control problem [2]. Branching methods and relaxation methods can be used to solve mixed-integer problems and are discussed in more detail.

4-4-1 Branching methods

Mixed-integer problems can be solved using branching techniques [2]. Using branching methods, a systematic enumeration of candidate solutions for the integer variables is done. Iteratively for each chosen candidate solution, continuous optimization can be done. The

idea is that this systematic search does not require solving the problem for each possible integer input, but only for those that might give a better result. The efficiency of branch-and-bound methods, therefore, depends on how efficiently the systematic search is done.

4-4-2 Relaxation methods

Relaxation methods aim at reformulating the mixed-integer problem as a continuous problem, which can be solved using regular NLP solvers.

Inner convexification The most simple approach would be simply to drop the integrality requirement of the integer control variable. Consequently, the solution found by the NLP solver will not necessarily be an integer one, so must be rounded, and therefore also is not guaranteed to be optimal [2].

Outer convexification In outer convexification, the problem is reformulated such that the binary constraint can be relaxed. First, the term 'modes' must be clarified. When a mode is chosen, a logical combination of valve positions is chosen, which allows the flow to either reach the heater or the radiator for example. First of all, a new binary variable w_i is

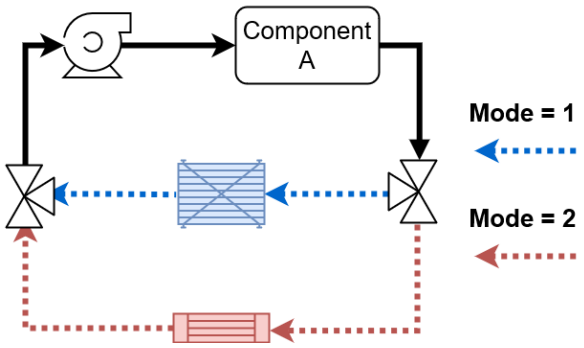


Figure 4-5: Modes also determine the flow of coolant.

introduced for all n possible 'modes'.

$$w = \{0, 1\} \tag{4-7}$$

All functions f depending on the integer control function are partially convexified, with respect to the integer variable only. This means that the introduced binary variables act as convex multipliers.

$$f(x, u, w) = w_1 f_1(x, u) + w_2 f_1(x, u)... \tag{4-8}$$

Additionally, an equality constraint is added that enforces that the sum of discrete options is 1.

$$\sum_{n=0}^{N} w_i = 1 \tag{4-9}$$

As a final step, these binary variables are relaxed to have real values between 0 and 1. Using this reformulation, even after relaxation, all model functions are often evaluated in integer feasible choices only.

$$w \in [0, 1] \tag{4-10}$$

The outer convexification method allows for efficient problem-solving and often binary feasible solutions but the performance needs to be compared to a reference controller in order to be able to judge the performance. Next to the dynamics, constraints can also be relaxed. A constraint depending on an integer input can be described as follows

$$g(x(t), u(t), w(t)) \geq 0 \tag{4-11}$$

The outer convexified constraint can be expressed by

$$w(t)g(x(t), u(t), w(t)) \geq 0 \tag{4-12}$$

The optimal rounded solution will still satisfy the original constraint but this constraint does not satisfy the Linear Independence Constraint Qualification (LICQ) at w(t) = 0, as the constraint gradient cannot be determined. Such constraints can be relaxed using a relaxation parameter M > 0

$$g(x(t), u(t), w(t)) \geq -M(1 - w(t)) \tag{4-13}$$

Which satisfies the constraint when w(t) is 1 or 0 but can attract fractional solutions.

4-5 Integration methods

To solve the OCP, we need to know how the states evolve over time for different control trajectories. The background on numerical ODE solvers given in this section is based on [64]. Integration methods solve IVP's for ODE's of the form

$$\frac{dx}{dt} = f(x, t) \tag{4-14}$$

over a certain time interval, t $\in [t_0, t_f]$, where the initial value of x is known

$$x(t_0) = x_0 \tag{4-15}$$

Numerical methods are used to find the solution to the initial value problem. The performance is measured by the computation time and accuracy. To measure the accuracy, let us define the Local Truncation Error (LTE).

Definition 4-5.1 (Local Truncation Error (LTE))**.** The LTE, T_{n+1}, is defined to be the difference between the exact and the numerical solution of the IVP at time t = t_{k+1}:

$$T_{k+1} = x(t_{k+1})x_{k+1}, \tag{4-16}$$

under the localizing assumption that $x_n = x(t_n)$, i.e. that the current numerical solution x_n is exact. The LTE is the error made in a single step.

whereas the Global Error (GE) is defined as follows

Definition 4-5.2 (Global Error (GE))**.** The GE,

$$e_k = x(t_k)x_k, \tag{4-17}$$

is defined as the error between an approximation of the state value $x(t_k)$ and the exact state value x_k. Therefore the global error can be seen as the cumulative of LTE's.

The following section elaborates on the difference between explicit and implicit ODE solvers. Additionally, the difference between fixed-step and adaptive solvers is explained.

Explicit methods The most common method for integration is the explicit (forward) Euler method, which linearizes the differential equation at the current timestep.

$$x_{\mathrm{k+1}} = x_{\mathrm{k}} + hf(x_{\mathrm{k}}, t_{\mathrm{k}}) \tag{4-18}$$

where h is the step size and t_{k} is the time at step k. The accuracy of the explicit Euler method depends on the size of the integration stepsize and the stiffness of the system. LTE$\propto h^2$ and for a wide range of problem, it can be proven that GE $\propto h$. The stability region of explicit Euler is bounded, which is of importance for stiff systems. Stiff systems require impractically small step sizes in a priori unknown regions [2], causing the integration accuracy to be more accurate than needed [64] and the computation slower. As the TMS model includes different states with both slow- and fast-changing dynamics, the battery temperature changes slowly and the coolant temperature changes quickly, an explicit method might not be optimal.

Compared to forward Euler, explicit Runge-Kutta (RK) allows for a more accurate estimation of the derivative. A four-step RK scheme (RK4) looks as follows:

$$x_{\mathrm{k+1}} = x_{\mathrm{k}} + h\frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \tag{4-19}$$

where

$$\begin{aligned} k_1 &= f(x_k, t_k), \\ k_2 &= f(x_k + h\frac{k_1}{2}, t_k + t_s/2), \\ k_3 &= f(x_k + h\frac{k_2}{2}, t_k + t_s/2), \\ k_4 &= f(x_k + hk_3, t_k + h, \end{aligned} \tag{4-20}$$

such the solution is influenced by the slope at different points of the integration interval. Using the RK4 scheme, LTE $\propto h^5$. It has to be noted that if an ODE solver requires a smaller stepsize compared to another solver, also needs more computations to reach the same accuracy.

Implicit methods Implicit methods however suffer less from the small stability region as the ODE is linearized at the next timestep $k+1$ instead of the current timestep k. For implicit (backward) Euler this looks as follows

$$x_{\mathrm{k+1}} = x_{\mathrm{k}} + hf(x_{\mathrm{k+1}}, t). \tag{4-21}$$

Similarly, as for explicit methods, an implicit RK scheme can be used. Implicit methods do require an additional computation compared to explicit methods. Equation 4-5 needs to be solved as the next state at k+1 is not directly known. Backward Differentiation Formula (BDF) methods are a family of implicit methods and are so-called linear multistep methods, that use previously computed steps to calculate the derivative in the next step. The general equation for a k-step BDF can be written as

$$x_{n+k} + \alpha_{k-1}x_{n+k-1} + ... + a_0 x_n = h\beta f(t_{n+s}, x_{n+s}), \tag{4-22}$$

where the remaining free coefficients are chosen to achieve order k. A BDF with an order of one equals the backward Euler method. As the equation for $xn + s$ can be written as a function of the state at $xn + k$ for $k = 0, 1, ..k$, reuse of previously computed solutions at every new timestep is possible. BDF methods are stable up to order 6.

Adaptive solvers When using fixed-step integrators, the integration error is not controlled online. An adaptive stepsize solver can be used which adapts the stepsize such that the integration error never exceeds the allowed error. Both RK and BDF methods can be used with an adaptive solver. The disadvantage is that the computation time is not known and depends on the to-be-solved system of ODE's.

4-6 Optimization methods

Various methods for solving NLP problems exist and different approaches were also used in the context of vehicle thermal management.

Particle swarm optimization was used for solving the battery thermal management problem [30], as well as the genetic algorithm [34]. Also Sequential Quadratic Programming (SQP) [1] [59] and interior point methods [4] [62] are used. The aforementioned methods are discussed in more detail.

4-6-1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a global optimization method iteratively trying to improve a possible solution. Initially, there is a population of possible solutions, particles, which can move in the search space according to mathematical formulas that describe the particle velocity and position. The idea is that over time, the 'swarm' of particles moves towards the optimal solution. PSO does not use the hessian of the optimization problem. As PSO does not use the hessian, convergence towards a global, or local optimum cannot be guaranteed.

4-6-2 Genetic Algorithm

The genetic algorithm is also a global optimization algorithm, inspired by the process of natural selection. Similarly with PSO, the evolution starts with a population of randomly generated individuals, which is called a 'generation'. Iteratively, individuals within the

population are selected to reproduce children such that the population changes. Over several generations, the population moves towards a better solution. As PSO, the genetic algorithm does not use the hessian, convergence towards a global, or local optimum cannot be guaranteed.

4-6-3 Sequential Quadratic Programming

SQP does use the hessian of the optimization problem and does find a local optimum, which is not guaranteed in the previous methods. SQP attempts to model the nonlinear problem iteratively by a quadratic subproblem to define a step towards the minimum of the nonlinear problem [65]. Depending on the problem, very low computation times can be achieved. The concept is explained for a general NLP problem, expressed by

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & b(x) = 0, \qquad c(x) \geq 0. \end{aligned} \tag{4-23}$$

The method of Lagrange multipliers can be used to find the minimum of a function subject to equality constraints. The Lagrangian $\mathcal{L}$ captures the cost function and constraints in a single function and for the problem described by Equation 4-23 is defined as

$$\mathcal{L} = f(x) - \lambda b(x) - \sigma c(x) \tag{4-24}$$

Where λ is the Lagrange multiplier for equality constraints and σ is the Lagrange multiplier for inequality constraints. At a new iteration k, a search direction $d(k)$ is defined by solving quadratic subproblems to the main problem. The subproblem resembles a Taylor expansion of the objective function and is described by

$$\begin{aligned} \min_{d} \quad & = f(x_\mathrm{k}) + \nabla f(x_\mathrm{k})^T d + \frac{1}{2} d^T \nabla^2_{\mathrm{xx}} \mathcal{L}_\mathrm{k} p \\ \text{s.t.} \quad & b(x_\mathrm{k}) + \nabla b(x_\mathrm{k})^T d \geq 0 \\ \text{s.t.} \quad & c(x_\mathrm{k}) + \nabla c(x_\mathrm{k})^T = 0 \end{aligned} \tag{4-25}$$

which is iteratively solved by an Quadratic Programming (QP) solver.

4-6-4 Interior point

Interior point methods do also use the hessian. The algorithm is explained by assuming we are finding the minimum of a function $f(x)$ with inequality constraints [65], expressed by

$$\begin{aligned} \min_{\mathrm{x}} \quad & f(x) \\ \text{s.t.} \quad & g(x) \geq b \end{aligned} \tag{4-26}$$

These inequality constraints can written as

$$g(x) - b \geq 0, \tag{4-27}$$

such that they can be replaced by equality constraints and a simple inequality constraint using slack variables s.

$$\begin{aligned} g(x) - b - s &= 0, \\ s &\geq 0. \end{aligned} \tag{4-28}$$

The equality constraint are now rewritten as $c(x) = 0$. the minimization function can then be reformulated to create just an equality-constraint problem. This so-called barrier function is minimized and is described by

$$\begin{aligned} \min_{x} \quad & f(x) - \mu \sum_{\mathrm{i}=0}^{\mathrm{N}} \log(x_{\mathrm{i}}) \\ \text{s.t.} \quad & c(x) = 0 \\ & x \geq 0 \end{aligned} \tag{4-29}$$

where N is the number of original inequality constraints in Equation 4-26, and the barrier parameter μ converges to zero when approaching the solution of the problem. Note that the value of $\log(x_{\mathrm{i}})$ is not defined for $x \leq 0$, as required by Equation 4-28. Next to the 'primal' optimization variable x, as with the method of Lagrange multipliers, a 'dual' variable λ is introduced. The equation for the gradient can now be written as

$$\nabla f(x) + \lambda \nabla c(x) - \mu \sum_{i=0}^{N} \frac{1}{x_i} = 0 \tag{4-30}$$

Depending on the problem, very low computation times can be achieved. SQP and interior point are the preferred options for real-time optimization.

4-7 Hierarchical control

The computational complexity of a control problem can be decreased by using hierarchical control. The main problem can be explicitly split into 2 subproblems. This could be done as well for the integrated TMS problem, by choosing the actuator configuration and component temperatures in two steps instead of a single step, which is already done in the benchmark controller. Generally, first, setpoints are created at a low frequency in a high-level controller. The setpoints can be tracked at a high frequency in a low-level controller.

Split mode switching and temperature control

In [3], an electric motor cooling system is developed. The control strategy is visualized in Figure 4-6 Multiple actuator configurations exist to keep the motor temperature controlled. Here, based on the current component temperature and current heat generation in the motor, the discrete mode is chosen using a finite state machine. The target temperature is assumed fixed and in the second step tracked using the controller that belongs to the chosen mode. An approach using a fixed reference temperature, does not take into account that energy consumption can be decreased by allowing temperature variations over time.

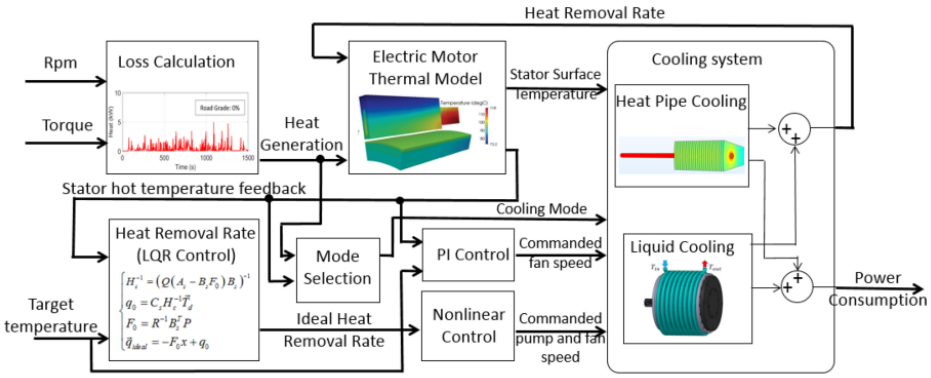


Figure 4-6: The control strategy presented in [3]

Split control of multiple components

An important variable to take into account is the timescale at which the temperatures of different components change. If the expected variation in temperature of certain components is a magnitude higher compared to other components, the control response is also allowed to be slower. A hierarchical controller can cover different dynamics (relatively fast and slow). [4] shows that the battery and cabin cooling thermal conditioning can be decoupled by controlling individual component temperatures separately, as visualized in Figure 4-7. This allows for easier implementation and simpler tuning. Here the component with slower dynamics can be controlled at a lower frequency than the component with faster dynamics.

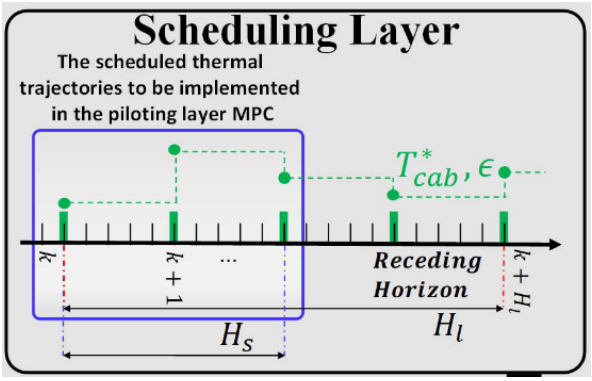


Figure 4-7: The battery temperature prediction horizon H_s is shorter than the cabin temperature prediction horizon H_L [4]

Planning and Tracking

Hierarchical control allows for using the benefits of different control strategies. A high-level controller can provide setpoints, for example, based on optimization. Low-level controllers ensure setpoint tracking and disturbance rejection [5]. Here the high accuracy of a high-fidelity model is combined with the good disturbances rejection properties of the fast

feedback controllers. In [5], the cooling capacity for a passenger cabin is controlled in the high-level controller, as visualized in Figure 4-8. The cooling capacity directly dictates the component temperature, which changes slowly. The low-level controller tracks this cooling capacity by controlling the actuators that need fast control.

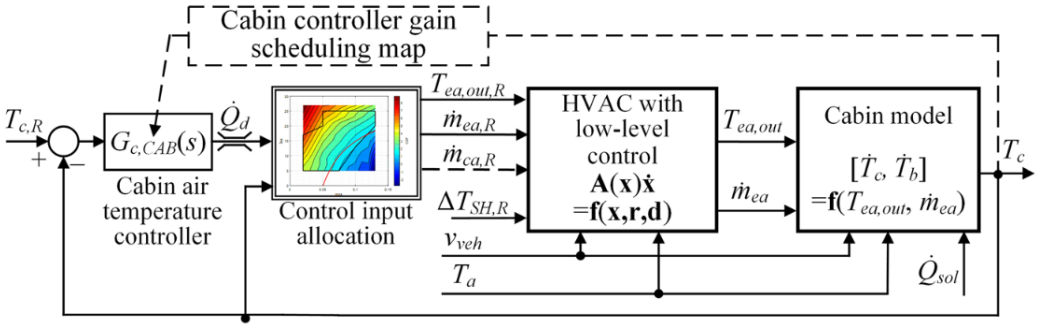


Figure 4-8: Control strategy combining optimization with feedback control. [5]

A similar approach for battery thermal management using a radiator only was taken in [6]. Optimal battery target temperatures are determined offline depending on the ambient temperature and the current drivecycle is not taken into account. After the target temperature is chosen, MPC tracks this temperature, based on a linearized model of the system, as seen in Figure 4-9.

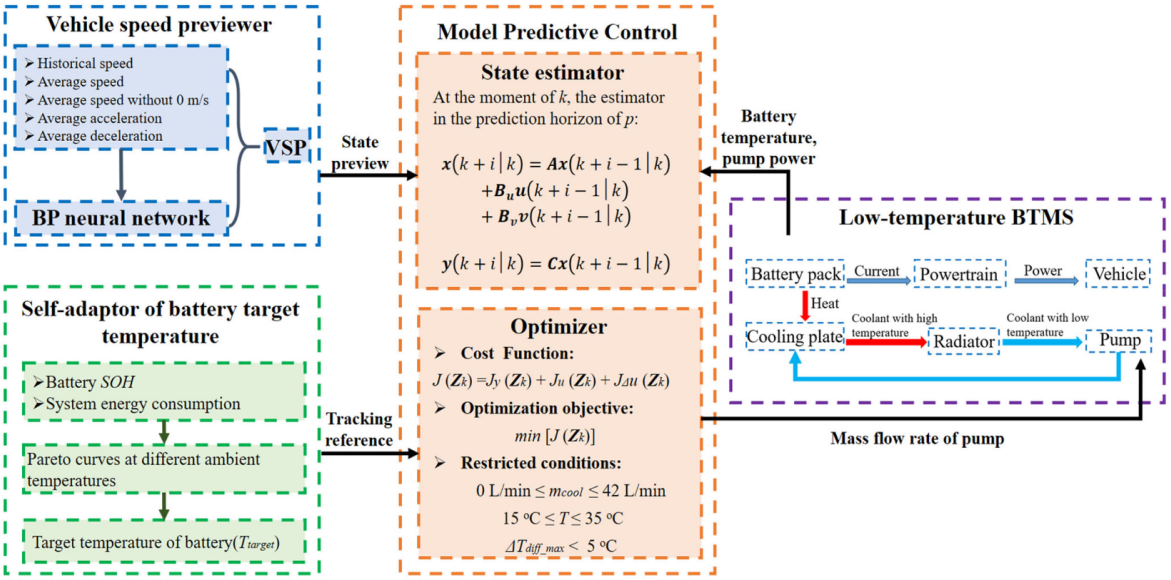


Figure 4-9: MPC is used for tracking [6]

Hierarchical control can be useful when using multiple sources of forecast information for the same prediction variables. In [4], cabin thermal management was split into two cascaded controllers that each use different forecast predictors. Vehicle speed is predicted accurately over the short term using vehicle-to-infrastructure and vehicle-to-vehicle information.

Traffic flow information gives a more accurate estimate in the long term. This was also done in [62]. Based on the idea that the forecasted speed is uncertain, they introduced "robust constraint tightening", where constraints in the first layer are tightened if the second layer, that uses a more accurate model, predicts constraint violations.

Similarly in [7], a first layer is developed to plan the battery temperature trajectory whilst forecasting the future vehicle load. By using a long horizon, battery temperature can be maintained within specified limits even in case of high peaks in future power demand. The second layer more accurately tracks the battery temperature and additionally, the cabin temperature, as visualized in Figure 4-10.

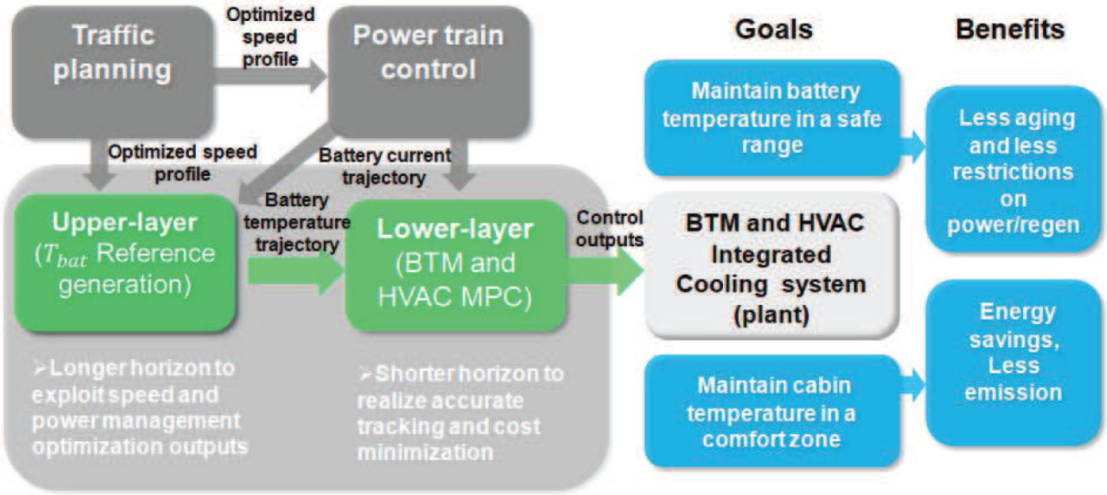


Figure 4-10: Different prediction horizons are used in cascased MPC control [7]

4-7-1 Hierarchical control for thermal management

If the model dynamics are understood well, hierarchical control allows for splitting the control problem by controlling slow-changing states at a slow rate and fast-changing states at a faster rate. This section elaborates on the speed of the dynamics in the TMS. Regarding the modelled states in the TMS, the 'speed' of the dynamics is depending on the maximum heat generation and the thermal inertia and can be summarized per component.

- Inverters: Fast dynamics. Can change significantly in a minute.
- Motors: Moderate dynamics. The losses are proportional to the vehicle torque. Can change significantly in several minutes.
- Battery: Moderate/slow dynamics. The losses are proportional to the vehicle torque. Temperature changes slowly due to the thermal capacity
- Coolant temperature: Fast/moderate dynamics, can change significantly in a minute.
- Refrigerant temperature/phase: Fast dynamics. Phase changes in the circuit need to be controlled at a high frequency

As mentioned in subsection 3-3-3, the refrigerant circuit dynamics can be simplified by directly relating the heat dissipation to the compressor power dissipation in the optimal controller. As the refrigerant dynamics are fast, the compressor and expansion valve are controlled at a higher frequency, saving computational power in the first layer. Similarly with the refrigerant temperature, the coolant temperature changes a lot faster than the motor, inverter or battery temperature. However, as the coolant directly exchanges heat with the components, ambient and refrigerant, modelling the coolant temperature is required for an accurate physical model.

Next to the states, the speed of variations is summarized for each of the modelled disturbances.

- Vehicle speed: Fast dynamics. Can change significantly in several seconds.
- Motor speed: Fast dynamics, proportional to vehicle speed
- Torque: Fast dynamics, proportional to vehicle speed squared at higher velocities.
- Battery current: Fast dynamics, proportional to the vehicle speed cubed at higher velocities.
- Ambient temperature: Slow dynamics. Significant changes generally take hours.

The vehicle speed, motor speed and torque, and battery current all have fast dynamics, so, therefore, it is important to have predictions for each timestep in the optimal controller. The ambient temperature changes slowly and therefore it can be modelled as a fixed value throughout the control horizon.

Chapter 5

Controller design

This chapter elaborates on the controller design for the Thermal Management System (TMS). First, the model dynamics are summarized, and then the sampling time, prediction horizon, integration method and optimization method chosen are discussed.

5-1 Prediction model

The system is modelled by differential equations, as described in chapter 3, where temperature dynamics are modelled as states:

$$x = \begin{pmatrix} T_{\text{bat}} \\ T_{\text{motor}} \\ T_{\text{inverter}} \\ T_{\text{coolant, chiller}} \\ T_{\text{coolant, battery}} \\ T_{\text{coolant, radiator}} \\ T_{\text{coolant, motors}} \\ T_{\text{coolant, inverters}} \end{pmatrix}, \tag{5-1}$$

the control inputs are denoted by:

$$u = \begin{pmatrix} w_1 \\ \dot{m}_1 \\ \dot{m}_2 \\ \dot{m}_{\text{air}} \\ \dot{Q}_{\text{chiller}} \end{pmatrix}, \tag{5-2}$$

and the disturbances are denoted by:

$$d = \begin{pmatrix} v_{\text{vehicle}} \\ T_{\text{ambient}} \\ \tau \\ \omega \\ I_{\text{battery}} \end{pmatrix}. \tag{5-3}$$

The differential state equations are nonlinear functions of the states, inputs and disturbances.

$$\dot{x} = f(x, u, d). \tag{5-4}$$

There are linear equality constraints g for the states and inputs.

$$g(x, u, d) \geq 0, \tag{5-5}$$

including the operating mode control input $u_1 = w_1$, for which the integer requirement is relaxed.

$$u_1 \in [0, 1]. \tag{5-6}$$

The objective to be minimized includes the ageing rate $\alpha_{\text{ageing}}(T)$, total losses and TMS power consumption, such that the Model Predictive control (MPC) is formulated as:

$$\begin{aligned} \min_{x,u} \quad &= \sum_{k=0}^{N_{\text{p}}-1} C_1(\alpha_{\text{ageing}}) + C_2(\dot{Q}_{\text{losses, total}} + P_{\text{consumed, total}})dt, \\ \text{s.t.} \quad & x(k+1) = f(x(k), u(k), d(k)), \\ & x(0) = x_0, \\ & g(x(k), u(k), d(k)) \geq 0, \end{aligned} \tag{5-7}$$

where N_p is the MPC prediction horizon and C_1 and C_2 are the weights for ageing and power consumption respectively..

5-2 Controller structure

Hierarchical control allows for planning the trajectory of slow-changing temperatures far ahead, whilst tracking fast-changing elements at a high rate and saving computational power. The supervisory MPC controls the slow-changing temperature states, x by taking into account forecast information d, as visualized in Figure 5-1, whereas simple PID-based controllers are used to track the setpoints generated u by regulating the chiller heat dissipation and the coolant and air mass flow rates. The plant outputs used in the PID-based controller are denoted by y_2, where the control actions, the pump speeds, fan speed, compressor speed and expansion valve positions, are denoted by u_2.

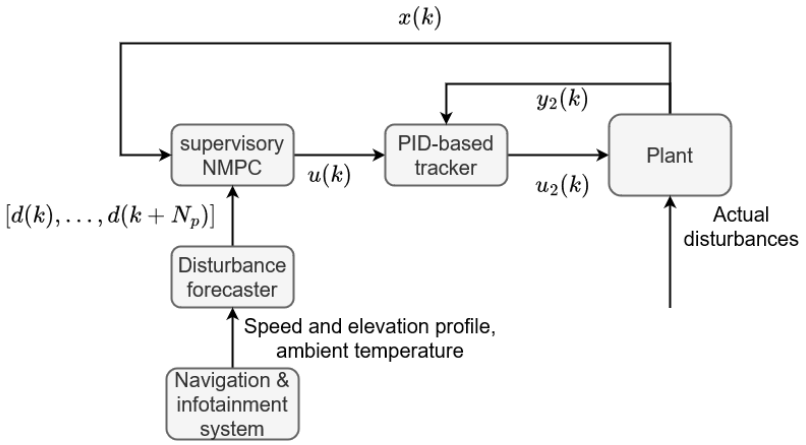


Figure 5-1: NMPC-based control structure

5-3 Sampling time

Based on the in-vehicle measurement rate, the low-level pump and compressor controllers are limited to running at a frequency f_1 of 2 Hz. Therefore, the frequency of the high-level controller should run at $f_2 < 0.1 f_1$ which means at 0.2 Hz or slower, to ensure the PID-based controllers are able the track the setpoints generated in the high-level controller. The temperature of the inverters can change significantly in several seconds. To ensure that the temperature will never exceed the temperature limits, shown in Table 2-1, the inverter can gradually reduce its power output to decrease heat production. Instead, being able to react to disturbances is preferred such that the inverter temperature can be decreased through cooling, which requires a low sampling time.
To ensure the desired disturbance rejection, the minimum sampling time of $T_s = \frac{1}{0.1 f_1} = 5$ s is chosen.

5-4 Integration

The integrator is responsible for solving the Ordinary Differential Equation (ODE)'s to simulate the model dynamics over time. Following the background given in chapter 4, different numerical ODE solvers are compared on two criteria: Computation time and accuracy. The computation time is evaluated in combination with the optimizer, which will be discussed in chapter 7.

Backward Differentiation Formula (BDF) method The Backward Differentiation Formula (BDF) method is an implicit, adaptive solver that allows for an error controlled solution that ensures numerical stability and accuracy for stiff and non-stiff systems.

Explicit fixed-step Runge-Kutta method (RK4) The RK4 method is an explicit and fixed-step method, meaning that the computation time is bounded. The step size must be chosen such that the desired accuracy in the integration is ensured.

5-4-1 Integration stepsize

While the BDF uses an adaptive stepsize to control the error, the RK4 integration method uses a fixed step size h. The Local Truncation Error (LTE), the error made in a single integration step, is proportional to h^5. The step size is chosen by evaluating the LTE on each of the inputs and initial states, obtained from the benchmark controller results that were also used for model validation in section 6-4. An error-controller integrator with a relative error tolerance RelTol = 1e-12 and absolute error tolerance AbsTol = 1e-20 is used as a benchmark integrator. The benchmark integrator is compared to the RK4 integrator with various step sizes. The integrator with stepsize $h = 0.5$ gives a maximum error of LTE=1e-6, which is assumed sufficient and therefore is chosen for the MPC controller.

5-5 Optimization

The multiple shooting method is used to formulate the Nonlinear Programming (NLP) method. Both Sequential Quadratic Programming (SQP) and interior point methods are suitable solvers for real-time problem-solving. Various available optimizers are compared in combination with various integrators by evaluating the achieved performance of the control strategy and computation time in chapter 7.

Tested solvers Many solvers are available. Casadi [66] is an open-source tool for nonlinear optimization and algorithmic differentiation. Within Casadi, various solvers for NLP problems are available, both SQP methods and interior point methods. Casadi was also used in [67] and [68].

Finally, Forcespro [69] is a commercial optimization framework that allows using branch and bound methods to solve mixed-integer problems, as well as interior point methods specifically designed for real-time MPC.

These methods will be compared with each other and the benchmark controller.

Chapter 6

System identification and model validation

System identification is done within the high-fidelity Simscape simulation environment, that has been developed at Lightyear. The simulation captures many more physical effects than captured in the control model described in chapter 3. Some of the main differences are:

- Most components are parametrized by tabular data retrieved from component datasheets or system identification experiments, instead of basic physical equations that may not truly capture the component behaviour.
- All four motors and inverters are modelled instead of just one.
- The battery, motors and inverters are modelled by multi-node models, instead of single-node models.
- The refrigerant circuit is modelled completely, whereas in the control model, the relation between compressor power consumption and heat dissipation in the chiller is modelled by the Coefficient Of Performance (COP) only.

By using only measurement data from the simulation that can also be obtained on the real vehicle, the experiments are designed such that the same methods can also be executed on a real vehicle.

6-1 Thermal model parameters

The thermal capacities are given, based on their materials. The thermal capacity of the coolant around a component is simply estimated by multiplying the coolant mass at room temperature around that component by the specific heat coefficient. The symbolic thermal capacities are given in Table 6-1.

This section elaborates on the system identification of the heat exchange model parameters.

Component	Thermal Capacity
Motor	C_{motor}
Inverter	C_{inverter}
HVB	C_{HVB}
Coolant around motor	$m_{\text{coolant, motor}} c_{\text{coolant}}$
Coolant around inverter	$m_{\text{coolant, inv}} c_{\text{coolant}}$
Coolant around HVB	$m_{\text{coolant, hvb}} c_{\text{coolant}}$
Coolant around radiator	$m_{\text{coolant, rad}} c_{\text{coolant}}$
Coolant around chiller	$m_{\text{coolant, chiller}} c_{\text{coolant}}$

Table 6-1: Thermal capacities used

6-1-1 Methodology

This section describes the procedures followed to estimate the heat transfer coefficients for the different heat exchangers. Summarizing the equations given in section 3-2, the heat transfer $\dot{Q}_{\text{dissipated}}$ between components and the coolant is described by

$$\dot{Q}_{\text{dissipated}} = \epsilon \dot{m}_{coolant} c_{\text{coolant}} (T_{\text{component}} - T_{\text{coolant}}), \tag{6-1}$$

whereas the heat transfer efficiency ϵ is defined as

$$\epsilon(\dot{m}_{\text{cool}}) = 1 - \exp(-\text{NTU}), \tag{6-2}$$

and NTU is defined as

$$\text{NTU} = \frac{UA}{\dot{m}_{coolant} c_{coolant}}. \tag{6-3}$$

The overall heat transfer coefficient, multiplied by the heat transfer surface UA is modelled by

$$UA = a \dot{m}_{\text{coolant}}^{b}, \tag{6-4}$$

where a and b are the parameters to be identified.

We assume the following data is measured for each test. The component temperature for the battery, motor or drivetrain is generalized below and denoted by $T_{\text{component}}$.

- Component temperature $T_{\text{component}}$.
- Mass flow rate of coolant $\dot{m}_{\text{coolant}}$.
- Temperature of the coolant at the inlet $T_{\text{coolant, inlet}}$ and outlet $T_{\text{coolant, outlet}}$ of the respective heat exchanger.

The following procedure is used to find heat transfer parameters:

System identification procedure: Component dissipation to coolant

1. Bring component up to a high-temperature such that $T_{\text{comp}} >> T_{\text{cool}}$

2. Gather component temperature data and additional coolant inlet and outlet temperature data over time at different coolant mass flow rates and estimate heat exchange between the coolant and the component.
3. The heat dissipation to the coolant is estimated as follows: $\dot{Q}_{\text{dissipation, coolant}} = (T_{\text{coolant, outlet}} - T_{\text{coolant, inlet}})\dot{m}_{\text{coolant}}c_{\text{coolant}}$
4. Fit Number of Transfer Units (NTU) heat exchange model parameters to the data using the MATLAB fit function.

At the radiator, the coolant mass flow and the ambient air mass flow exchange heat through the double-stream heat exchanger. The heat transfer is defined by

$$\dot{Q}_{\text{dissipated, radiator}} = \epsilon C_{\text{r}}(T_{\text{coolant}} - T_{\text{ambient}}), \tag{6-5}$$

where the minimum heat capacity rate C_{r} is described by

$$C_{\text{r}} = \min(\dot{m}_{\text{coolant}}c_{\text{coolant}}, \dot{m}_{\text{ambient}}c_{\text{ambient}}). \tag{6-6}$$

For the double-stream heat exchange, ϵ is defined as

$$\epsilon(\dot{m}_{\text{coolant}}) = 1 - \exp\frac{\exp -C_{\text{r}}\text{NTU}^{0.78} - 1}{C_{\text{r}}\text{NTU}^{-0.22}}. \tag{6-7}$$

The definition for the NTU is the same as for the single-stream heat exchanger as described in Equation 6-3. The overall heat transfer coefficient, multiplied by the heat transfer surface, UA is now expressed by

$$UA = \frac{1}{\frac{1}{a_1\dot{m}_{\text{coolant}}^{b_1}} + \frac{1}{a_2\dot{m}_{\text{ambient}}^{b_2}}}, \tag{6-8}$$

where a_1, a_2, b_1 and b_2 are the parameters to be identified. To avoid using a minimum in the optimization, the direct use of the UA, such that

$$\dot{Q}_{\text{dissipated, radiator}} = UA(T_{\text{coolant}} - T_{\text{ambient}}), \tag{6-9}$$

is compared to the equation using the NTU method as described in Equation 6-1-1. For the radiator experiments, we assume the following data is measured:

- Mass flow rate of the coolant $\dot{m}_{\text{coolant, radiator}}$.
- Mass flow rate of the ambient air at the radiator $\dot{m}_{\text{ambient}}$.
- Temperature of the coolant at the inlet $T_{\text{coolant, inlet}}$ and outlet $T_{\text{coolant, outlet}}$ of the radiator.
- Temperature of the air at the inlet, equal to the ambient temperature, T_{ambient} and outlet of the respective heat exchanger $T_{\text{air, outlet}}$.

The following procedure is used for the radiator:

System identification procedure: Coolant to air at the radiator

1. Ensure there is a temperature difference between the coolant and the ambient air such that $T_{\text{comp}} \neq T_{\text{cool}}$
2. Gather coolant inlet and outlet temperature and additionally ambient air temperature data over time at different combinations of coolant mass flow rates and air mass flow rates and estimate heat exchange between the coolant and the air.
   $\dot{Q}_{\text{dissipation, coolant to air}} = (T_{\text{coolant, outlet}} - T_{\text{coolant, inlet}})\dot{m}_{\text{coolant}} c_{\text{coolant}}$
3. Fit NTU heat exchange model parameters to the data using the MATLAB fit function.

6-1-2 Results of system identification

The heat exchange coefficients are identified for the electrical components and radiator. Complete results including each of the electrical components are shown in section A-1.

Electrical components

For each of the electrical components, the heat transfer at different mass flow rates was measured. The heat transfer equation parameters a and b were fitted to the heat transfer equation, resulting in the values presented in Table 6-2, whereas the results for the High Voltage Battery (HVB) are shown in Figure 6-1 for visualization.

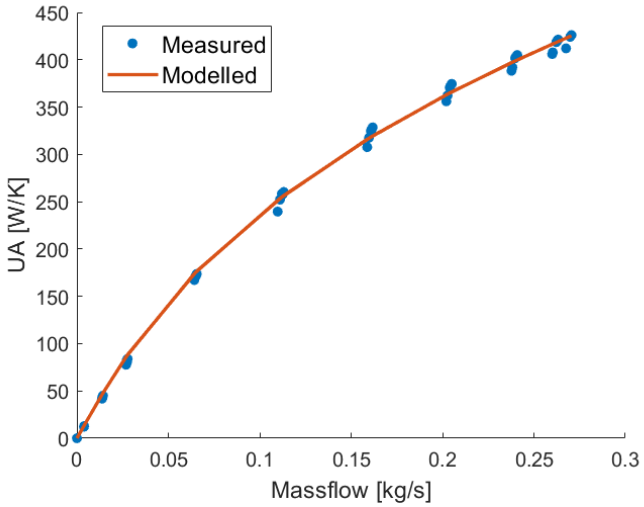


Figure 6-1: Battery to coolant heat transfer coefficient

Radiator

For the radiator, the heat transfer at different mass flow rates of the ambient air and coolant was measured. The approach using the the NTU method does give the most

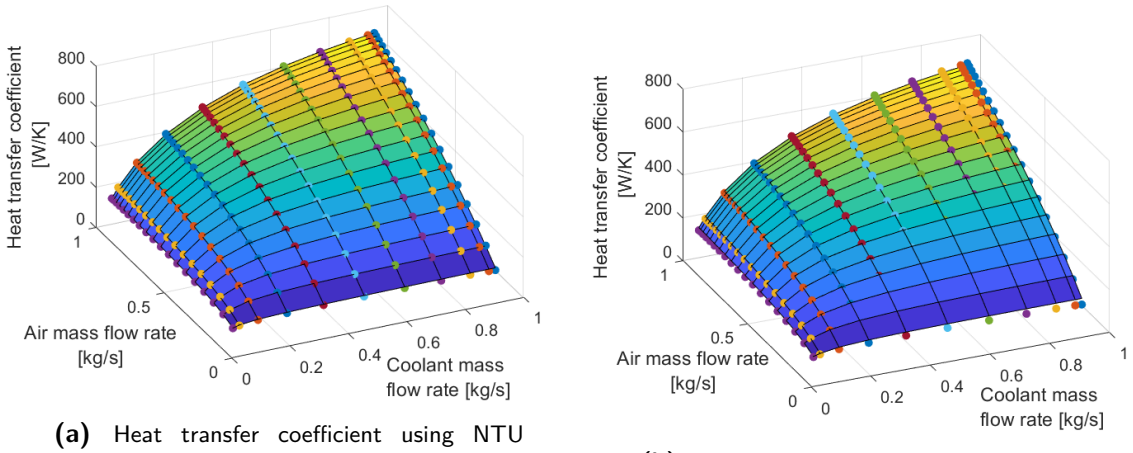
	Battery	Inverters	Motors
a	955.5	220	75.35
b	0.40	0.79	0.24

Table 6-2: Resulting heat transfer parameters from system identification experiments

accurate results, as shown in Figure 6-2a. Figure 6-2b shows the results of the model using the heat transfer coefficient directly, which prevents using a discontinuous minimum function in the optimization problem and is therefore chosen. At lower mass flow rates, the heat transfer coefficient is slightly underestimated. The values for the heat transfer parameters are shown in Table 6-3.

	Radiator using NTU method	Radiator using UA directly
a_1	2279	937
b_1	0.8	0.8
a_2	3873	2614
b_2	0.79	0.8

Table 6-3: Resulting radiator heat transfer parameters from system identification experiments



(a) Heat transfer coefficient using NTU method

(b) Heat transfer coefficient directly using UA

Figure 6-2: Comparison of two models for the heat transfer coefficient at the radiator. The dots denote the measured data, whereas the surface plot is made using the model.

6-2 Hydraulic model parameters

This section elaborates on the methodology and results of the system identification of the hydraulic model parameters.

6-2-1 Methodology

For the model parameters related to the coolant fluid, ambient air and refrigerant, first the used models are summarized, and then the experimental procedure is described.

Ambient air mass flow rate

Summarizing the equations described in section 3-3, using the affinity laws, the fan power P_{fan} is related to the volumetric flow rate $\dot{V}_{\text{air}}$ by

$$P_{\text{fan}} = a_{\text{fan}} \dot{V}_{\text{air}}^3, \tag{6-10}$$

where a is the parameter to be fitted. A linear relationship is assumed between the air volumetric flow rate $\dot{V}_{\text{air}}$, when the fan is turned off, and vehicle speed v_{car}.

$$\dot{V}_{\text{air}} = a_{\text{car}} v_{\text{car}}, \tag{6-11}$$

where a_{car} needs to be found through experiments.

We assume the following data is measured:

- Vehicle speed v_{vehicle}
- Mass flow rate of air at the radiator $\dot{m}_{\text{air}}$ that can be converted to the volumetric flow rate $\dot{V}_{\text{air}}$ through the air density.
- Fan power consumption P_{fan}

The following procedure is executed to find the model parameters required to model the ambient air mass flow rate:

System identification procedure: Ambient air mass flow rate

1. Vary the vehicle speed between $v_{\text{vehicle}} = 0$ and $v_{\text{vehicle}} = v_{\text{vehicle, max}}$ and for each speed, measure V_{air}. Maintain each speed for step 2, after which the speed can be increased.
2. Vary the fan speed between the 0 and 100% and measure the volumetric flow rate $\dot{V}_{\text{air}}$ and fan power consumption P_{fan}.
3. Fit affinity law model at $v_{\text{vehicle}} = 0$ using the MATLAB fit function.
4. Fit volumetric flow rate to vehicle speed at $P_{\text{fan}} = 0$ using MATLAB fit function.

Coolant mass flow rate

For the pumps, a similar relationship can be used to describe the pump power consumption P_{pump}, but first and second-order terms are added to capture nonlinearities.

$$P_{\text{pump}} = a_1 \dot{V}^3_{\text{coolant}} + a_2 \dot{V}^2_{\text{coolant}} + a_3 \dot{V}_{\text{coolant}}, \tag{6-12}$$

where $\dot{V}_{\text{coolant}}$ is the volumetric flow rate of the coolant and the remaining parameters a_1, a_2 and a_3 have to be identified for the different modes. When $w_1 = 1$, the drivetrain components and battery are both cooled through the radiator, there is an interaction between the two circuits, requirements an additional term.

$$P_{\text{pumps, } w_1=1} = f(\dot{V}_{\text{coolant, 1}}) + f(\dot{V}_{\text{coolant, 2}})) + c\dot{V}_{\text{coolant, 1}}\dot{V}_{\text{coolant, 2}}, \tag{6-13}$$

where $f(\dot{V}_{\text{coolant, 1}})$ and $f(\dot{V}_{\text{coolant, 1}})$ are the power consumption when the other pump is turned off, as described by Equation 6-12 and c needs to be identified.

The following data is measured in order to execute the system identification procedure:

- Coolant mass flow rate at pump 1 $\dot{m}_{\text{coolant, 1}}$ and at pump 2 $\dot{m}_{\text{coolant, 2}}$ that can be related to the volumetric flow rate through the coolant density
- Valve position w_1
- Power consumption of pump 1 $P_{\text{coolant, 1}}$ and pump 2 $P_{\text{coolant, 1}}$.

The following procedure is executed to find the model parameters relating the coolant mass flow rate to the pump power consumption, for each of the modes $w_1 = 0$ and $w_1 = 1$.

System identification procedure: Coolant mass flow rate

1. Vary the pump speeds between 0 and 100% and measure the mass flow rate $\dot{m}_{\text{coolant, 1}}$, $\dot{m}_{\text{coolant, 2}}$ and pump power consumptions $P_{\text{pump, 1}}$ and $P_{\text{pump, 2}}$.
2. Fit the model for each pump and each mode individually, as described by Equation 6-12 using the MATLAB fit function.
3. Fit the complete model, including the model parameter c, as described in Equation 6-13 using the MATLAB fit function.

Chiller

To summarize the model equations given in subsection 3-3-3, let the Coefficient Of Performance (COP) be expressed by

$$\text{COP} = \frac{\dot{Q}_{\text{dissipated, chiller}}}{P_{\text{compressor}}}, \tag{6-14}$$

where $\dot{Q}_{\text{dissipated, chiller}}$ and $P_{\text{compressor}}$ are the heat dissipated at the chiller and power consumption of the compressor.

We assume the following data is measured:

- Coolant mass flow rate at the chiller $\dot{m}_{\text{coolant, 1}}$
- Ambient air mass flow rate $\dot{m}_{\text{air}}$
- Coolant temperature at the chiller inlet, denoted as $T_{\text{coolant, inlet}}$
- Coolant temperature at the chiller outlet, denoted as $T_{\text{coolant, outlet}}$
- Ambient temperature T_{ambient}
- Power consumption of the compressor $P_{\text{compressor}}$

We assume a test matrix where the COP at all possible combinations shown in Table 6-4 are evaluated such that the COP at all realistic use cases can be estimated.

	Possible values
T_{amb}[°C]	[20, 30, 40]
$T_{\text{coolant, inlet}}$[°C]	[20, 30, 40]
n_{pump}[%]	[25, 50, 75, 100]
v_{vehicle}[km/h]	[30, 50, 70, 90, 120]
$\dot{Q}_{\text{dissipated, chiller}}$[W]	[1000, 2000, 3000]

Table 6-4: Chiller COP test parameter settings

System identification procedure: Chiller cooling power

1. Iteratively set the initial conditions according to the upper use case. The heat dissipation is controlled by a simple PID controller.
2. Wait for the low-level tracking controller to reach the desired heat dissipation.
3. Measure the power consumption of the compressor.

6-2-2 Results of system identification

This section shows results for the hydraulic model identification. Complete results for each pump in each valve position are shown in section A-2.

Ambient air mass flow rate

The parameter a_{car}, relating the vehicle speed and resulting air volumetric flow rate is determined first. Then the affinity law parameter a_{fan} relating the volumetric flow rate to the fan power consumption is estimated. The found values are shown in Table 6-5 and the resulting fitted model is visualized in Figure 6-3.

It was observed that at higher speeds when the fan is turned, increased fan speed would actually lower the mass flow rate. Because of this, correct lower and upper mass flow rate

Parameter	Value
a_{car}	0.008
a_{fan}	1700

Table 6-5: Resulting hydraulic model parameters from system identification experiments

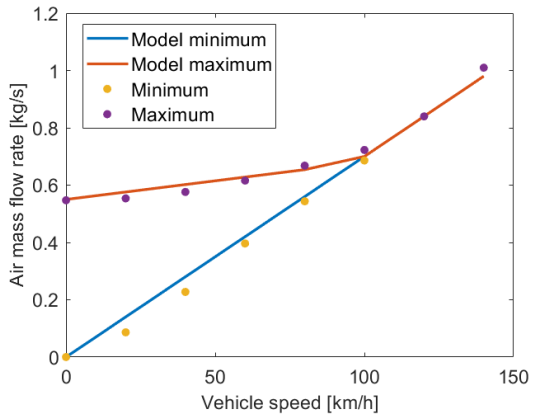
limits at different velocities were determined. The lower bound $\dot{m}_{\text{air, min}}$ is straightforward, as it is the mass flow rate induced by the vehicle speed.

$$\dot{m}_{\text{air, min}} = \dot{m}_{\text{air, natural}}. \tag{6-15}$$

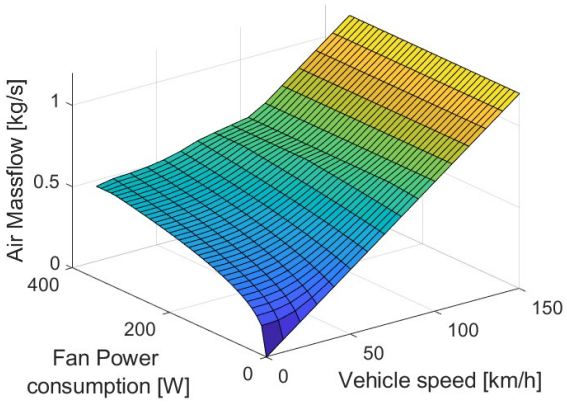
The upper limit $\dot{m}_{\text{air, max}}$ depends on specific fan characteristics at lower speeds, as shown in Figure 6-3a. A linear relation between the vehicle speed and upper limits is drawn. This limit at lower speeds is denoted by $\dot{m}_{\text{air, max, fan}}$. At higher speeds, the fan is not able to increase the mass flow anymore and the mass flow is proportional to the vehicle speed. Now, the resulting limit are simply expressed by

$$\dot{m}_{\text{air, max}} = \max(\dot{m}_{\text{air, max, fan}}, \dot{m}_{\text{air, natural}}). \tag{6-16}$$

The graph Figure 6-3b shows how the mass flow rate changes at different fan - vehicle speed combinations, taking into account the limits.



(a) Ambient air mass flow limits at different velocities



(b) Ambient air mass flow rate at different velocities and fan speeds

Figure 6-3: The relation between the ambient air mass flow rate, the vehicle speed and fan power consumption is identified.

Coolant mass flow rate

The coolant mass flow rate does not only depend on the pump speed but also on the valve positions, as different components cause different pressure drops. First, the relation between pump speed and mass flow rate was determined assuming only one of the two pumps was turned on. After this, both pumps were both turned on to capture the interactive term. The resulting model parameters are shown in Table 6-6, whereas the results for pump 1 and $w_1 = 1$ are plotted in Figure 6-4. For the interactive term, a value of c = 1621 was found.

An example of the fitted model can be seen below for the short circuit mode. Adding a first order allows for increased accuracy in the low mass flow rate section.

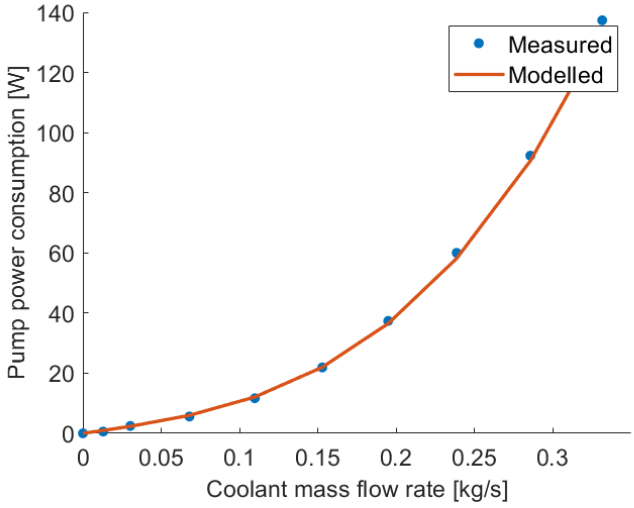


Figure 6-4: Power consumption related to coolant mass flow for pump 1 and w_1=1

	Pump 1, w1 = 0	**Pump 1, w1=1**	**Pump 2, w1=0**	**Pump 2, w1=1**
a_1	3841	2981	3076	2955
a_2	322.2	0	66.1	0
a_3	11.8	74.3	0	15.5

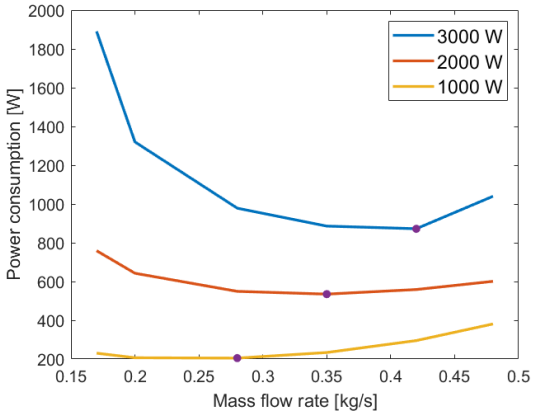
Table 6-6: Resulting model parameters relating the mass flow rate to the mass flow rate

Coefficient Of Performance (COP)

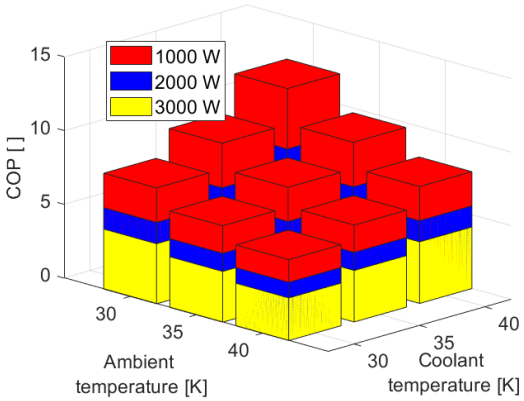
First, the influence of the mass flow rate on the Coefficient Of Performance (COP) is investigated. up to a certain point, higher mass flow rates cause a decrease in power consumption, thus increasing the COP, as can be seen in Figure 6-5a for the ambient air mass flow rate. At a certain point, an increase in mass flow rate costs too much energy and the combined power consumption of the compressor and fan increases. For each combination of ambient temperature, coolant temperature and desired heat dissipation, an energy-optimal air and coolant mass flow rate can be chosen using Look-Up Table (LUT)'s.

To conclude, the influence of coolant temperature, ambient temperature and cooling power are evaluated, as visualized in Figure 6-5b . The following choices are made:

- The ambient temperature changes very slowly, so a fixed COP with relation to the ambient temperature is assumed at every function evaluation.
- The coolant temperature can change fast, but it reaches a steady-state temperature based on the current battery temperature. Based on experiments, a fixed COP with relation to the current coolant temperature is assumed at every function evaluation. A



(a) Combined power consumption for the fan and compressor for different air mass flow rates for an ambient and battery temperature of 40 °C. The circles represent the optimal mass flow rate for each desired heat dissipation.



(b) Chiller COP for different temperatures and cooling rates

Figure 6-5: Chiller COP for different operating conditions

coolant temperature of five degrees lower than the current battery temperature is assumed.

- There is no single physical equation describing the relation between $\dot{Q}_{\text{desired}}$ and the COP. However, because the battery temperature changes slowly, also a slow change in optimal cooling demand can be expected. Therefore at every function evaluation, a fixed COP with relation to the previous cooling demand is assumed. To ensure no sudden fluctuations in the optimal control action, a low-pass filter is required. This is implemented by constraining the chiller heat dissipation to only change by a limited amount every timestep compared to the previously chosen heat dissipation.

This method allows for using simply a LUT to estimate the COP for every new function evaluation.

6-3 Electrical model parameters

The system identification process for the models describing the losses of the electrical components used for propulsion, the motors, inverters and HVB are discussed in this section.

6-3-1 Motors

Summarizing the equations described in subsection 3-4-4, the motor losses $\dot{Q}_{\text{losses, motor}}$ are described by

$$\dot{Q}_{\text{losses, motor}} = f(\omega, \tau) + a\tau^2(1 + \alpha(T_{\text{motor}} - T_0)), \tag{6-17}$$

where $f(\omega, \tau)$ are the non-winding losses, a needs to be identified and α is the copper temperature resistivity coefficient. The non-winding losses can be stored using a LUT.

The following data is used to estimate the losses of the motors:

- Total motor losses and phase currents I_{phase} at various torque and speed setpoints, at known temperature T_{motor}.
- Measured phase resistance R_{phase} at initial temperature T_{init}

From this data, the goal is to model the temperature-dependent losses at different speed and torque setpoints. The procedure to find the parameters of the temperature-dependent winding losses is described as follows:

System identification procedure: Motor losses

1. Calculate winding losses for all three phases using $\dot{Q}_{\text{losses, winding}} = 3I^2_{\text{phase}}R_{\text{phase}}$.
2. Fit winding losses to torque setpoints using $\dot{Q}_{\text{losses, winding}} = a\tau^2(1 + \alpha(T_{\text{motor}} - T_{\text{init}})$.
3. Winding losses can now be modeled as $\dot{Q}_{\text{losses, winding}} = a\tau^2(1 + \alpha(T_{\text{motor}} - T_{\text{init}})$.

This allows splitting the losses into temperature-dependent winding losses and non-temperature-dependent losses, such that the winding losses can be shown in Figure 6-6. The non-winding losses are given in section A-3.

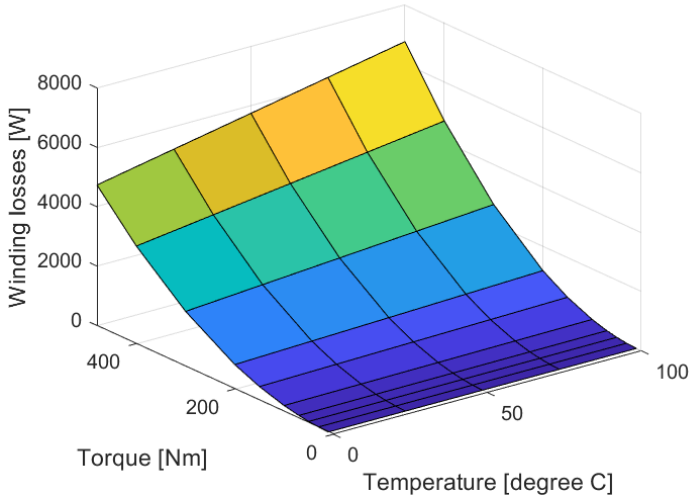


Figure 6-6: Motors winding losses as a function of torque and temperature

6-3-2 Inverter

Based on the available models, the temperature-dependent inverter losses are assumed negligible, as well as the speed-dependent losses. A polynomial model is available relating the torque and the inverter losses that is used in the remainder of this thesis. The inverter losses are given in section A-3.

6-3-3 HVB

Summarizing the equations described in , the battery losses $\dot{Q}_{\text{losses, battery}}$ are expressed by

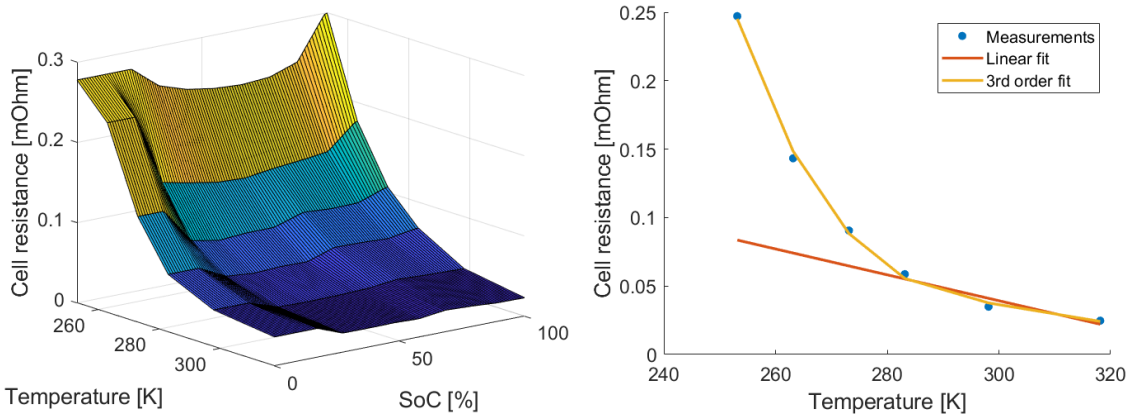
$$\dot{Q}_{\text{losses, battery}} = I_{\text{bat}}^2 R_{\text{bat}}(T_{\text{bat}}). \tag{6-18}$$

where R_{bat} is the battery resistance, depending on the temperature T_{bat}.

$$R_{\text{bat}} = a - bT_{\text{bat}}. \tag{6-19}$$

The calculate the losses in the HVB, the following data was available:

- Battery cell internal resistance for discharging, as a function of temperature and State of Charge (SoC). Data for charging is also available, but as the car is mostly driving and the cell resistance for charging is similar compared to discharging, only discharging data is used.



(a) Battery cell resistance as a function of SoC and temperature

(b) Battery cell resistance as a function of only temperature

Figure 6-7: The battery cell resistance depends on both the SoC as well as the temperature.

The cell resistance is shown in Figure 6-7a, where it can be observed that the SoC indeed has a very small effect when larger than 20 %, as was discussed in subsection 3-4-3. By removing the dependence on SoC, a nonlinear relationship with temperature can be seen. When only looking at battery temperatures above 25 °C, the region considered in this thesis, a linear relationship can be drawn, as shown in Figure 6-7b. The values for parameters a and b in Equation 6-19 are shown in Table 6-7.

Parameter	Value
a [Ω]	0.065
b [$\Omega \cdot$ K^{-1}]	0.95 $\cdot 10^{-3}$

Table 6-7: Resulting model parameters relating the battery temperature to the cell resistance

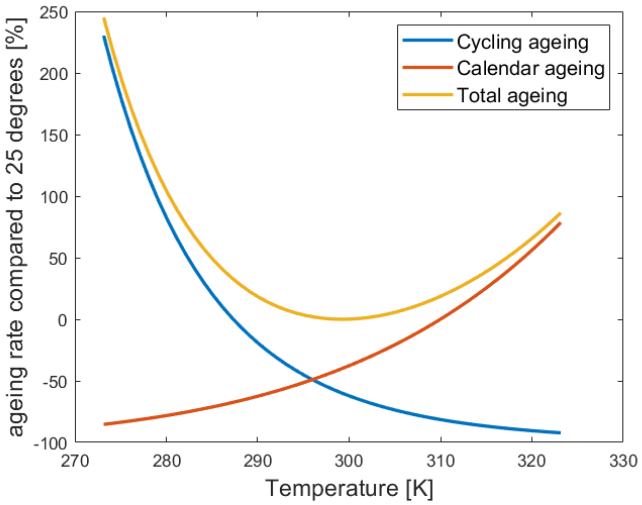


Figure 6-8: Ageing rate at different temperatures

Ageing As both mechanisms of cycling ageing and calendar ageing need to be taken into account, we assume the model including model parameters explained in [46], as discussed in subsection 3-4-3. The ageing behaviour of cycled cells is tested between -20 °C and +70 °C. Between 25 and 70 °C, a clear trend of rising ageing rates with increasing temperature was found. On the other hand, between -20 and 25 °C, higher ageing with lower temperatures was found. This shows that two different ageing mechanisms take place. The ageing rates for low and high temperatures are expressed as a percentage compared to ageing at 25°C, as shown in Figure 6-8. When adding the Arrhenius plots, we find a convex function. Here the total ageing rate $\alpha_{\text{ageing, total}}$ is defined as the sum of cycling and calendar ageing and expressed by

$$\alpha_{\text{ageing, total}} = \alpha_{\text{ageing, cycling}} + \alpha_{\text{ageing, calendar}}. \tag{6-20}$$

A polynomial fit is done which can be used in the optimization:

$$\alpha_{\text{ageing, total}} = a_1 T_{\text{Bat}}^4 + a_2 T_{\text{Bat}}^3 + a_3 T_{\text{Bat}}^2 + a_4 T_{\text{Bat}} + a_5. \tag{6-21}$$

The resulting model parameters are shown in Table 6-8.

Parameter	Value
a_1	0.0001268
a_2	-0.01728
a_3	1.004
a_4	-25.89
a_5	240.8

Table 6-8: Resulting model parameters relating the battery temperature to the ageing rate

6-4 Validation of the integrated model

To validate the integrated model, the following procedure was done:

System validation procedure: Integrated model

1. Generate control input data using the benchmark controller in the TMS simulation environment for different drive cycles and different initial temperatures.
2. Feed generated control inputs to the MPC model.
3. Compare component temperatures over time.

Various use cases were used to validate the complete system model.

6-4-1 Use cases cycles

To evaluate the effect of using forecast information, various drive cycles were chosen with different characteristics. The drive cycles are called "Stelvio" drivecycle, "Artemis" drive cycle and the Worldwide Harmonised Light Vehicle Test Procedure (WLTP) drive cycle. Also, tests are done for different ambient condtiions.

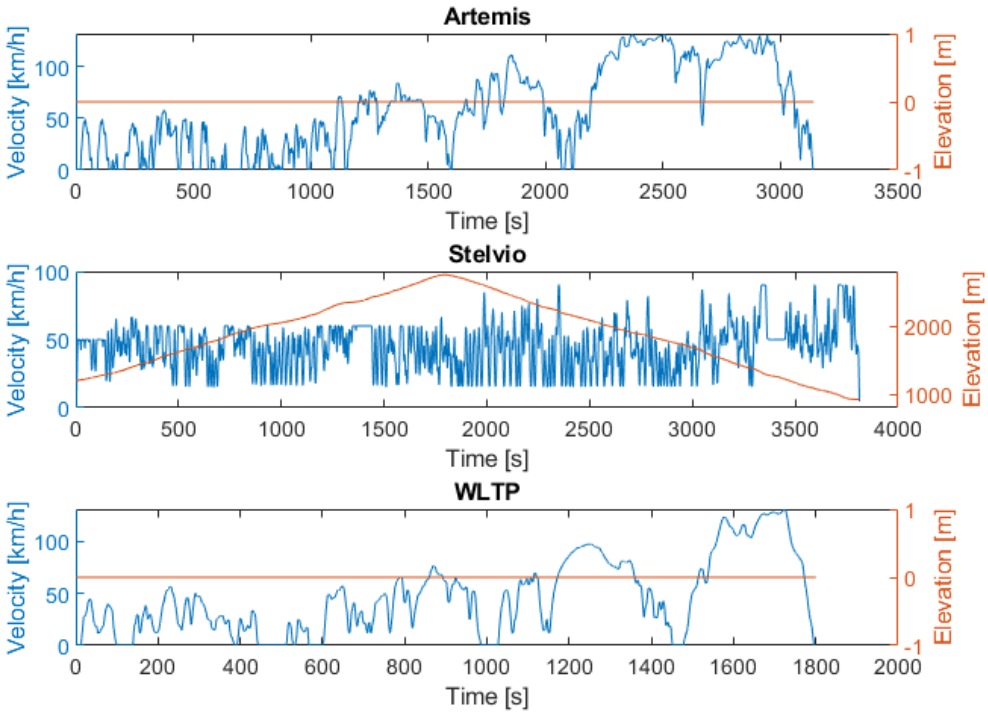


Figure 6-9: Velocity/Elevation profiles of the drivecycles

Stelvio drive cycle The vehicle is going up and down the Stelvio mountain. This drive cycle has a high average torque and a low average speed. The ambient temperature varies due to the elevation profile.

	Average torque[Nm]	Max torque[Nm]	Average speed[km/h]	Duration[s]
Artemis	258	1673	58	3400
Stelvio	529	1690	44	3815
WLTP	216	1039	46	1800

Table 6-9: Drive cycle properties

Artemis drive cycle This drive cycle has a moderate average torque and a varying speed.

WLTP drive cycle This drive cycle has a moderate average torque and a varying speed. WLTP is the standard drive cycle for vehicle efficiency testing and has the shortest duration.

Ambient conditions The controller is tested for two cases, to examine the effect of ambient temperature. The initial component temperature is assumed to be five degrees above the ambient temperature.

- Moderate weather: the ambient temperature is 20 degrees
- Hot weather: the ambient temperature is 40 degrees

6-4-2 Results

To show the validity of the integrated model, example input data was generated using the reference controller. For the drive cycle WLTP, the example inputs generated from the baseline controller are fed into the MPC model. The true component behaviour is compared to the simulated behaviour over the same inputs as can be seen in Figure 6-10. The integration time is over the entire drive cycle, whereas during real-time control, only several minutes of prediction is done so error accumulation should be lower. Complete validation results including results obtained over the remaining use cases are shown in section A-4.

The mean error and standard deviations in the error of the electric components are shown in Table 6-10 for each of the drive cycles.

	$\mu_{\epsilon,HVB}$ [K]	$\sigma_{\epsilon,HVB}$ [K]	$\mu_{\epsilon,mot,max}$ [K]	$\sigma_{\epsilon,mot,max}$ [K]	$\mu_{\epsilon,inv,max}$ [K]	$\sigma_{\epsilon,inv,max}$ [K]
WLTP moderate	-0.14	0.01	1.40	1.06	-0.75	1.62
WLTP hot	-0.16	0.03	0.92	3.01	0.20	2.16
Artemis moderate	0.25	0.10	0.78	0.99	-0.79	1.16
Artemis hot	0.21	0.09	0.14	1.51	0.00	0.98
Stelvio moderate	-0.22	0.09	0.74	2.77	2.52	2.12
Stelvio hot	0.03	0.29	1.48	1.90	1.26	1.38

Table 6-10: Integrated model error

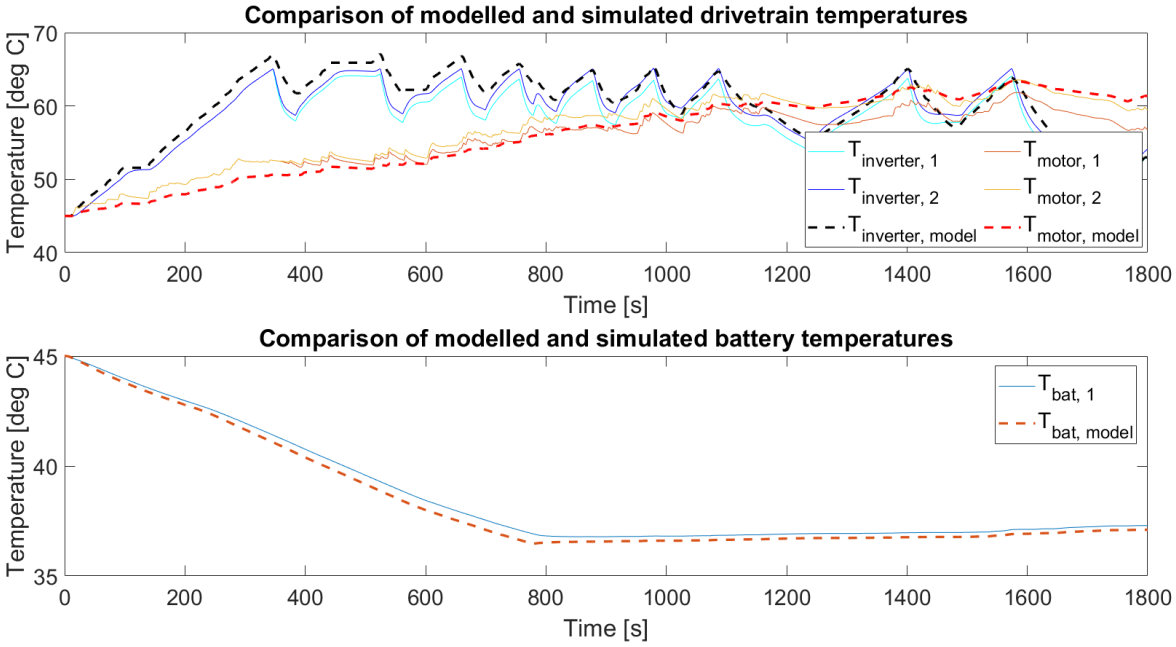


Figure 6-10: Component temperatures compared at WLTP - hot weather use case

6-5 Vehicle dynamics

Applying the vehicle model, the model disturbances, motor rpm, torque setpoints and battery current can be estimated from the vehicle speed profile and an elevation profile. Known vehicle parameters are used and no optimization is done to fit the fit parameters. For simplicity, a time-invariant model is assumed. The resulting model accuracy of the battery current and torque estimation is shown in Table 6-11.

	$\mu_{\epsilon,I_{bat}}[A]$	$\sigma_{\epsilon,I_{bat}}[A]$	$\mu_{\epsilon,\tau}[\tau]$	$\sigma_{\epsilon,\tau}[\tau]$
WLTP	-0.9	3.2	-2.0	7.1
Artemis	-1.8	8.8	-2.8	10.8
Stelvio	-0.3	5.6	-2.4	12.4

Table 6-11: Propulsion load estimation error

6-5-1 Modelling the inaccuracy of forecast information

To test the robustness of the Model Predictive control (MPC) strategy to inaccurate forecast information or imperfectness of driving behaviour, noise can be added to the velocity profile. To create realistic noise on the velocity profile, the following is assumed:

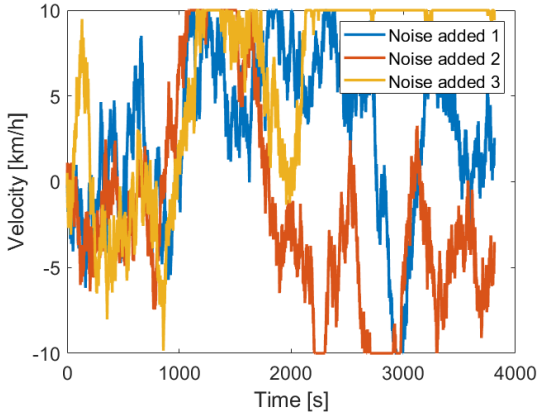
1. The noise level is correlated such that the resulting velocity profile is feasible. For example, if the error in prediction at timestep A is 10 km/h, the error in prediction at timestep A+1 should be close to 10 km/h.

2. The maximum noise level is limited to +10/-10 km/h.

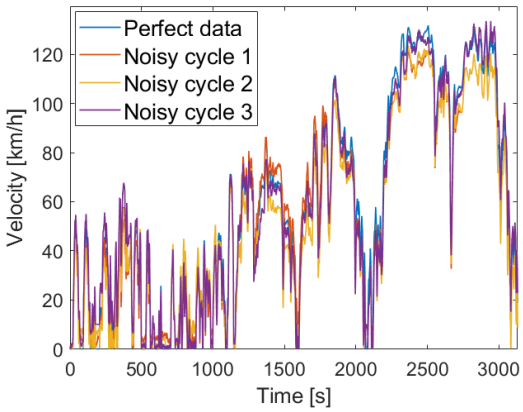
Based on the assumption, a cumulative sum of random noise is used, where the maximum level is limited to 10 km/h. The steps in creating the noise profile are shown in below:

1. create a vector with length equal to the full drive cycle, multiplied by the velocity data sampling rate f_s and assigned random values in the range [-0.25, 0.25].
2. Apply a cumulative sum over the vector elements, such the resulting noise level is correlated, allowing a maximum noise level of -10/+10 km/h.
3. Add the accumulated noise to the perfect velocity profile to obtain the noisy velocity profile.

The principle is visualized in Figure 6-11, where three different random noise signals are created and added to the Artemis drivecycle.



(a) Three resulting noise signals after a cumulative sum is applied over random noise



(b) Comparison of the true velocity cycle and the three resulting noisy velocity profiles

Figure 6-11: Noise is added to the velocity profile

Chapter 7

Results

The following chapter elaborates on the results of the tests done to validate the Model Predictive control (MPC) strategy. First, the control strategy with initial controller parameters is compared with the benchmark controller. Then, the effect of changing the ageing weight is discussed, as well as different options for the integrator, optimizer and prediction horizon. Finally, the effect of removing forecast information by freezing the predicted disturbances over the prediction horizon and adding noise to the forecast information is evaluated.

7-1 Validation of the MPC-based control strategy

As a baseline, the MPC-based control strategy with the settings shown in Table 7-1 is compared with the benchmark control strategy. The energy consumed in the Thermal Management System (TMS) throughout each of the use cases proposed in section 6-4 is discussed, as well as the relative battery ageing and the computation times required to find the optimal control action.

Sample time [s]	5
Prediction horizon [s]	300
Ageing weight $C_{\mathbf{ageing}}$	1
Integrator	RK4 fixed-step
Optimizer	IPOPT
Forecast source	Perfect without noise

Table 7-1: MPC initial settings

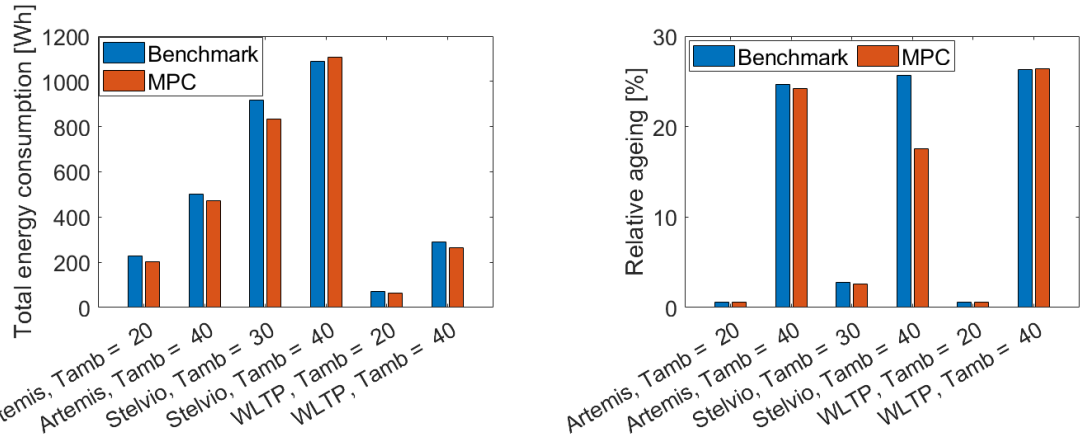
7-1-1 Energy consumption

In the base case, the MPC controller is able to save energy throughout each of the three drive cycles for moderate ambient conditions, between 20 and 30 °C. For hot ambient

conditions, between 30 and 40 °C, a slight increase in energy consumption can be observed for the Stelvio drive cycle. This is visualized in Figure 7-1a. The energy consumption difference is shown in Table 7-2. Assuming a total vehicle energy consumption of 96 wh/km, based on the Lightyear 0 WLTP range [70], the driving range is extended with almost one kilometer of driving for the moderate Stelvio use case, where the most energy is saved.

7-1-2 Ageing and power limits

Whilst being able to save energy, an equal or decreased relative ageing rate compared to the benchmark controller is achieved, as can be seen in Figure 7-1b. For the Stelvio drive cycle, a significant decrease in the relative ageing rate can be seen. The relative ageing difference is shown in Table 7-2.



(a) Energy consumption using the MPC control strategy and the benchmark strategy

(b) Relative ageing using the MPC control strategy and the benchmark strategy

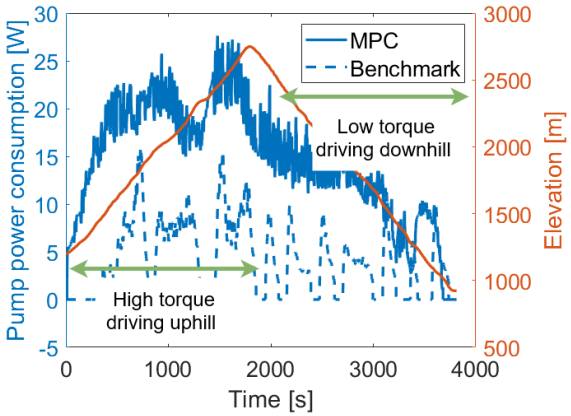
Figure 7-1: Performance metrics compared between MPC and benchmark strategy

Usecase	Energy spent [Wh]	Decrease in relative ageing [%]
Artemis, $T_{\text{amb}} = 20$	-26.1 (-11.4%)	-0.0 (-0.0%)
Artemis, $T_{\text{amb}} = 40$	-31.5 (-6.3%)	-0.5 (-2.0%)
Stelvio, $T_{\text{amb}} = 20$	-84.0 (-9.2%)	+0.2 (-7.1%)
Stelvio, $T_{\text{amb}} = 40$	+17.5 (+1.6%)	-8.0 (-31.4%)
WLTP, $T_{\text{amb}} = 20$	-5.4 (-7.7%)	-0.0 (-0.0%)
WLTP, $T_{\text{amb}} = 40$	-25.7 (-8.9%)	+0.1 (+0.4%)

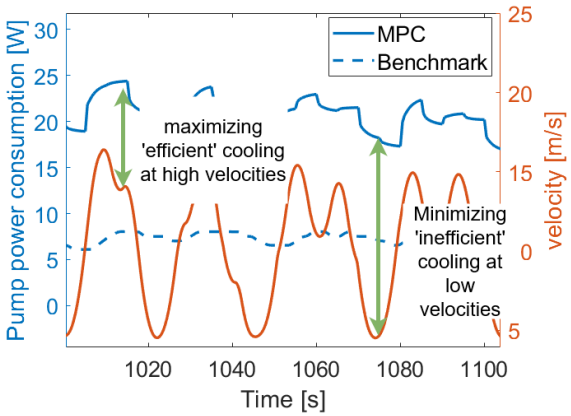
Table 7-2: Energy spend and relative ageing using the MPC-based strategy compared to using the benchmark strategy over different drive cycles and ambient conditions. The relative difference is shown in brackets.

7-1-3 Behavioral analysis

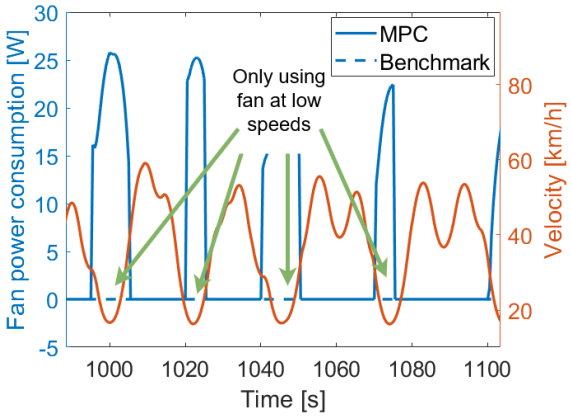
In this section, the relation between the MPC behavior and the losses, the velocity and the time-varying Coefficient Of Performance (COP), that are included in the prediction model,



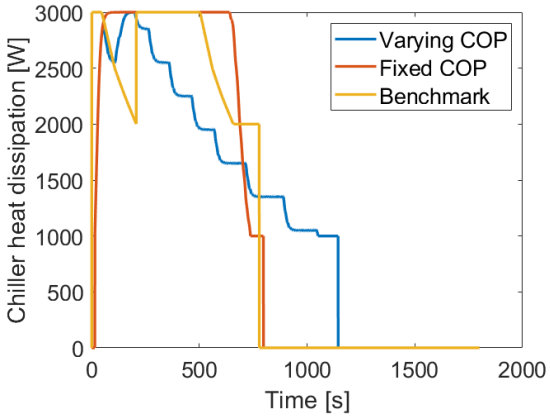
(a) Pump power consumption uphill and downhill for moderate Stelvio drive cycle



(b) Pump power consumption and velocity zoomed in for moderate Stelvio drive cycle



(c) Fan power consumption and velocity zoomed in for moderate Stelvio drive cycle



(d) Chiller heat dissipation for the hot WLTP drive cycle

Figure 7-2: The effect of several runtime parameters on controller behaviour

are analysed. Plots showing the control trajectories, component temperature trajectories and computation times for each of the six use cases, for the benchmark and MPC-based control strategy, are shared in Appendix B. This section shares the summarized results.

Component losses In the Stelvio drive cycle, there is a clear distinction between a high-torque section, when driving uphill and a low-torque section, when driving downhill. At high torque, the motor losses are higher and intuitively, more cooling is desired to minimize temperature-dependent losses. The MPC behaviour reflects this intuition, as can be seen in Figure 7-2a. Both in advance of reaching the peak, and the end of the drive cycle, pump power consumption is decreased to save energy. This is further discussed in section 7-4, where different prediction horizons are compared. The benchmark control is also plotted, showing a lower pump power consumption, only occurring when an inverter or motor reaches the respective temperature limit, as discussed in section 2-2, not taking into account temperature-dependent component efficiencies, such that the overall power consumption is higher.

Vehicle velocity Heat dissipation at the radiator is more effective at higher velocities. Therefore, if there are many fluctuations in velocity, intuitively, more cooling should be done at higher velocities and less cooling at lower velocities. This means a higher pump power consumption at higher velocities, to achieve a lower total average power consumption. This expected behaviour can be seen back in Figure 7-2b, where the pump power consumption is increased at higher velocities to cool most effectively compared to at lower velocities. Additionally, the fan is only turned on when the vehicle speed is too low and remains inactive at higher velocities, as can be seen in Figure 7-2c. In this case, the benchmark control strategy never turns on the fan.

Effect of the time-varying Coefficient Of Performance (COP) At hot ambient conditions, the chiller is used to cool down the battery until reaching a healthy temperature. Tests were done both applying a prediction model using a time-varying COP and a fixed COP that is independent of the cooling rate. When using a fixed COP, the optimal control is bang-bang control, as shown in Figure 7-2d. When using a time-varying COP that is higher at a higher chiller heat dissipation, a different control strategy is found, as also shown in Figure 7-2d. The battery is cooled down at a slower rate to save energy until reaching the same temperature of around $T_{\text{battery}} = 309$ K. This causes the ageing rate to be slightly higher, but the energy consumption to be significantly decreased. Only results for the hot WLTP and hot Artemis use case are compared as can be seen in Table 7-3. Only in these use cases, the chiller is used to cool down the battery from the initial hot temperature.

	Fixed COP		Time-varying COP	
Usecase	Relative ageing [%]	Energy consumption [Wh]	Relative ageing [%]	Energy consumption [Wh]
WLTP, 40 deg	24.6	295.1	25.8	263.3
Artemis, 40 deg	22.4	509.8	23.7	470.8

Table 7-3: Results using a fixed and a time-varying COP

7-2 Ageing weight

The ageing weight C_{ageing} can be set to achieve the desired trade-off between minimizing the ageing rate and power consumption, two conflicting objectives. For the control strategy validation, initially, the ageing weight $C_{\text{ageing}} = 1$ was set, as described in section 7-1. As can be seen in Table 7-4, when the ageing weight is increased, the relative ageing decreases at the cost of a higher energy consumption. Figure 7-3 shows the battery temperature profile for the WLTP drivecycle at hot ambient conditions for the MPC-based control strategy for different ageing weights.

The strategy should trade-off minimizing the energy consumption and ageing rate as desired. The battery temperature should be kept below 45 °C such that discharging is not limited, as was visualized in Table 2-1 and preferably closer to 35 °C, such that also charging is least limited. Secondly, a decrease in energy consumption compared to the benchmark strategy is desired whilst maintaining an equal ageing rate. The results obtained

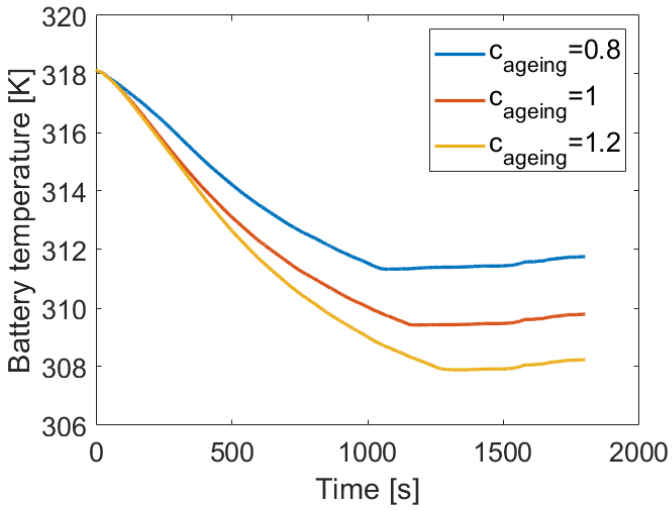


Figure 7-3: Battery temperature profiles compared for the MPC strategy using different ageing weights compared and the benchmark control strategy

using an ageing weight of $C_{\text{ageing}} = 1$ trade-off ageing and energy consumption as desired. Therefore the ageing weight of $C_{\text{ageing}} = 1$ is chosen and used in the remainder of this thesis.

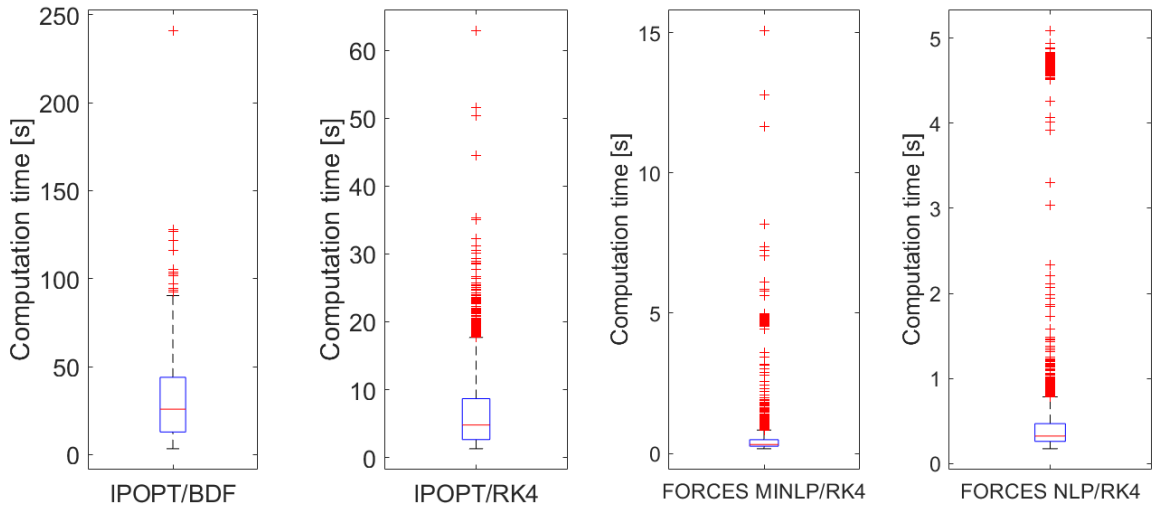
WLTP	Benchmark	C_{ageing}=0.8	C_{ageing}=1.0	C_{ageing}=1.2
Energy consumption [Wh]	288.9	219.9 (-23.8%)	263.35 (-8.8%)	300.6 (+4.0%)
Relative ageing rate [%]	26.3	31.2 (+18.6%)	26.4 (+0.4%)	23.7 (-9.9%)
Artemis				
Energy consumption [Wh]	502.3	432.1 (-14.0%)	470.8 (-6.3%)	505.6 (+0.7%)
Relative ageing rate [%]	24.7	30.0 (+21.5%)	24.2 (-2.0%)	20.5 (-17.0%)
Stelvio				
Energy consumption [Wh]	1089.3	1087.2 (-0.2%)	1106.8 (+1.6%)	1141.9 (+4.8%)
Relative ageing rate [%]	25.7	19.8 (-23.0%)	17.5 (-31.9%)	12.5 (-51.4%)

Table 7-4: Results on energy consumption for different ageing weights in the different drive cycles at hot ambient conditions. The relative difference is shown in brackets.

7-3 Comparison of integrators and optimizers

The results for using the fixed-step RK4 integrator are compared to the results applying the BDF integrator. Figure 7-4a and Figure 7-4b show the computation times for the implicit adaptive-stepsize BDF integrator and the explicit fixed-stepsize RK4 integrator, in

combination with the IPOPT optimizer. The energy consumption within the TMS and ageing rate are compared in Table 7-5 and Table 7-6 respectively for the different integrators. As can be seen, when using the same IPOPT optimizer, the RK4 integrator is computationally more efficient whilst achieving no performance loss compared to the MPC control strategy using the BDF integrator. By looking at the number of datapoints outside of the boxplot whisker, it can be seen that the spread in computation times is larger. Due to the lower computation times, for the TMS, the RK4 integrator is the preferred integrator due to the increased computational efficiency.
Applying the RK4 integration method, The IPOPT optimizer is compared to the Forces interior-point optimizer and the Forces mixed-integer method that uses the same interior-point optimizer. Additionally, tests were done using both the Casadi and Forces Sequential Quadratic Programming (SQP) optimizers, but these were not able to converge towards optimal solutions. Therefore further analysis is only done based on the results obtained using the interior point methods. The computation times are shown in Figure B-25 for each of the tested optimizers. The results on the energy consumption and ageing rate are shown in Table 7-5 and Table 7-6 respectively. The computation times using the Forces mixed-integer method have the largest difference between the box size and the maximum computation times. This can be caused due to the fact that the computation time can vary significantly depending on how many options for the integer valve position are evaluated throughout the branch bound process. The Forces interior-point optimizer is computationally most efficient whilst achieving minimal losses in performance compared to the mixed-integer solver and no loss in performance compared to the IPOPT optimizer. Due to the low computation times, the Forces NLP solver is the preferred optimizer in combination with the RK4 integrator.



(a) IPOPT optimizer + BDF integrator

(b) IPOPT optimizer + RK4 integrator

(c) Forces MINLP optimizer + RK4

(d) Forces NLP optimizer + RK4 integrator

Figure 7-4: Computation times for different combinations of integrators and optimizers

	Absolute energy consumption [Wh]				
Usecase	Benchmark	BDF IPOPT	RK4 IPOPT	Forces NLP	Forces MINLP
WLTP, 20 deg	69.8	64.4 (-7.7%)	64.4 (-7.7%)	64.7 (-7.3%)	64.7 (-7.3%)
Artemis, 20 deg	229.3	X	203.2 (-11.4%)	203.2 (-11.4%)	203.2 (-11.4%)
Stelvio, 20 deg	917.4	X	833.3 (-9.2%)	833.3 (-9.2%)	833.6 (-9.1%)
WLTP, 40 deg	288.9	263.3 (-8.9%)	263.3 (-8.9%)	260.6 (-9.8%)	259.8 (-10.1%)
Artemis, 40 deg	502.3	X	470.8 (-6.3%)	467.8 (-6.9%)	470.4 (-6.4%)
Stelvio, 40 deg	1089.3	X	1106.9 (+1.6%)	1116.8 (+2.5%)	1117.7 (+2.6%)

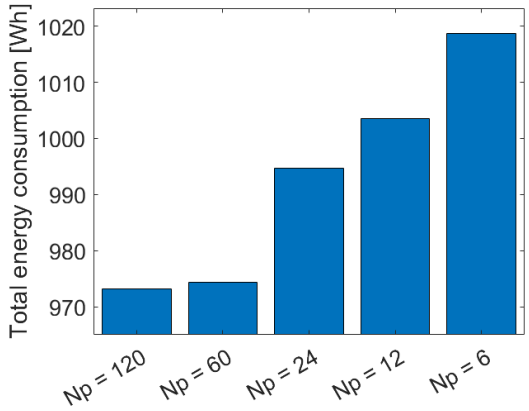
Table 7-5: Results on energy consumption for each of the different drive cycles and control configurations. The relative difference is shown in brackets.

	Relative ageing [%]				
Usecase	Benchmark	BDF IPOPT	RK4 IPOPT	Forces NLP	Forces MINLP
WLTP, 20 deg	0.6	0.6 (+0.0%)	0.6 (+0.0%)	0.6 (+0.0%)	0.58 (+0.0%)
Artemis, 20 deg	0.6	X	0.6 (+0.0%)	0.6 (+0.0%)	0.6 (+0.0%)
Stelvio, 20 deg	2.8	X	2.6 (-7.1%)	2.6 (-7.1%)	2.6 (-7.1%)
WLTP, 40 deg	26.3	X	26.4 (+0.4%)	26.6 (+1.1%)	28.6 (+8.7%)
Artemis, 40 deg	24.7	X	24.2 (-2.0%)	24.4 (-1.2%)	25.3 (+2.4%)
Stelvio, 40 deg	25.7	X	17.5 (-31.4%)	14.2 (-44.3%)	14.1 (-44.7%)

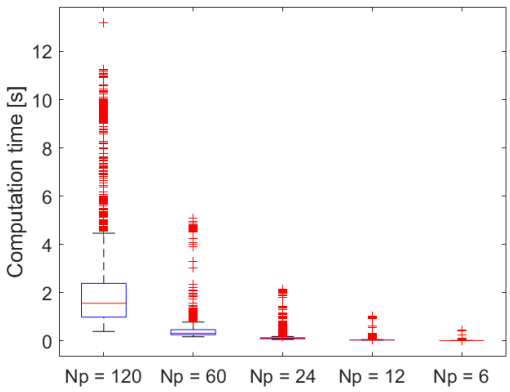
Table 7-6: Results on ageing rate for each of the different drivecycles and control configurations. The relative difference is shown in brackets.

7-4 Influence of prediction horizon

The computation times for different prediction horizons between $N_p = 6$ and $N_p = 120$, between 30 and 600 seconds, are evaluated and can be seen in Figure 7-5b. Here the ageing weight is set to $C_{\text{ageing}} = 0$, to make quantitative comparison possible, whilst not taking into account the battery ageing. If a nonzero ageing weight would be chosen, a different ageing weight is necessary for each of the different prediction horizons to get comparable results in terms of battery ageing. The energy consumption of the TMS for different prediction horizons for the Stelvio drive cycle is shown in Figure 7-5a. The results for the other drive cycles are shown in subsection B-2-4.



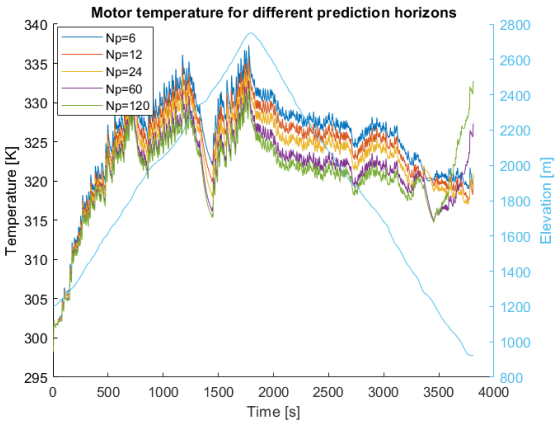
(a) Energy consumption for different prediction horizons



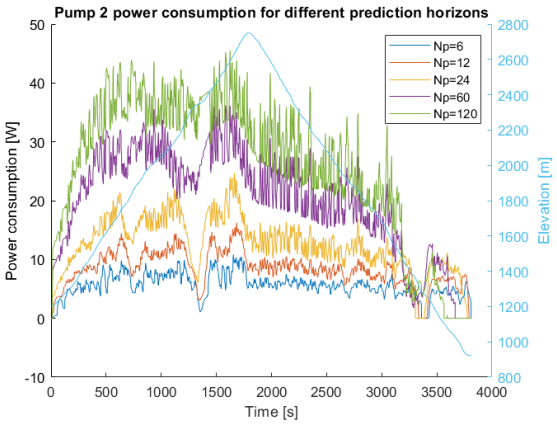
(b) Computation time for different prediction horizons

Figure 7-5: Different prediction horizons and their effect on TMS energy consumption and computation times for the moderate Stelvio use case and $C_{\text{ageing}} = 0$

The effect of increasing the prediction horizon is different for each drive cycle. For the Stelvio drive cycle, there are high losses and there is a significant change in torque after reaching the peak of the mountain. Here, there is an advantage of using a longer prediction horizon for the control strategy. For the motor temperature, the effect of increasing the prediction horizon in the Stelvio drive cycle result in a lower average motor temperature, as visualized in Figure 7-6a. Also, it can be seen that the motor temperature changes more in advance of future disturbances. For example. at the end of the drive cycle no more losses are predicted. The active cooling is decreased as there is no gain in cooling the motors. Similar behaviour can be seen in advance of reaching the peak. A prediction horizon of N_p=60 is chosen, as it is the largest prediction horizon for which the maximum computation time is still within the sample time that is set to T_s =5s, as described in section 5-3.



(a) Motor temperature for different prediction horizons



(b) Pump 2 behavior for different prediction horizons

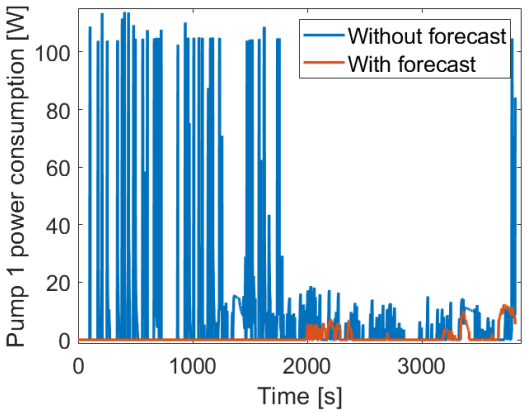
Figure 7-6: Motor related behavior of controller for different prediction horizons in the Stelvio drivecycle

7-5 Influence of removing forecast information

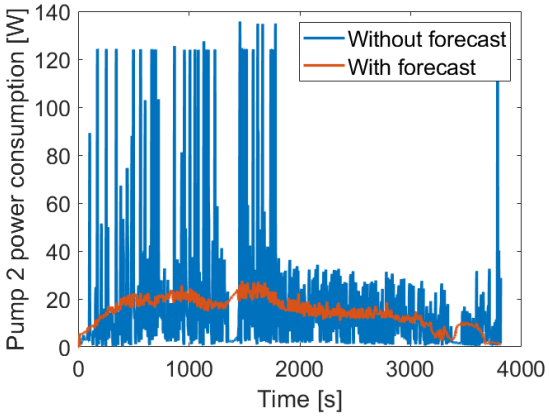
The value of using forecast information has not yet been quantified. In real-life scenarios, forecast information may not always be useful or it may not even be available. To evaluate the controller performance when no forecast information is used, in this test, the forecast parameters are frozen over the entire prediction horizon. This means that the prediction is based only on the current disturbances and not on external forecasts. The results for the moderate use case are shown in Table 7-7. When the forecast information is removed, the general trend is that still a relative improvement compared to the benchmark is observed, but also a significant decrease in performance compared to when forecast information is used. Typical behavior is shown in Figure 7-7, showing that when the current disturbances are used over the entire prediction horizon, the control strategy overreacts to the current disturbances. This results in a larger increase in energy consumption for the Stelvio drive cycle compared to the other drive cycles, as the forecasted parameters vary most throughout the drive cycle. Sudden peaks in power consumption can be observed for both pumps and the fan in Figure 7-7a, Figure 7-7b and Figure 7-7c, respectively. For the chiller heat dissipation, used in the hot use cases and shown in Figure 7-7d, the cooling trajectory fluctuates more causing an increased compressor power consumption.

Usecase	Ageing			Energy consumption		
	Bench-mark	Forecast	No forecast	Bench-mark	Forecast	No forecast
WLTP 20	0.6	0.6 (+0.0%)	0.6 (+0.0%)	69.8	64.7 (-7.3%)	65.5 (-6.2%)
Artemis 20	0.6	0.6 (+0.0%)	0.6 (+0.0%)	229.3	203.2 (-11.4%)	210.7 (-8.1%)
Stelvio 20	2.8	2.6 (-7.1%)	2.8 (+0.0%)	917.4	833.4 (-9.2%)	871.5 (-5.0%)
WLTP 40	26.3	26.6 (+1.1%)	26.2 (-0.4%)	288.9	260.6 (-9.8%)	265.3 (-8.0%)
Artemis 40	24.7	24.4 (-1.2%)	23.7 (-4.0%)	502.3	467.8 (-6.9%)	481.2 (-4.2%)
Stelvio 40	25.7	14.2 (-44.3%)	11.5 (-55.3%)	1089.3	1116.6 (+2.5%)	1153.3 (+5.9%)

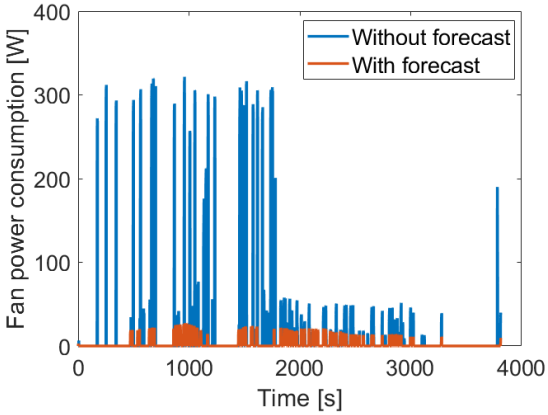
Table 7-7: Energy saved and decrease in relative ageing using MPC without forecast information compared to the benchmark and MPC with forecast information. The relative difference is shown in brackets.



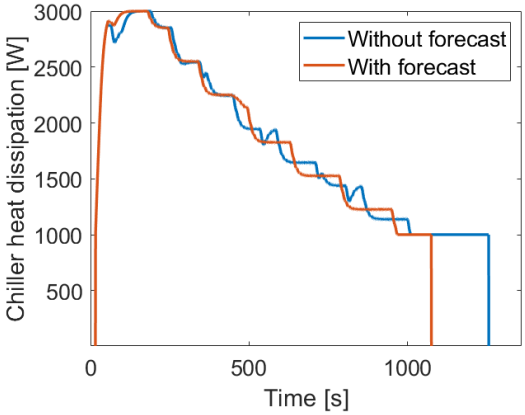
(a) Power consumption for the pump in the battery cooling circuit for the moderate Stelvio use case



(b) Power consumption for the pump in the drivetrain cooling circuit for the moderate Stelvio use case



(c) Fan power consumption for the moderate Stelvio use case



(d) Chiller heat dissipation for the hot Artemis use case

Figure 7-7: Controller behavior for the moderate Stelvio use case using time-varying forecast information in the prediction horizon compared to using the current disturbances throughout the prediction horizon

7-6 Influence of imperfect forecast information

Instead of perfect forecast information, noisy imperfect forecast information is used in this test. Only tests are done for the moderate ambient use cases, as the effect of adding noise to the velocity profile is evaluated, as described in subsection 6-5-1, and not to the temperature profile. Noise is added to the true velocity profile and, using both the velocity and elevation profiles, the forecast parameters, the torque and battery current, are estimated using a basic vehicle model. A minimal loss in performance can be recognized for the WLTP and Artemis drive cycles, as shown in Table 7-8. For the Stelvio drive cycle, a more significant difference can be seen. This can be explained by the fact that a small difference in future velocity prediction has a large impact on the losses in the Stelvio drive cycle, as the car is driving in mountainous terrain and is often accelerating and decelerating.

WLTP	Perfect forecast	Noisy cycle 1	Noisy cycle 2	Noisy cycle 3
Energy consumption [Wh]	64.4	64.7 (+0.4%)	64.7 (+0.5%)	64.7 (+0.4%)
Relative ageing rate [%]	0.58	0.58 (+0.0%)	0.58 (+0.0%)	0.58 (+0.2%)
Artemis				
Energy consumption [Wh]	203.1	203.2 (+0.1%)	205.58 (+1.2%)	203.3 (+0.1%)
Relative ageing rate [%]	0.61	0.61 (-0.2%)	0.61 (-0.2%)	0.61 (-0.1%)
Stelvio				
Energy consumption [Wh]	833.4	843.7 (+1.2%)	834.0 (+0.1%)	844.9 (+1.4%)
Relative ageing rate [%]	2.6	3.9 (+49.9%)	2.9 (+9.7%)	4.10 (+57.2%)

Table 7-8: Energy saved and decrease in relative ageing using MPC with forecast parameters obtained using the vehicle model and three noisy velocity profiles compared to MPC using ideal forecast without noise. The relative difference is shown in brackets.

Chapter 8

Conclusion & Discussion

A nonlinear Model Predictive control (MPC)-based control strategy for the Thermal Management System (TMS) of Electric Vehicle (EV)'s was presented and compared to a benchmark control strategy. The advantages, reducing High Voltage Battery (HVB) ageing and overall vehicle energy consumed, were quantified. Depending on the drive cycle, up to 84 Wh can be saved, while maintaining an equal ageing rate. A decrease in power consumption between 7% and 11% is achieved for 5 out of 6 use cases, while additionally decreasing the ageing rate by 0% to 7%. For one use case, an increase in energy consumption is achieved by 2.5%, but the relative ageing rate is decreased by 44%.

The sensitivities with respect to predictable disturbances were analyzed. The energy is saved partly by taking into account temperature-dependent component efficiencies. At long drive cycles and drive cycles with high average torque, the most energy can be saved. Secondly, by modelling the chiller Coefficient Of Performance (COP), an energy-efficient battery cooling trajectory can be achieved at hot ambient conditions. The behavior of the control strategy is analyzed when no forecast information is used, but the forecasted parameters are frozen throughout the prediction horizon, based on the current disturbances. Highly fluctuating behavior is observed, overreacting to the current disturbances, which results in inferior performance compared to when forecast information is used, but an improved performance compared to the benchmark. The loss in performance when using imperfect, noisy, forecast information instead of perfect forecast information is quantified. Using noisy forecast information, the energy consumption and ageing rate increase by between 0.0% to 1.2%, depending on the drive cycle. The losses are highest for the Stelvio drive cycle, where a small change in forecasted velocity has a relatively large effect on the predicted losses.

The optimal control strategy developed in this thesis is also fast enough for real-time use, as the maximum computation time is less than the sample time $T_s = 5$s. By using a fixed-stepsize integrator, a prediction horizon of 300 seconds, and the Forces NLP optimizer, both energy-efficient control and fast enough computation are achieved.

The methodology using white box models and simple tuning of the weights of ageing and energy consumption costs allow for scalability and reusability, which are important design

requirements in the automotive context. It has to be noted that actual ageing tests are required for vehicle deployment, in order to say anything about the true vehicle performance.

In this thesis, the effect of adding noise to the velocity profile, used to estimate the vehicle torque and battery current, was quantified. However, noise in real life may have a different distribution, which in future research has to be validated by using actual driver data. Also, several assumptions were used in this thesis. In real life, the navigation system may not always be turned on, in which case the taken route is not known in advance. Next to that, even if the route is known, the driver might change destination which can have an impact on the optimal temperature trajectories.

The current methodology is fast enough for online control in the simulation platform. Following the controller design methodology shown in Figure 8-1, there are steps to take until the method would be suitable for vehicle deployment. System identification tests need to be done using the real vehicle, instead of the simulation platform. The tests used in this thesis are suitable for real-vehicle experiments as well. Then, controller validation should be done on a prototype vehicle. If the control strategies fulfil the functional requirements, the controller should be prepared for embedded implementation, and integrated into the vehicle compute platform.

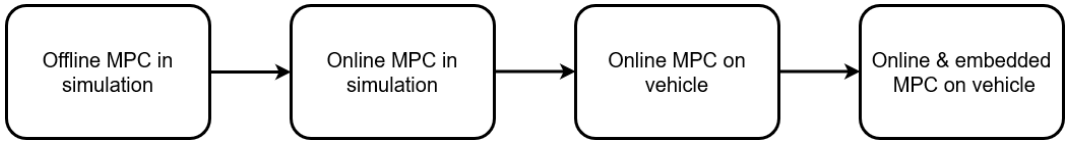


Figure 8-1: Process from the development onto real-vehicle deployment

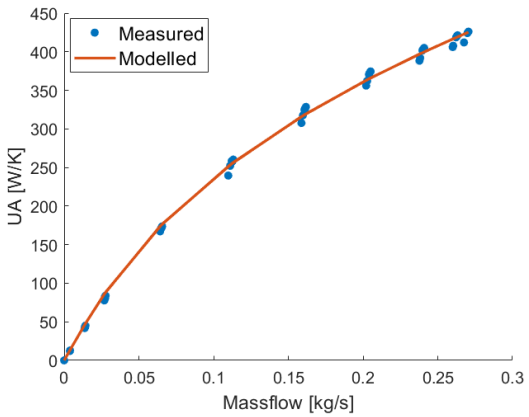
Appendix A

Integrated Model Validaton

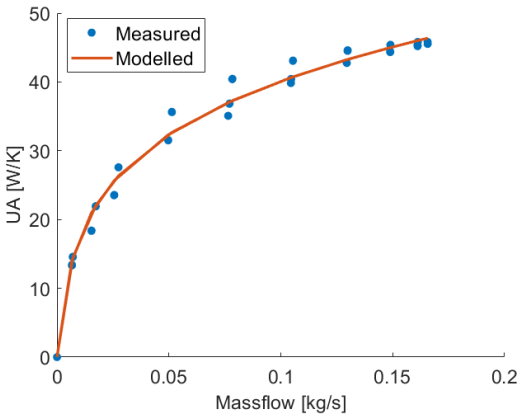
In this appendix, complete model validation results are shown for the thermal model, hydraulic model and electrical model parameters. Finally the complete results for the integrated model validation are shown.

A-1 Thermal model validation

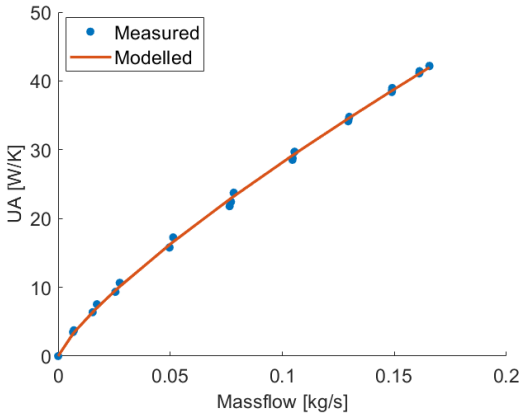
The thermal model validation results for the battery, motor, inverter and radiator are shown in Figure A-1.



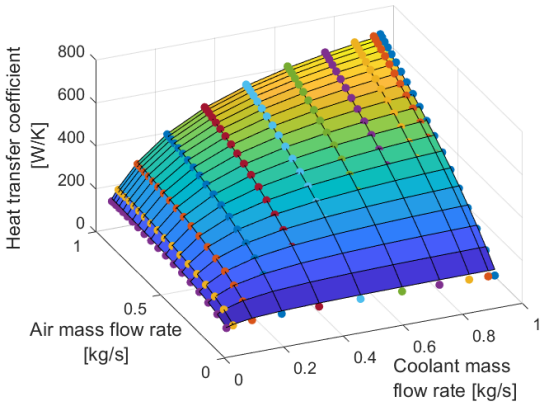
(a) Heat transfer coefficient from battery to coolant



(b) Heat transfer coefficient from motor to coolant



(c) Heat transfer coefficient from inverter to coolant

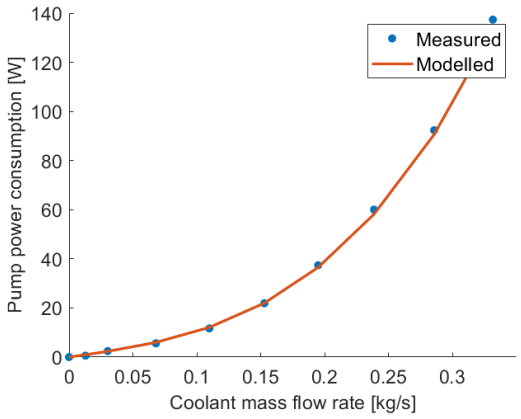


(d) Heat transfer coefficient from coolant to air

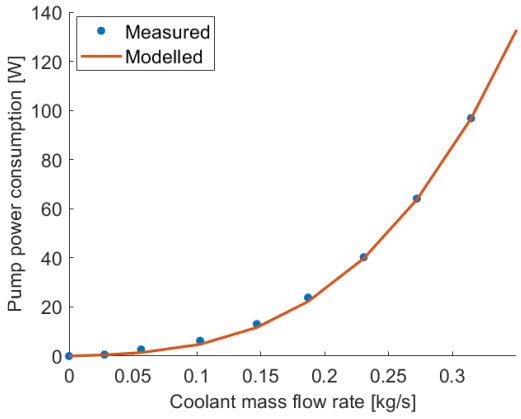
Figure A-1: Heat transfer coefficient for each of the different components

A-2 Hydraulic model validation

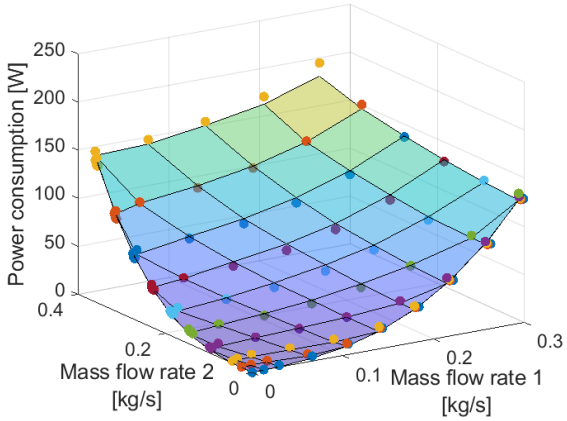
The hydraulic model parameters for the case when the battery is cooled through the chiller, when w_1 =1 and when the battery is cooled through the radiator, when w_1=0, are shown in Figure A-2.



(a) Pump 1 power consumption when $w_1 = 1$



(b) Pump 2 power consumption when $w_1 = 1$



(c) Combined power consumption of the two pumps when $w_1 = 0$

Figure A-2: Hydraulic model validation results

A-3 Electrical model validation

The inverter losses, depending on function are plotted in Figure A-3, whereas the different types of motor losses are shown in Figure A-4

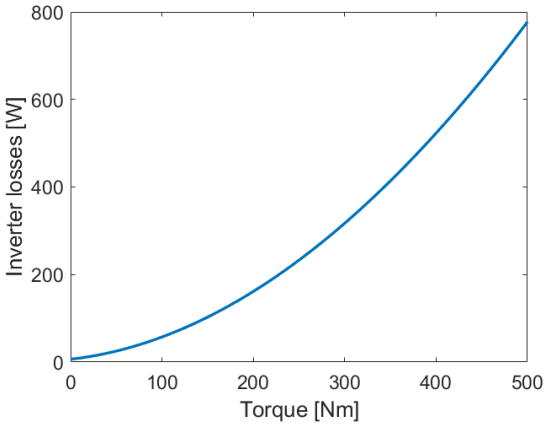
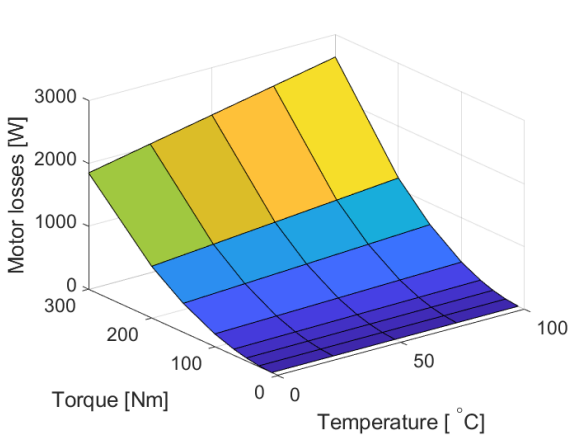
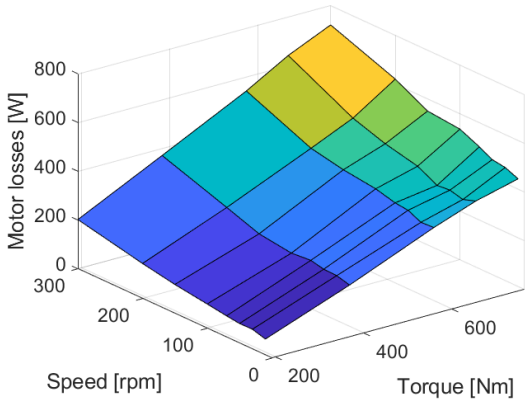


Figure A-3: Inverter losses are described as a function of torque



(a) Temperature dependent winding losses



(b) Temperature-independent motor losses are described as a function of torque and speed

Figure A-4: Temperature-dependent and independent motor losses

A-4 Integrated model validation

The integrated model validation results for the WLPT moderate and hot use case are shown in Figure A-5 and Figure A-6 respectively, for the Artemis moderate and hot use case in Figure A-7 and Figure A-6 respectively and finally for the Stelvio moderate and hot use case in Figure A-9 and Figure A-10 respectively.

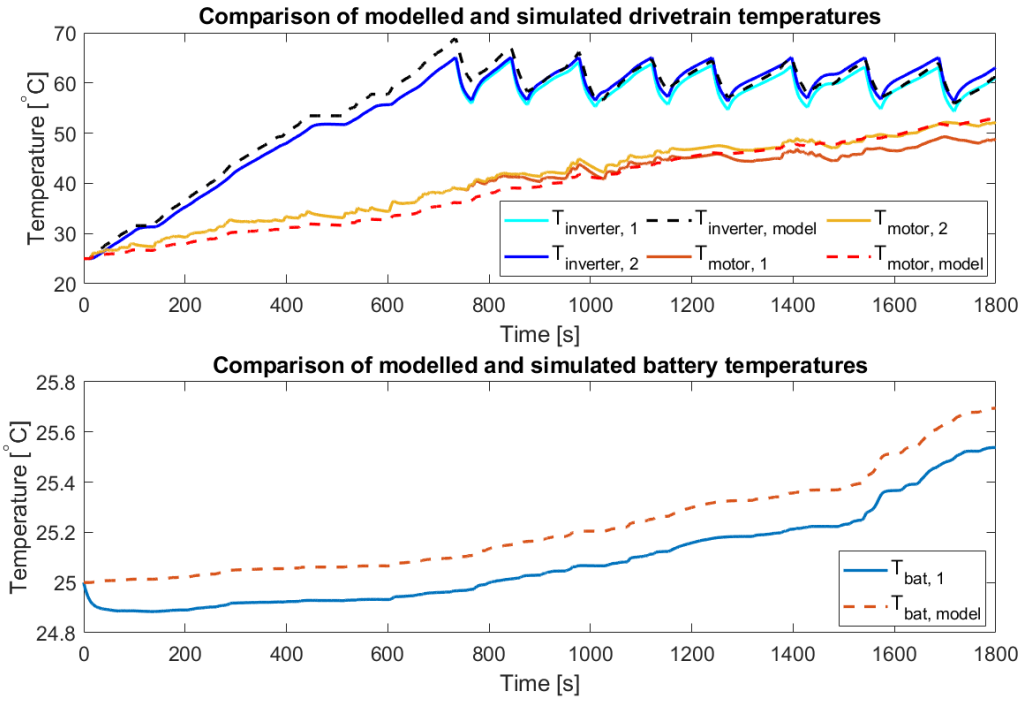


Figure A-5: Component temperatures compared at WLTP - moderate weather use case

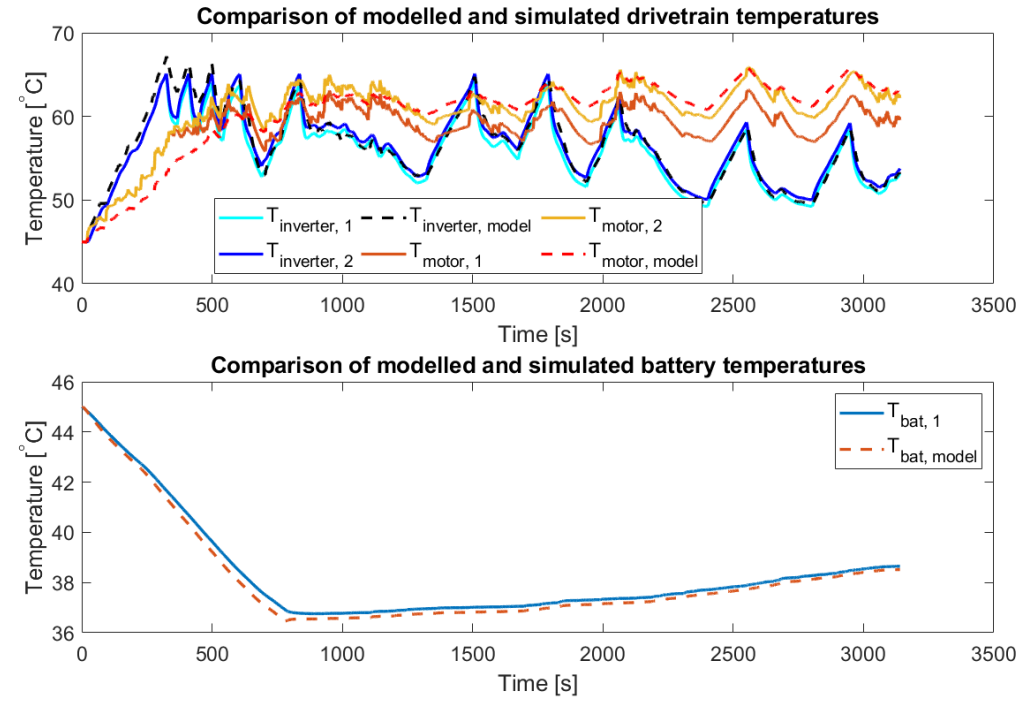


Figure A-6: Component temperatures compared at WLTP - hot weather use case

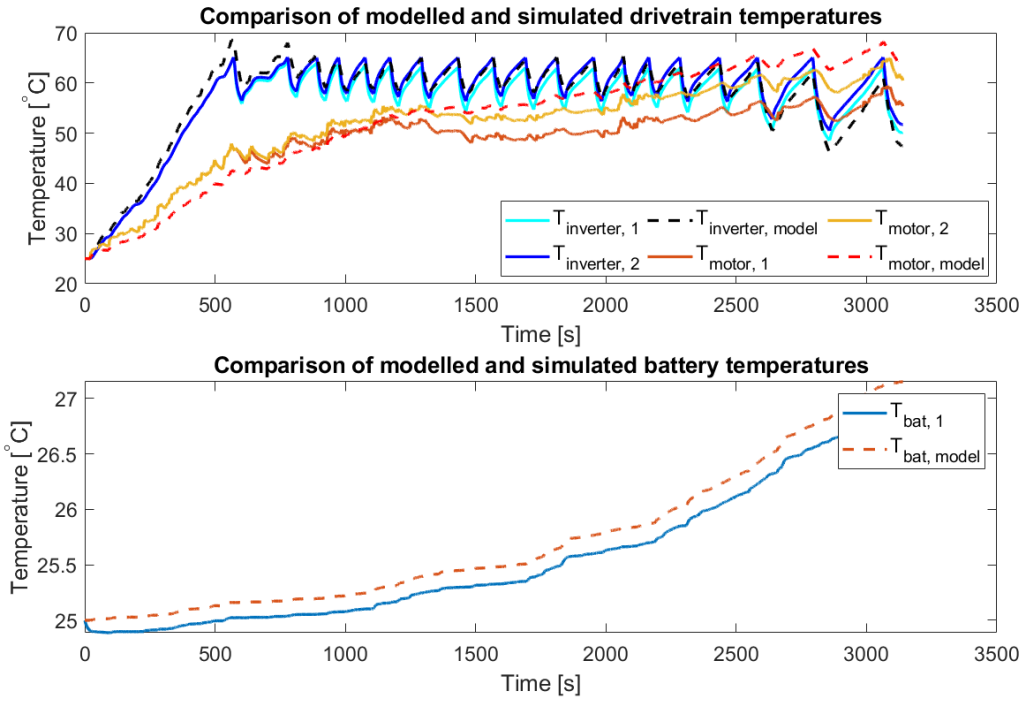


Figure A-7: Component temperatures compared at Artemis - moderate weather use case

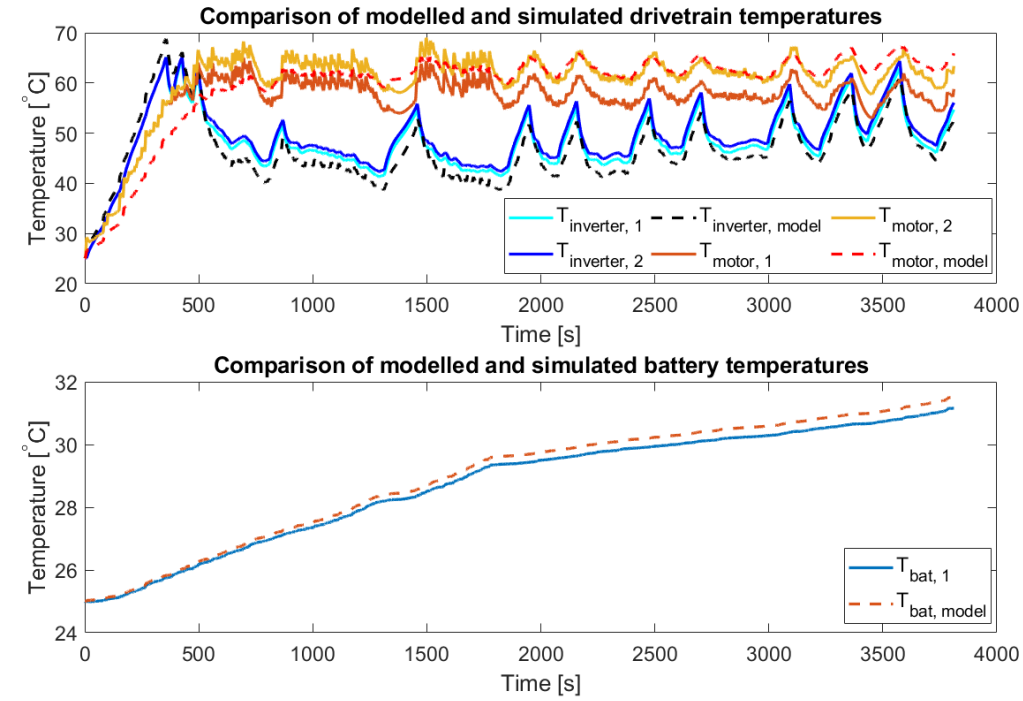


Figure A-8: Component temperatures compared at Moderate - hot weather use case

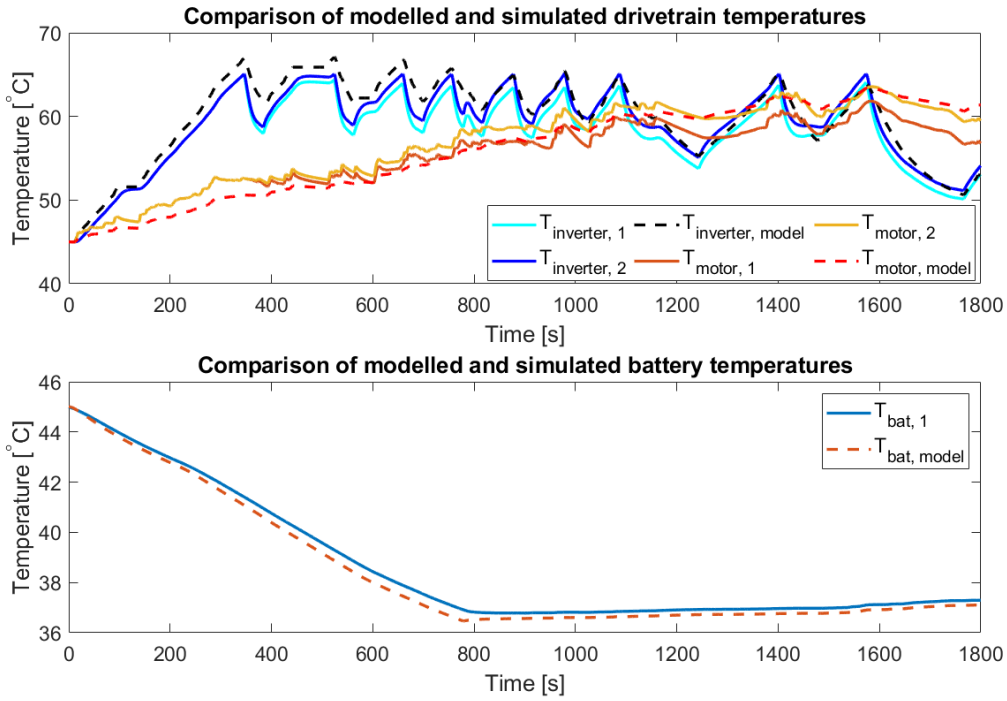


Figure A-9: Component temperatures compared at Stelvio - moderate weather use case

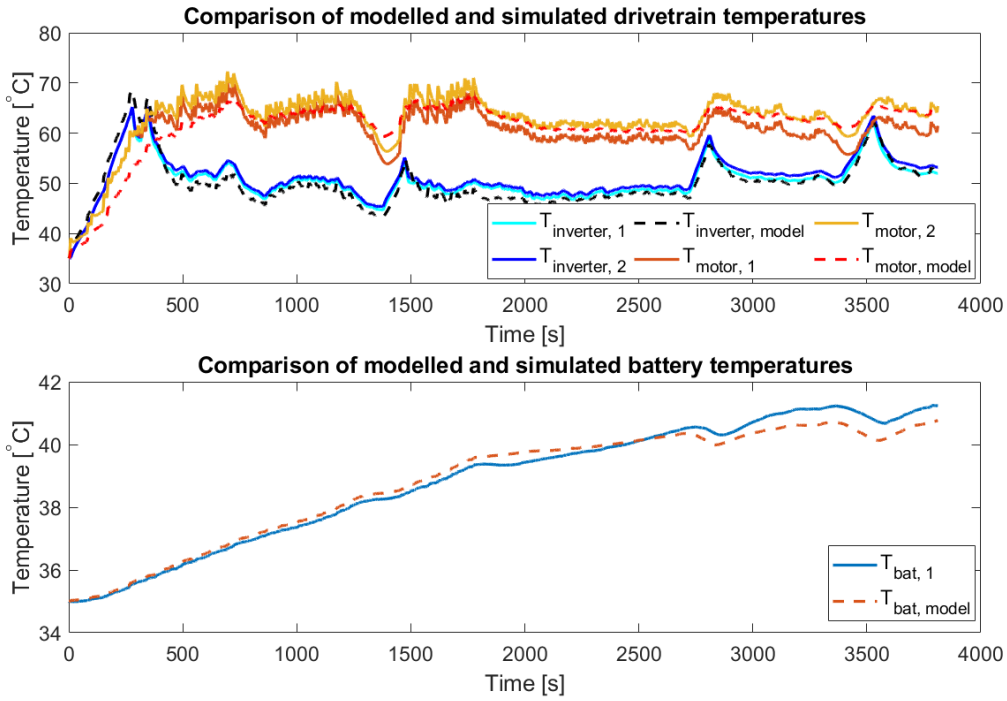


Figure A-10: Component temperatures compared at Stelvio - hot weather use case

Appendix B

Control Strategy Validation

In this appendix, the complete results are given for the benchmark control strategy and the MPC-based control strategy. Detailed results for each use case are shown, including component temperature trajectories, all control trajectories and computation times.

B-1 Benchmark controller

For the benchmark controller, first, the control trajectories are shown for each use case. Then the component temperature trajectories are shown.

B-1-1 Control behavior

The control trajectories for the WLPT moderate and hot use case are shown in Figure B-1 and Figure B-4 respectively, for the Artemis moderate and hot use case in Figure B-2 and Figure B-5 respectively and finally for the Stelvio moderate and hot use case in Figure B-3 and Figure B-6 respectively.

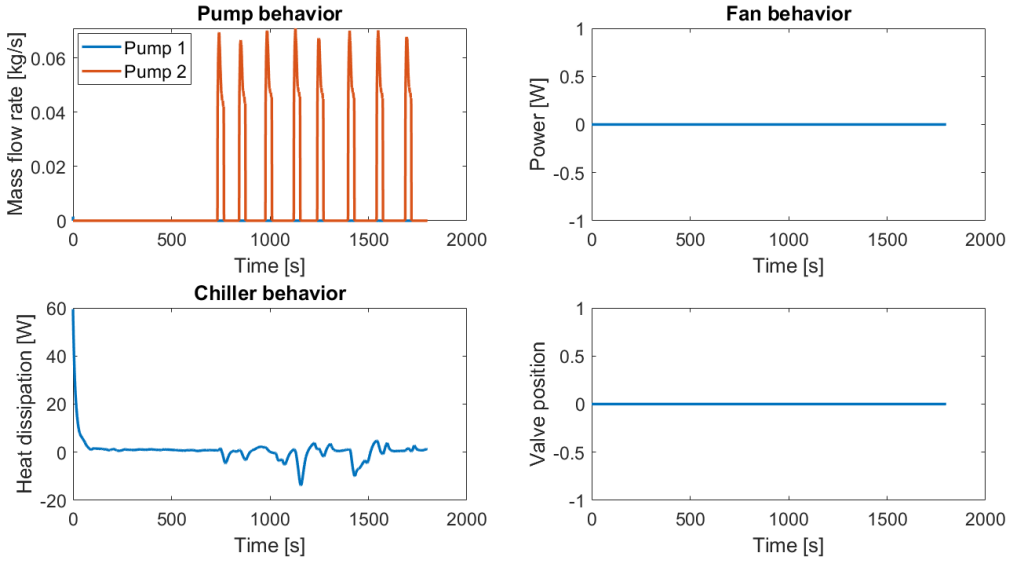


Figure B-1: Resulting control actions using the benchmark control strategy for the moderate WLTP use case

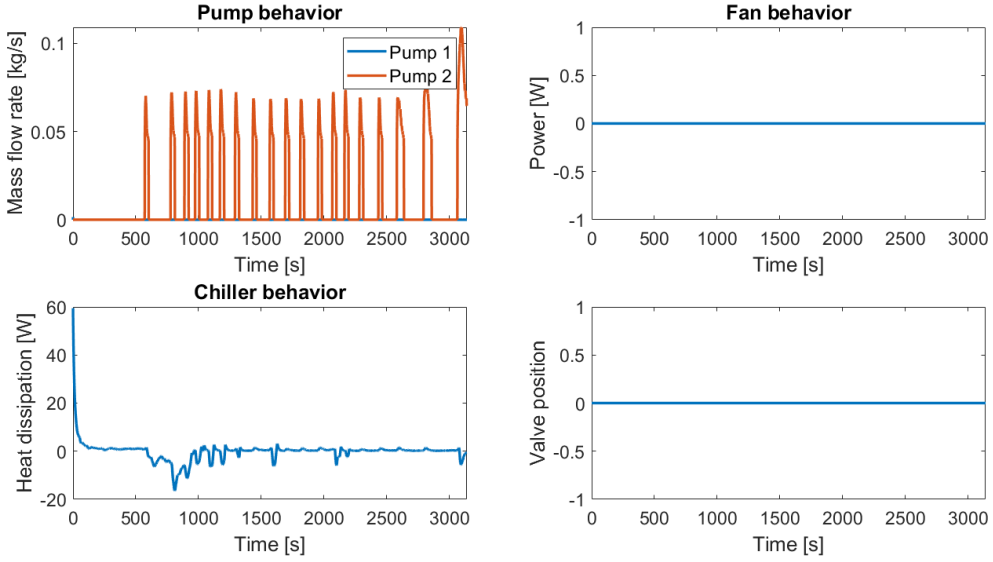


Figure B-2: Resulting control actions using the benchmark control strategy for the moderate Artemis use case

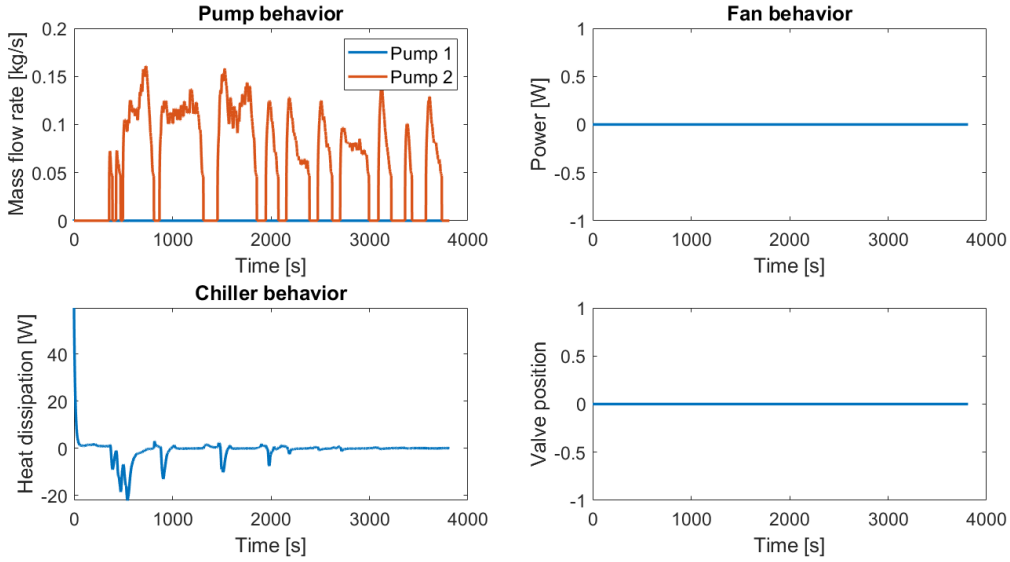


Figure B-3: Resulting control actions using the benchmark control strategy for the moderate Stelvio use case

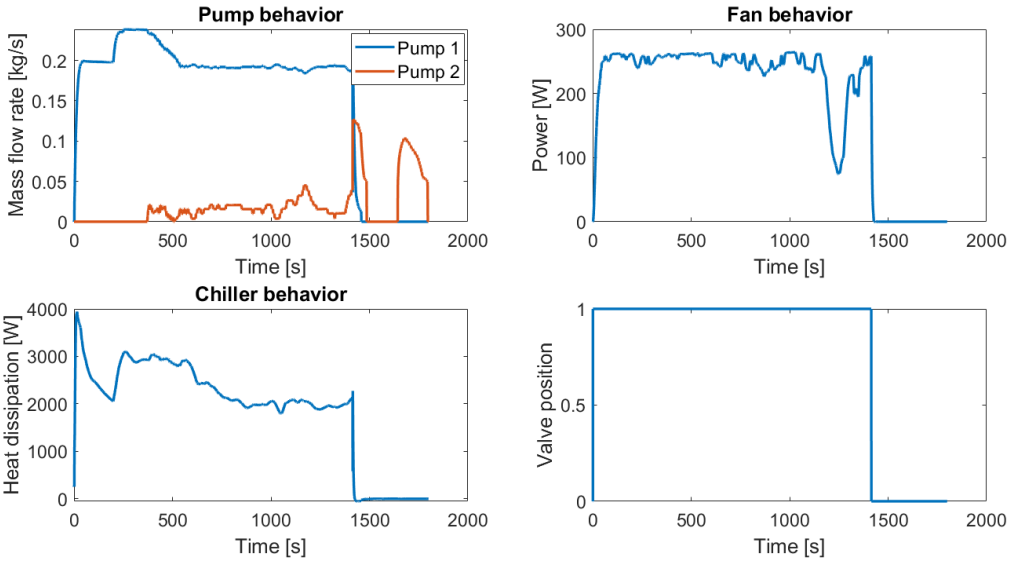


Figure B-4: Resulting control actions using the benchmark control strategy for the hot WLTP use case

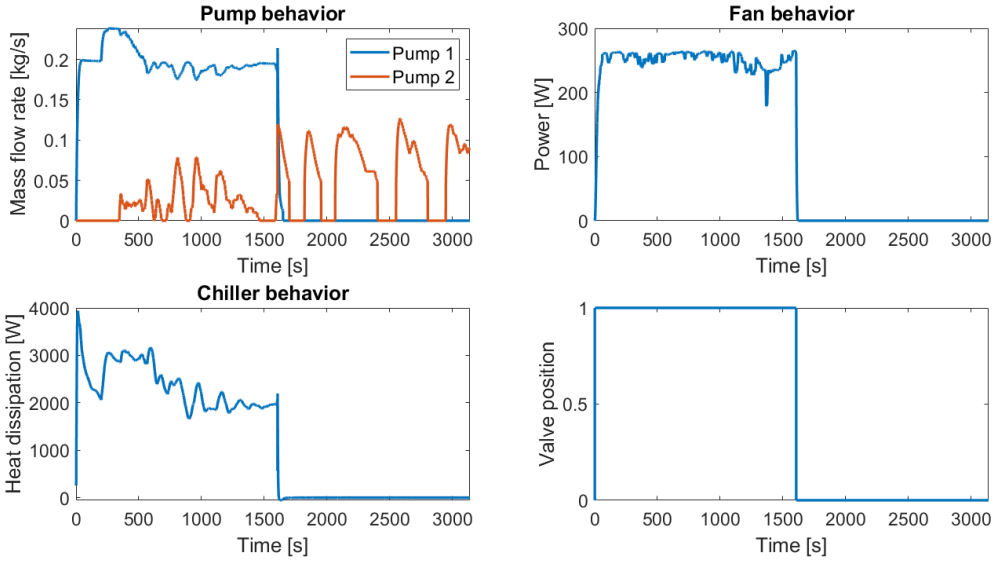


Figure B-5: Resulting control actions using the benchmark control strategy for the hot Artemis use case

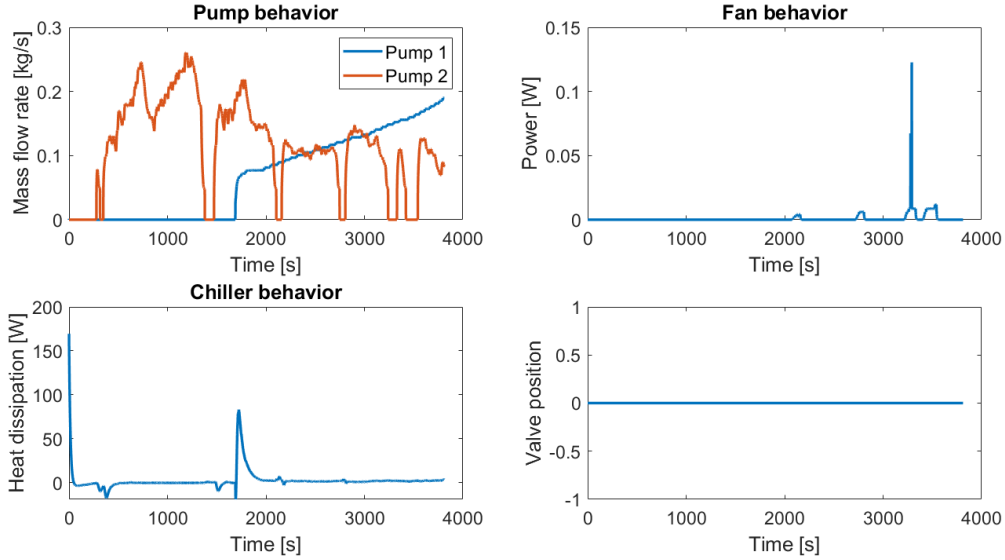


Figure B-6: Resulting control actions using the benchmark control strategy for the hot Stelvio use case

B-1-2 Resulting temperature trajectories

The component temperature trajectories for the WLPT moderate and hot use case are shown in Figure B-7 and Figure B-10 respectively, for the Artemis moderate and hot use case in Figure B-8 and Figure B-11 respectively and finally for the Stelvio moderate and hot use case in Figure B-9 and Figure B-12 respectively.

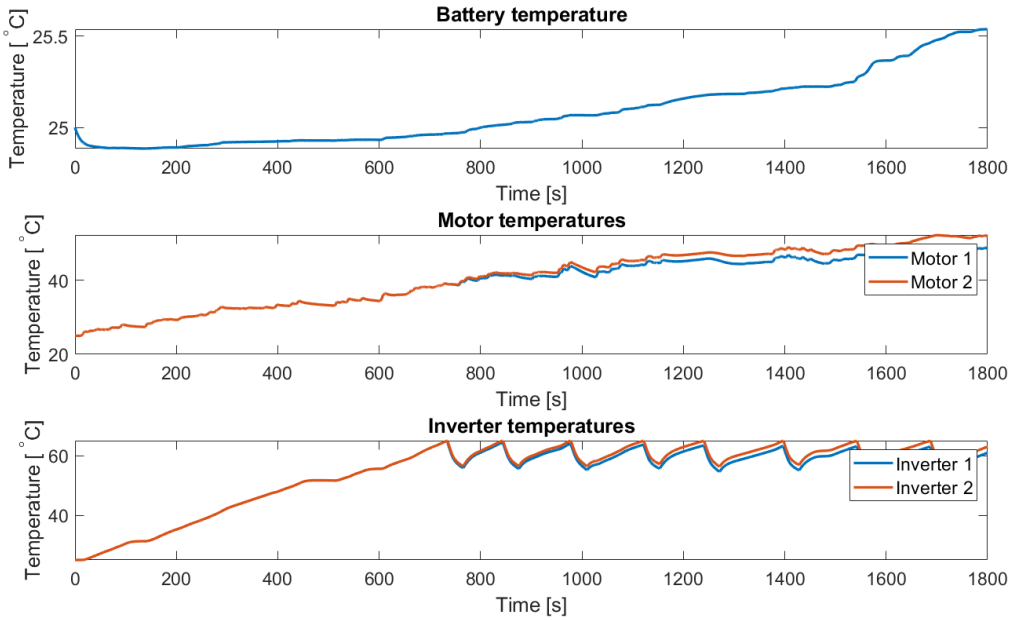


Figure B-7: Resulting component temperature trajectories using the benchmark control strategy for the moderate WLTP use case

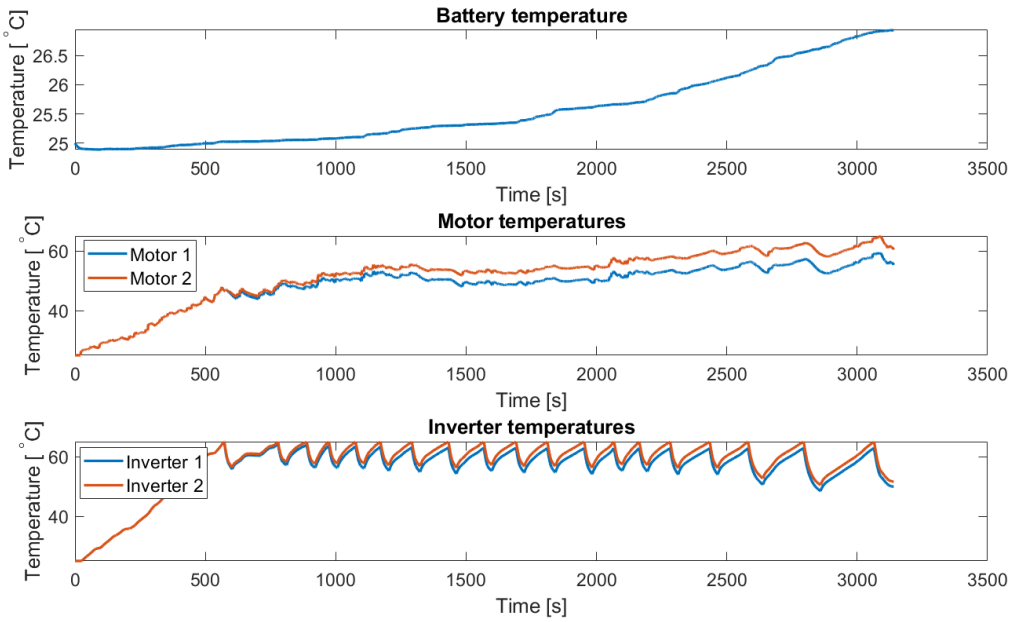


Figure B-8: Resulting component temperature trajectories using the benchmark control strategy for the moderate Artemis use case

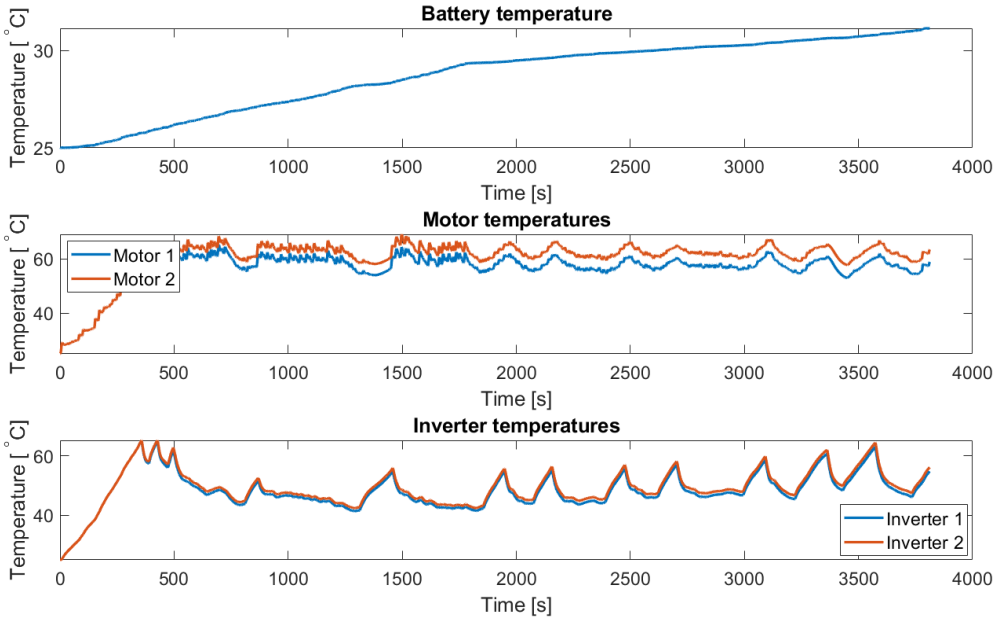


Figure B-9: Resulting component temperature trajectories using the benchmark control strategy for the moderate Stelvio use case

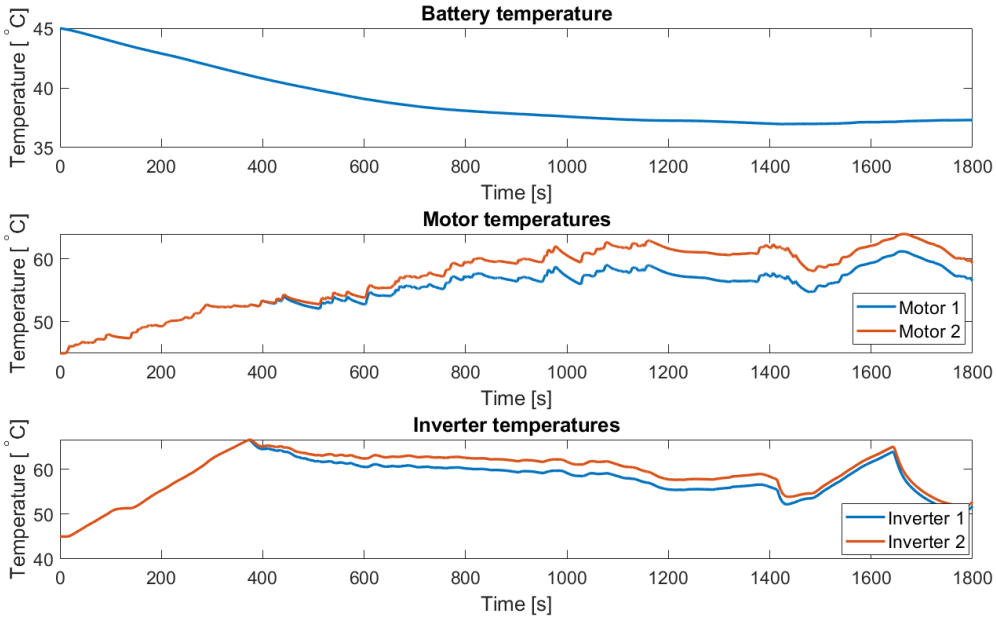


Figure B-10: Resulting component temperature trajectories using the benchmark control strategy for the hot WLTP use case

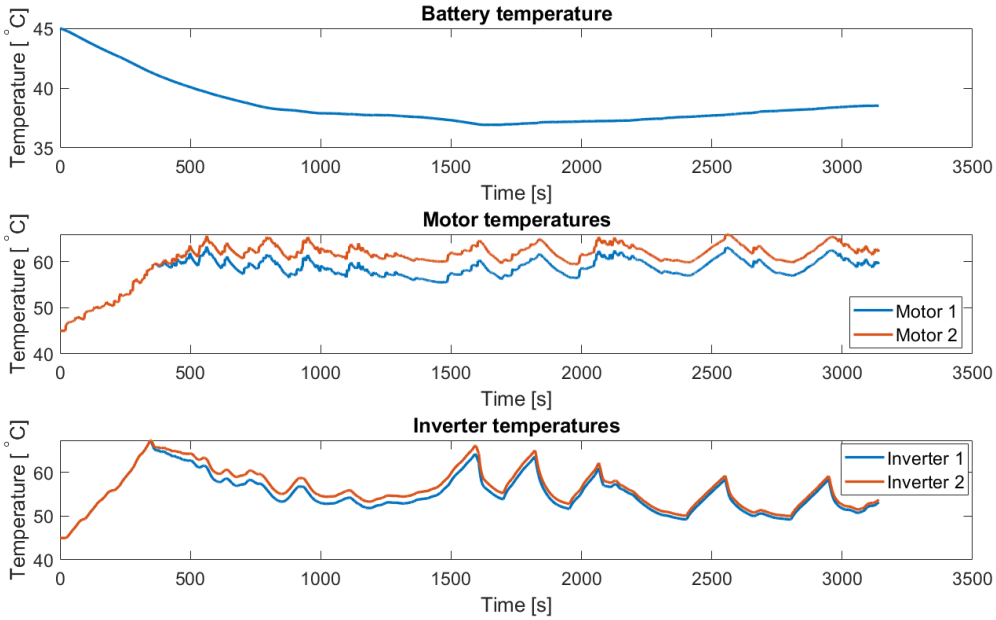


Figure B-11: Resulting component temperature trajectories using the benchmark control strategy for the hot Artemis use case

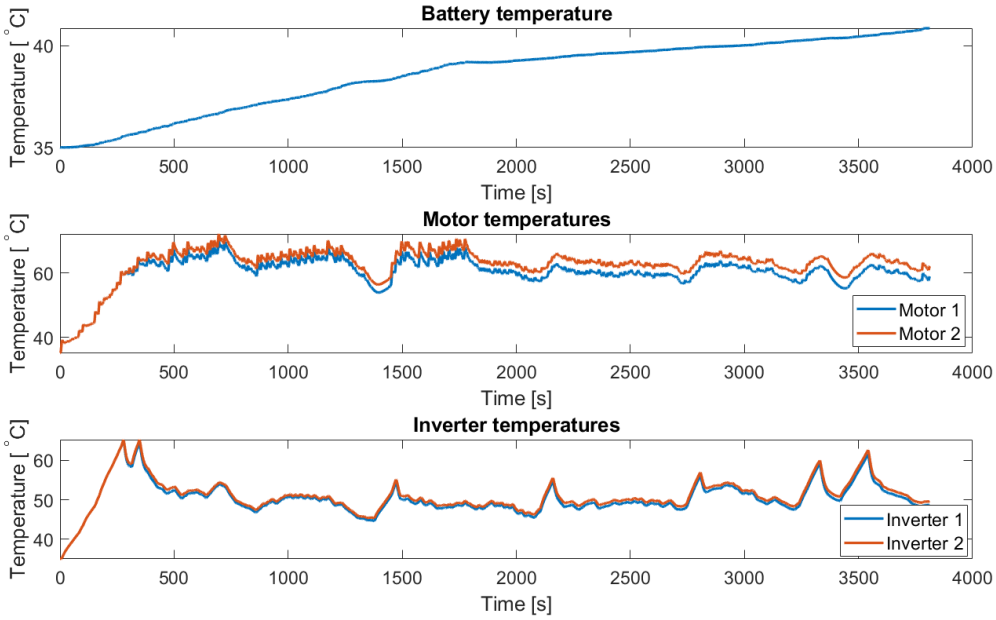


Figure B-12: Resulting component temperature trajectories using the benchmark control strategy for the hot Stelvio use case

B-2 MPC-based controller

For the MPC-based control strategy, first, the control trajectories are shown for each use case. Then the component temperature trajectories are shown, as well as the computation times per drivecycle.

B-2-1 Control behavior

The control trajectories for the WLPT moderate and hot use case are shown in Figure B-13 and Figure B-16 respectively, for the Artemis moderate and hot use case in Figure B-14 and Figure B-17 respectively and finally for the Stelvio moderate and hot use case in Figure B-15 and Figure B-18 respectively.

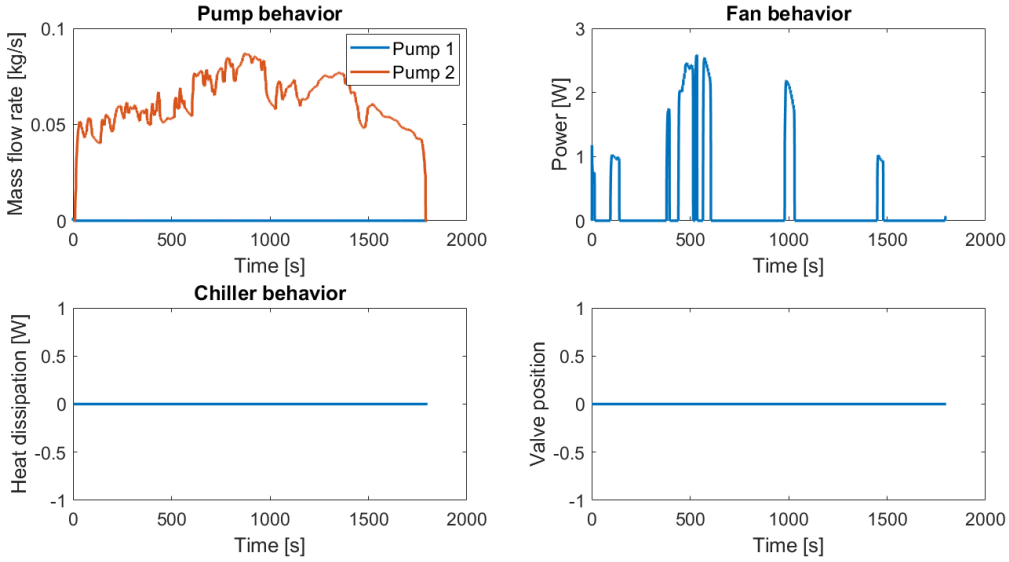


Figure B-13: Resulting control actions using the MPC-based control strategy for the moderate WLTP use case

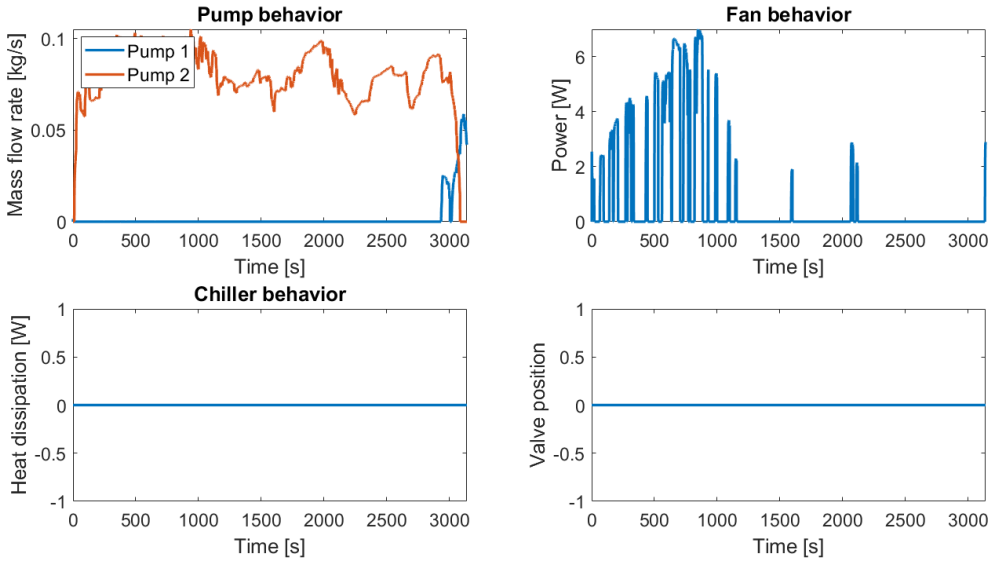


Figure B-14: Resulting control actions using the MPC-based control strategy for the moderate Artemis use case

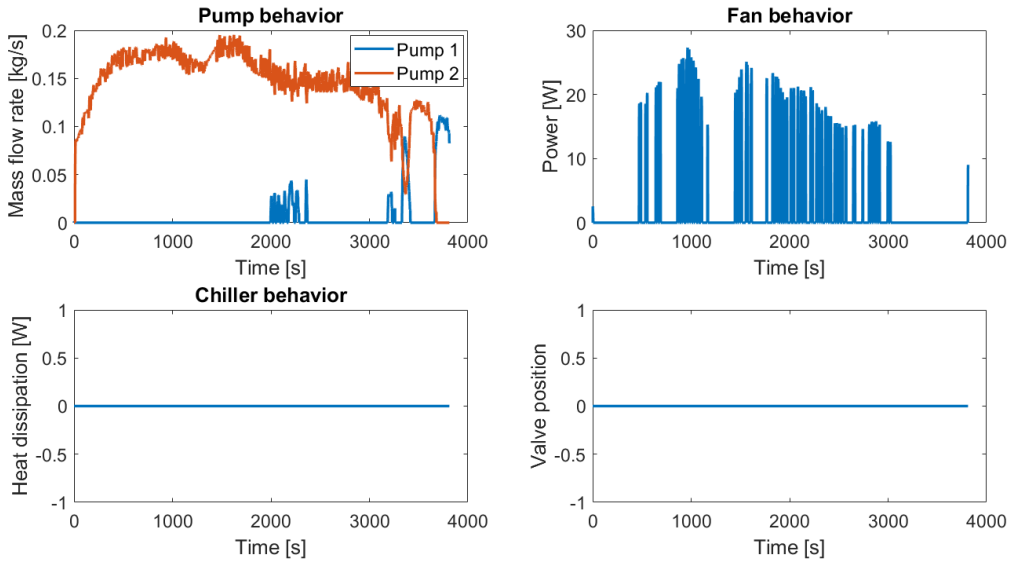


Figure B-15: Resulting control actions using the MPC-based control strategy for the moderate Stelvio use case

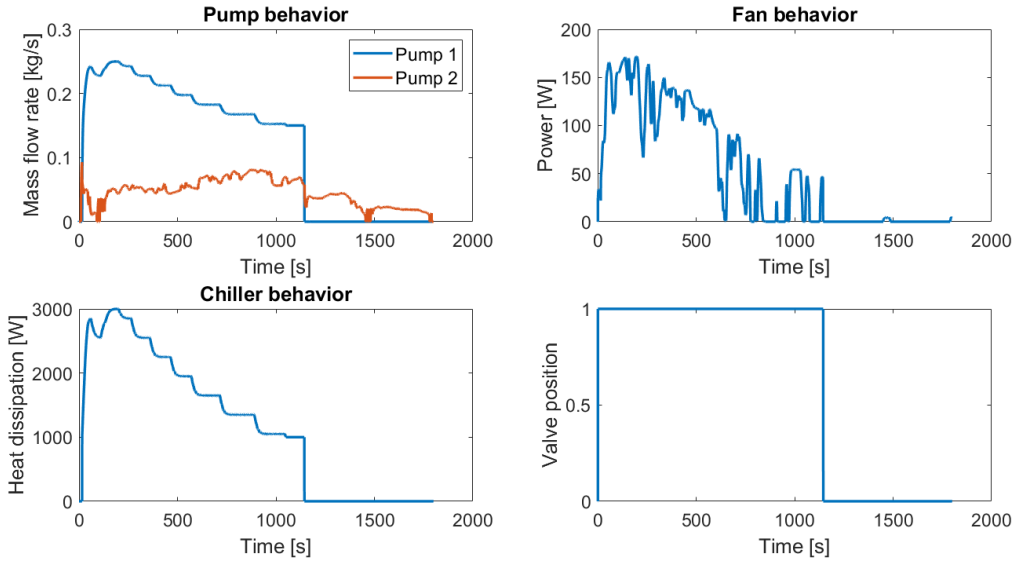


Figure B-16: Resulting control actions using the MPC-based control strategy for the hot WLTP use case

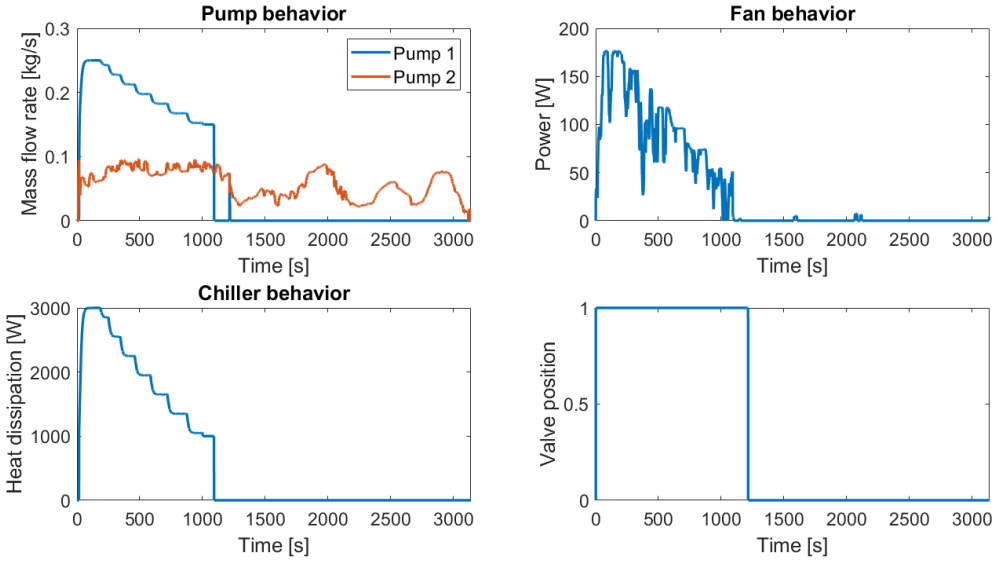


Figure B-17: Resulting control actions using the MPC-based control strategy for the hot Artemis use case

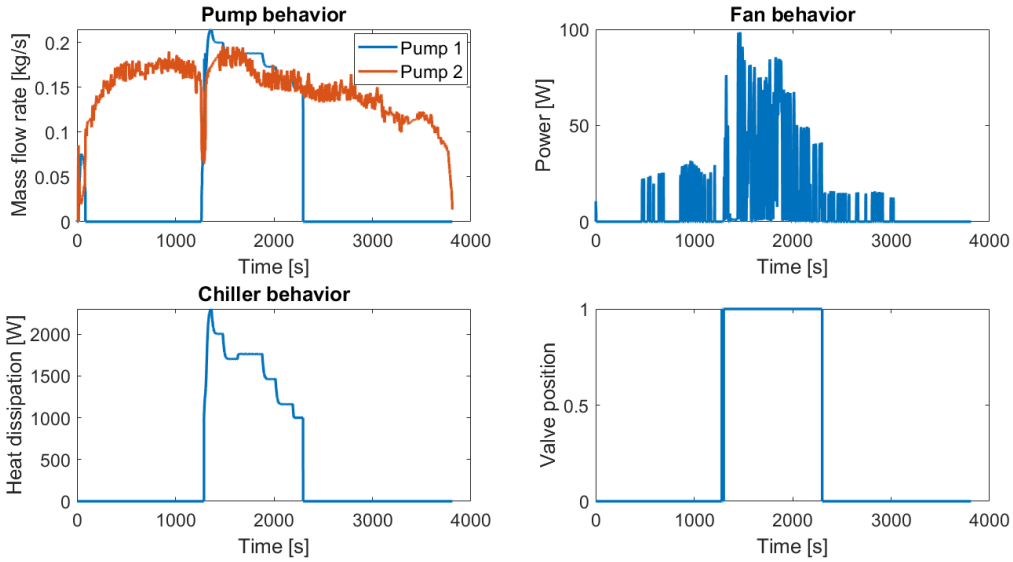


Figure B-18: Resulting control actions using the MPC-based control strategy for the hot Stelvio use case

B-2-2 Resulting temperature trajectories

The control trajectories for the WLPT moderate and hot use case are shown in Figure B-19 and Figure B-22 respectively, for the Artemis moderate and hot use case in Figure B-20 and Figure B-23 respectively and finally for the Stelvio moderate and hot use case in Figure B-21 and Figure B-24 respectively.

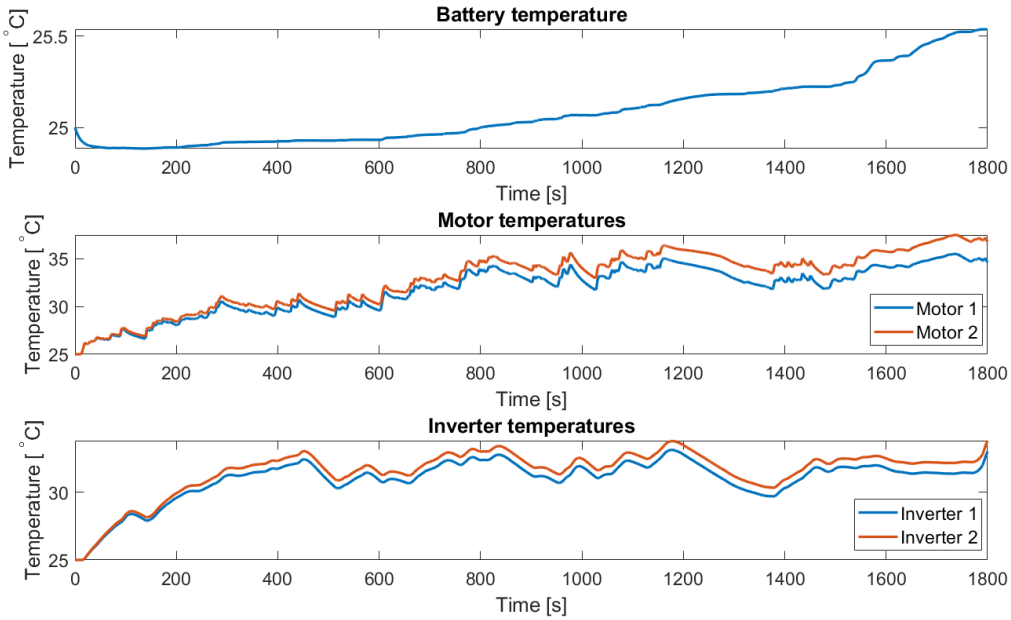


Figure B-19: Resulting component temperature trajectories using the MPC-based control strategy for the moderate WLTP use case

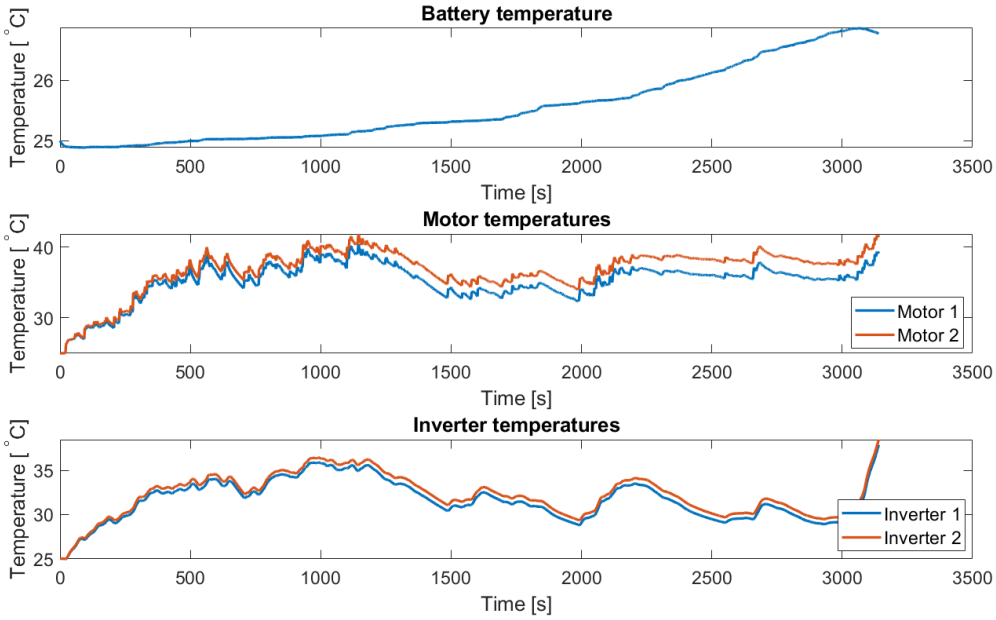


Figure B-20: Resulting component temperature trajectories using the MPC-based control strategy for the moderate Artemis use case

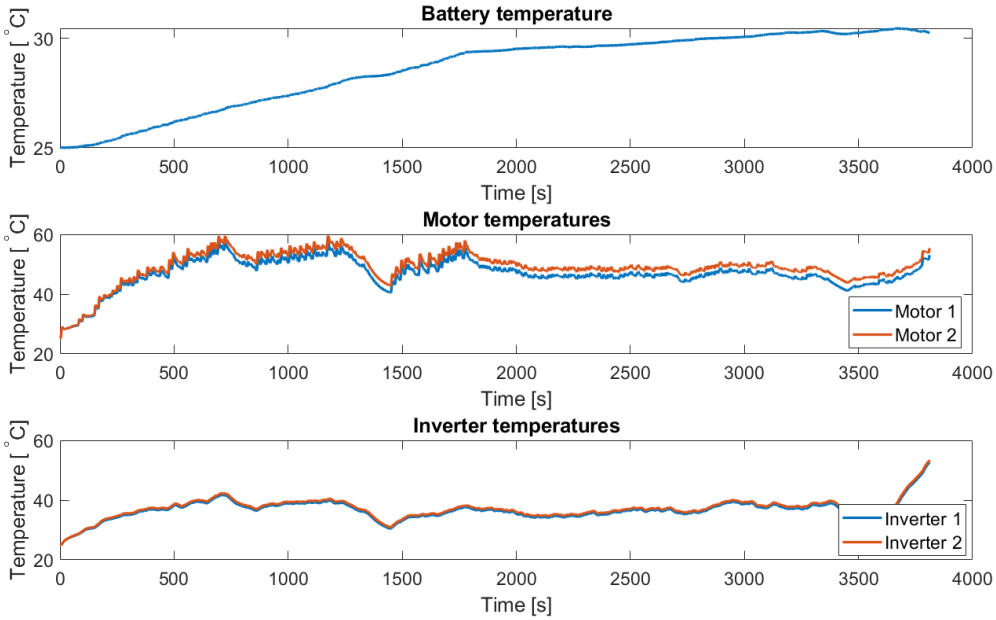


Figure B-21: Resulting component temperature trajectories using the MPC-based control strategy for the moderate Stelvio use case

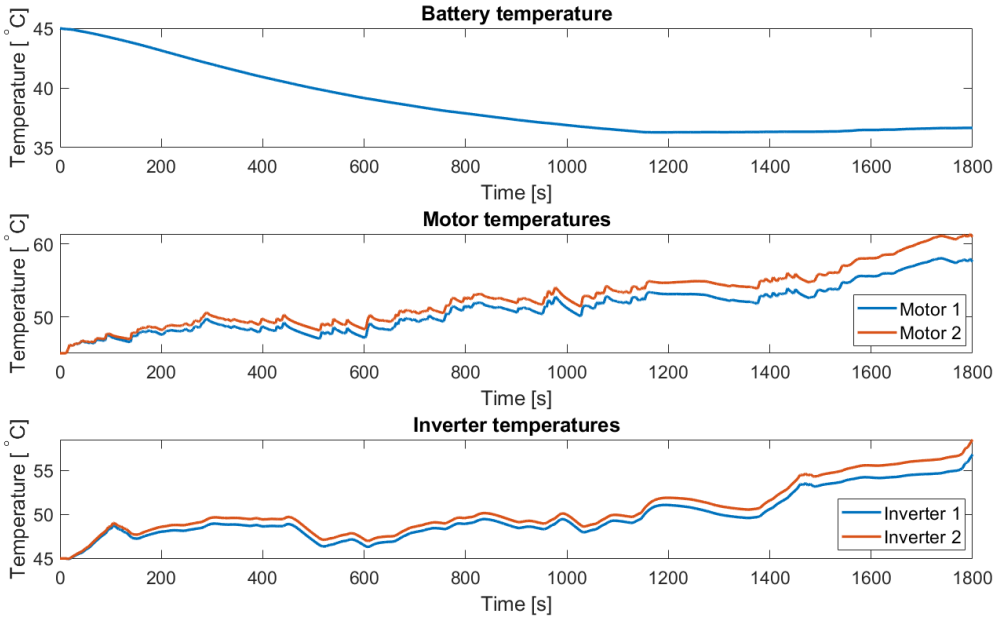


Figure B-22: Resulting component temperature trajectories using the MPC-based control strategy for the hot WLTP use case

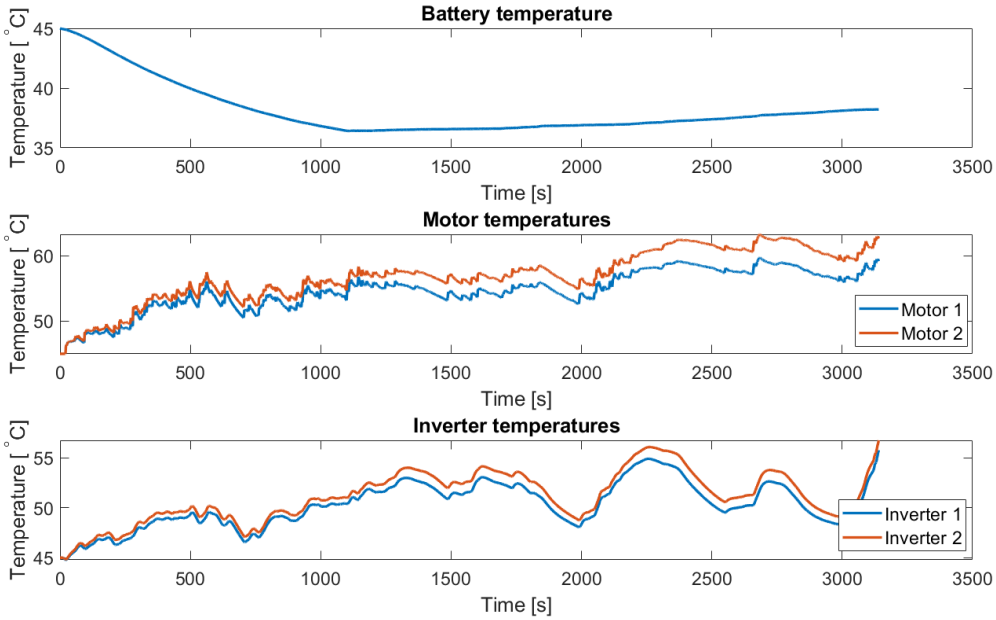


Figure B-23: Resulting component temperature trajectories using the MPC-based control strategy for the hot Artemis use case

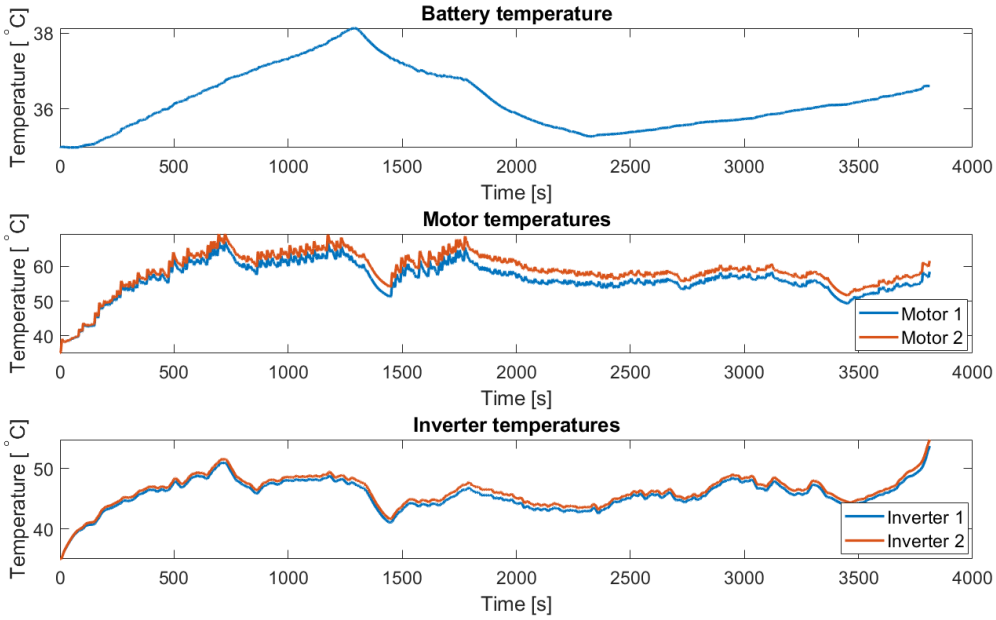
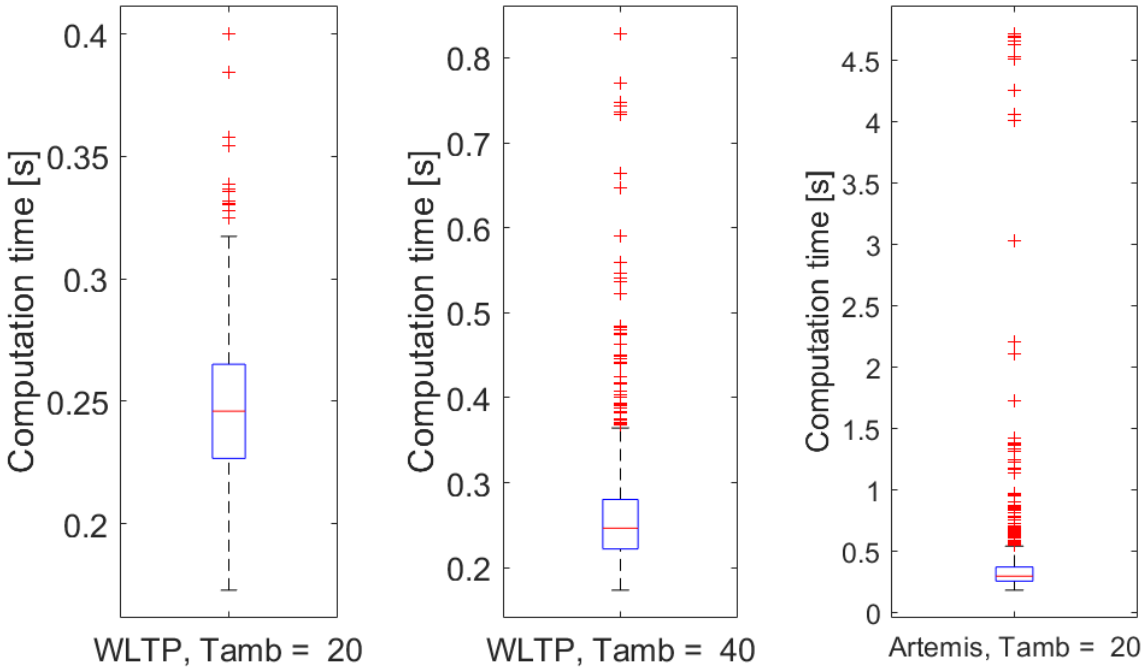


Figure B-24: Resulting component temperature trajectories using the MPC-based control strategy for the hot Stelvio use case

B-2-3 Computation times

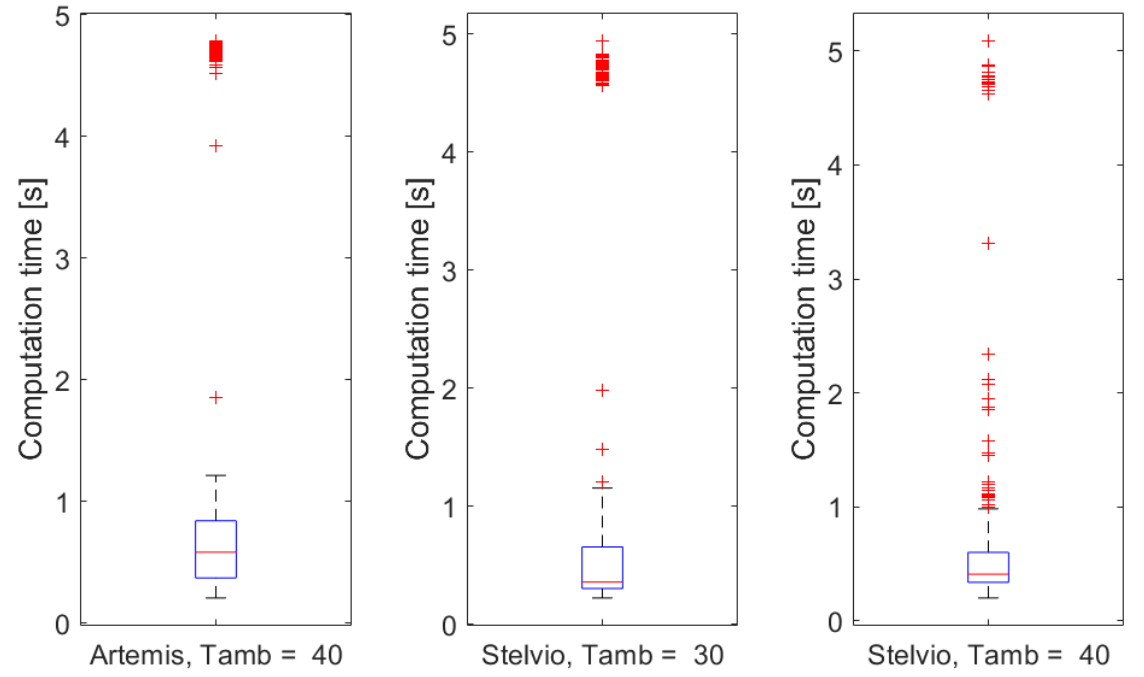
The computation times for the WLPT moderate and hot use case are shown in Figure B-25a and Figure B-25d respectively, for the Artemis moderate and hot use case in Figure B-25b and Figure B-25e respectively and finally for the Stelvio moderate and hot use case in Figure B-25c and Figure B-25f respectively.



(a) Computation times for the MPC-based control strategy for the moderate WLTP use case

(b) Computation times for the MPC-based control strategy for the moderate Artemis use case

(c) Computation times for the MPC-based control strategy for the moderate Stelvio use case



(d) Computation times for the MPC-based control strategy for the hot WLTP use case

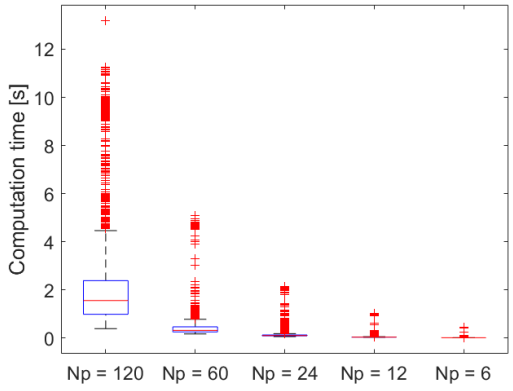
(e) Computation times for the MPC-based control strategy for the hot Artemis use case

(f) Computation times for the MPC-based control strategy for the hot Stelvio use case

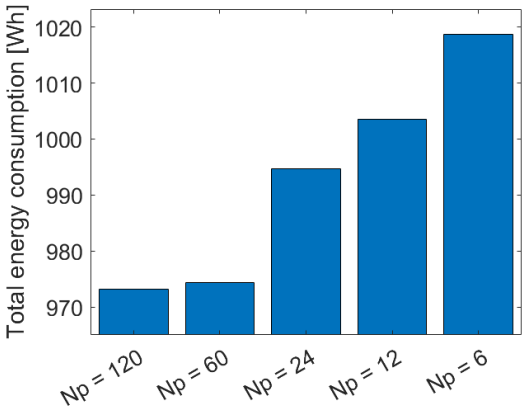
Figure B-25: Computation times for different combinations of integrators and optimizers

B-2-4 Prediction horizon

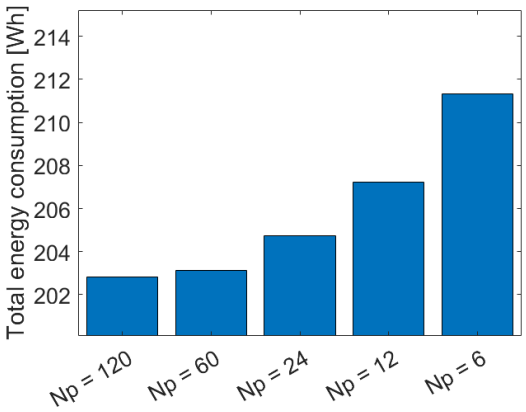
This section shows the complete results elaborating on the relation between the length of the prediction N_p horizon and the computation time as well as the energy consumption of the Thermal Management System (TMS) of the the Electric Vehicle (EV) in each of the three drive cycles. The computation times for different prediction horizons are shown in Figure B-26a. The energy consumption at different lengths of the prediction horizon for the moderate Stelvio, Artemis and WLTP use cases are shown in Figure B-26b, Figure B-26c and Figure B-26d, respectively.



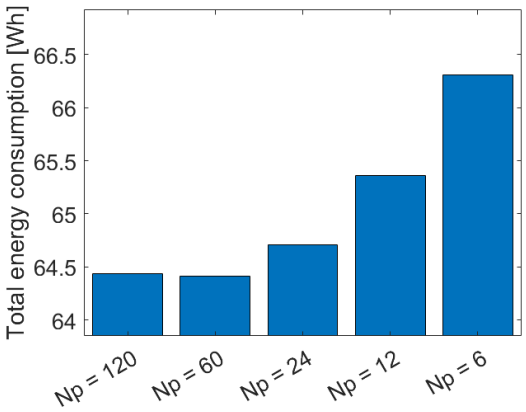
(a) Computation time for different prediction horizons



(b) Energy consumption for the moderate Stelvio use case



(c) Energy consumption for the moderate Artemis use case



(d) Energy consumption for the moderate WLTP use case

Figure B-26: The effect of the prediction horizon on the computation times and the energy consumption for each of the three drivecycles.

Bibliography

[1] J. Lopez-Sanz, C. Ocampo-Martinez, J. Alvarez-Florez, M. Moreno-Eguilaz, R. Ruiz-Mansilla, J. Kalmus, M. Gräeber, and G. Lux, "Nonlinear model predictive control for thermal management in plug-in hybrid electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 3632–3644, 2017.

[2] C. Kirches, *Fast numerical methods for mixed-integer nonlinear model-predictive control*. Springer, 2011.

[3] J. Huang, S. Shoai Naini, R. Miller, D. Rizzo, K. Sebeck, S. Shurin, and J. Wagner, "A hybrid electric vehicle motor cooling system—design, model, and control," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4467–4478, 2019.

[4] M. R. Amini, H. Wang, X. Gong, D. Liao-McPherson, I. Kolmanovsky, and J. Sun, "Cabin and battery thermal management of connected and automated hevs for improved energy efficiency using hierarchical model predictive control," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 5, pp. 1711–1726, 2020.

[5] I. Cvok, B. Škugor, and J. Deur, "Control trajectory optimisation and optimal control of an electric vehicle hvac system for favourable efficiency and thermal comfort," *Optimization and Engineering*, vol. 22, 03 2021.

[6] Y. Xie, C. Wang, X. Hu, X. Lin, Y. Zhang, and W. Li, "An mpc-based control strategy for electric vehicle battery cooling considering energy saving and battery lifespan," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 14657–14673, 2020.

[7] S. Zhao, M. R. Amini, J. Sun, and C. C. Mi, "A two-layer real-time optimization control strategy for integrated battery thermal management and hvac system in connected and automated hevs," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 7, pp. 6567–6576, 2021.

[8] M. Azhar Khan, M. Zahir Khan, K. Zaman, and L. Naz, "Global estimates of energy consumption and greenhouse gas emissions," *Renewable and Sustainable Energy Reviews*, vol. 29, pp. 336–344, 2014.

[9] S. Manabe, "Role of greenhouse gas in climate change**," *Tellus A: Dynamic Meteorology and Oceanography*, vol. 71, no. 1, p. 1620078, 2019.

[10] W. F. Lamb, T. Wiedmann, J. Pongratz, R. Andrew, M. Crippa, J. G. J. Olivier, D. Wiedenhofer, G. Mattioli, A. A. Khourdajie, J. House, S. Pachauri, M. Figueroa, Y. Saheb, R. Slade, K. Hubacek, L. Sun, S. K. Ribeiro, S. Khennas, S. de la Rue du Can, L. Chapungu, S. J. Davis, I. Bashmakov, H. Dai, S. Dhakal, X. Tan, Y. Geng, B. Gu, and J. Minx, "A review of trends and drivers of greenhouse gas emissions by sector from 1990 to 2018," *Environmental Research Letters*, vol. 16, p. 073005, jun 2021.

[11] IEA, "Global ev outlook," 2020.

[12] G. Cecere, N. Corrocher, and M. Guerzoni, "Price or performance? a probabilistic choice analysis of the intention to buy electric vehicles in european countries," *Energy Policy*, vol. 118, pp. 19–32, 2018.

[13] M. A. Jeffers, L. Chaney, and J. P. Rugh, "Climate control load reduction strategies for electric drive vehicles in cold weather," *SAE International Journal of Passenger Cars - Mechanical Systems*, vol. 9, pp. 75–82, apr 2016.

[14] S. Schaut, E. Arnold, and O. Sawodny, "Predictive thermal management for an electric vehicle powertrain," *IEEE Transactions on Intelligent Vehicles*, 2021.

[15] T. Zhang, C. Gao, Q. Gao, G. Wang, M. Liu, Y. Guo, C. Xiao, and Y. Yan, "Status and development of electric vehicle integrated thermal management from btm to hvac," *Applied Thermal Engineering*, vol. 88, pp. 398–409, 2015. Special Issue for International Heat Transfer Symposium 2014.

[16]

[17] H. Zou, B. Jiang, Q. Wang, C. Tian, and Y. Yan, "Performance analysis of a heat pump air conditioning system coupling with battery cooling for electric vehicles," *Energy Procedia*, vol. 61, pp. 891–894, 2014. International Conference on Applied Energy, ICAE2014.

[18] S. Hemmati, N. Doshi, D. Hanover, C. Morgan, and M. Shahbakhti, "Integrated cabin heating and powertrain thermal energy management for a connected hybrid electric vehicle," *Applied Energy*, vol. 283, p. 116353, 2021.

[19] T. Zhang, C. Gao, Q. Gao, G. Wang, M. Liu, Y. Guo, C. Xiao, and Y. Yan, "Status and development of electric vehicle integrated thermal management from btm to hvac," *Applied Thermal Engineering*, vol. 88, pp. 398–409, 2015. Special Issue for International Heat Transfer Symposium 2014.

[20] R. Eini and S. Abdelwahed, "Learning-based model predictive control for smart building thermal management," in *2019 IEEE 16th International Conference on Smart Cities: Improving Quality of Life Using ICT IoT and AI (HONET-ICT)*, pp. 038–042, 2019.

[21] D. Sturzenegger, D. Gyalistras, M. Morari, and R. S. Smith, "Model predictive climate control of a swiss office building: Implementation, results, and cost–benefit analysis," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 1, pp. 1–12, 2016.

[22] F. Oldewurtel, A. Parisio, C. N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and M. Morari, "Use of model predictive control and weather forecasts for energy efficient building climate control," *Energy and Buildings*, vol. 45, pp. 15–27, 2012.

[23] Y. Zhang, L. Chu, Z. Fu, N. Xu, C. Guo, X. Zhang, Z. Chen, and P. Wang, "Optimal energy management strategy for parallel plug-in hybrid electric vehicle based on driving behavior analysis and real time traffic information prediction," *Mechatronics*, vol. 46, pp. 177–192, 10 2017.

[24] T. Sun, R. Yang, H. Li, X. Zhang, and T. Xu, "Active motor rotor temperature management based on one-node thermal network model predictive control," *IEEE Transactions on Power Electronics*, vol. 35, no. 10, pp. 11213–11221, 2020.

[25] A. Wahl, C. Wellmann, B. Krautwig, P. Manns, B. Chen, C. Schernus, and J. Andert, "Efficiency increase through model predictive thermal control of electric vehicle powertrains," *Energies*, vol. 15, no. 4, 2022.

[26] S. Ma, M. Jiang, P. Tao, C. Song, J. Wu, J. Wang, T. Deng, and W. Shang, "Temperature effect and thermal impact in lithium-ion batteries: A review," *Progress in Natural Science: Materials International*, vol. 28, no. 6, pp. 653–666, 2018.

[27] J. Wang, S. Li, H. Chen, Y. Yuan, and Y. Huang, "Data-driven model predictive control for building climate control: Three case studies on different buildings," *Building and Environment*, vol. 160, p. 106204, 2019.

[28] N. S. Nandagopal, *Fluid and Thermal Sciences*. Springer, 2022.

[29] S. Park and C. Ahn, "Computationally efficient stochastic model predictive controller for battery thermal management of electric vehicle," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8407–8419, 2020.

[30] Y. Ma, H. Ding, H. Mou, and J. Gao, "Battery thermal management strategy for electric vehicles based on nonlinear model predictive control," *Measurement*, vol. 186, p. 110115, 2021.

[31] N. Javani, I. Dincer, and G. Naterer, "New latent heat storage system with nanoparticles for thermal management of electric vehicles," *Journal of Power Sources*, vol. 268, pp. 718–727, 2014.

[32] L. Cabezas-Gómez, H. A. Navarro, and J. M. Saíz-Jabardo, *Thermal performance modeling of cross-flow heat exchangers*. Springer, 2015.

[33] Y. Masoudi and N. L. Azad, "Mpc-based battery thermal management controller for plug-in hybrid electric vehicles," in *2017 American Control Conference (ACC)*, pp. 4365–4370, 2017.

[34] X. Kuang, K. Li, Y. Xie, C. Wu, P. Wang, X. Wang, and C. Fu, "Research on control strategy for a battery thermal management system for electric vehicles based on secondary loop cooling," *IEEE Access*, vol. 8, pp. 73475–73493, 2020.

[35] I. Ratković, I. Cvok, V. Soldo, and J. Deur, "Control-oriented modelling of vapour compression cycle including model-order reduction and analysis tools," in *14th SDEWES conference, Dubrovnik*, 2019.

[36] D. Rong, B. Yang, and C. Chen, "Model predictive climate control of electric vehicles for improved battery lifetime," in *2019 Chinese Automation Congress (CAC)*, pp. 5457–5462, 2019.

[37] H. He, H. Jia, C. Sun, and F. Sun, "Stochastic model predictive control of air conditioning system for electric vehicles: Sensitivity study, comparison, and improvement," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4179–4189, 2018.

[38] S. Schaut and O. Sawodny, "Thermal management for the cabin of a battery electric vehicle considering passengers' comfort," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 4, pp. 1476–1492, 2020.

[39] M. Alizadeh, S. Dhale, and A. Emadi, "Model predictive control of hvac system in a battery electric vehicle with fan power adaptation for improved efficiency and online estimation of ambient temperature," in *IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society*, pp. 1–6, 2021.

[40] J. Glos, L. Otava, and P. Václavek, "Non-linear model predictive control of cabin temperature and air quality in fully electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 2, pp. 1216–1229, 2021.

[41] M. S. Elliott and B. P. Rasmussen, "Decentralized model predictive control of a multi-evaporator air conditioning system," *Control Engineering Practice*, vol. 21, no. 12, pp. 1665–1677, 2013.

[42] C. Zhu, F. Lu, H. Zhang, J. Sun, and C. C. Mi, "A real-time battery thermal management strategy for connected and automated hybrid electric vehicles (cahevs) based on iterative dynamic programming," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8077–8084, 2018.

[43] S. Bauer, A. Suchaneck, and F. Puente León, "Thermal and energy battery management optimization in electric vehicles using pontryagin's maximum principle," *Journal of Power Sources*, vol. 246, pp. 808–818, 2014.

[44] S. Atalay, M. Sheikh, A. Mariani, Y. Merla, E. Bower, and W. D. Widanage, "Theory of battery ageing in a lithium-ion battery: Capacity fade, nonlinear ageing and lifetime prediction," *Journal of Power Sources*, vol. 478, p. 229026, 2020.

[45] E. Redondo-Iglesias, P. Venet, and S. Pelissier, "Calendar and cycling ageing combination of batteries in electric vehicles," *Microelectronics Reliability*, vol. 88-90, pp. 1212–1215, 2018. 29th European Symposium on Reliability of Electron Devices, Failure Physics and Analysis (ESREF 2018).

[46] T. Waldmann, M. Wilka, M. Kasper, M. Fleischhammer, and M. Wohlfahrt-Mehrens, "Temperature dependent ageing mechanisms in lithium-ion batteries – a post-mortem study," *Journal of Power Sources*, vol. 262, pp. 129–135, 2014.

[47] M. Lucu, E. Martinez-Laserna, I. Gandiaga, K. Liu, H. Camblong, W. Widanage, and J. Marco, "Data-driven nonparametric li-ion battery ageing model aiming at learning from real operation data – part a: Storage operation," *Journal of Energy Storage*, vol. 30, p. 101409, 2020.

[48] J. Schmalstieg, S. Käbitz, M. Ecker, and D. U. Sauer, "From accelerated ageing tests to a lifetime prediction modhorizonel: Analyzing lithium-ion batteries," in *2013 World Electric Vehicle Symposium and Exhibition (EVS27)*, pp. 1–12, 2013.

[49] M. Ecker, J. B. Gerschler, J. Vogel, S. Käbitz, F. Hust, P. Dechent, and D. U. Sauer, "Development of a lifetime prediction model for lithium-ion batteries based on extended accelerated aging test data," *Journal of Power Sources*, vol. 215, pp. 248–257, 2012.

[50] J. Wang, P. Liu, J. Hicks-Garner, E. Sherman, S. Soukiazian, M. Verbrugge, H. Tataria, J. Musser, and P. Finamore, "Cycle-life model for graphite-lifepo4 cells," *Journal of Power Sources*, vol. 196, no. 8, pp. 3942–3948, 2011.

[51] S. Xie, X. Hu, Z. Xin, and J. Brighton, "Pontryagin's minimum principle based model predictive control of energy management for a plug-in hybrid electric bus," *Applied Energy*, vol. 236, pp. 893–905, 2019.

[52] Y. Xie, Z. Liu, K. Li, J. Liu, Y. Zhang, D. Dan, C. Wu, P. Wang, and X. Wang, "An improved intelligent model predictive controller for cooling system of electric vehicle," *Applied Thermal Engineering*, vol. 182, p. 116084, 2021.

[53] S. P. Sethi, *Optimal Control Theory*. Springer, 2021.

[54] J. H. Lee, "Model predictive control and dynamic programming," in *2011 11th International Conference on Control, Automation and Systems*, pp. 1807–1809, 2011.

[55] L. Grüne, J. Pannek, L. Grüne, and J. Pannek, *Nonlinear model predictive control*. Springer, 2017.

[56] F. Oldewurtel, A. Parisio, C. N. Jones, M. Morari, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and K. Wirth, "Energy efficient building climate control using stochastic model predictive control and weather predictions," in *Proceedings of the 2010 American Control Conference*, pp. 5100–5105, 2010.

[57] H. He, M. Yan, C. Sun, J. Peng, M. Li, and H. Jia, "Predictive air-conditioner control for electric buses with passenger amount variation forecast," *Applied Energy*, vol. 227, pp. 249–261, 2018. Transformative Innovations for a Sustainable Future – Part III.

[58] J. Glos, F. Šolc, L. Otava, and P. Václavek, "Hybrid model predictive control for fully electric vehicle thermal management system optimal mode selection," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, pp. 2036–2043, 2020.

[59] J. Lopez-Sanz, C. Ocampo-Martinez, J. Álvarez Flórez, M. Moreno-Eguilaz, R. Ruiz-Mansilla, J. Kalmus, M. Gräeber, and G. Lux, "Thermal management in plug-in hybrid electric vehicles: A real-time nonlinear model predictive control implementation," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 9, pp. 7751–7760, 2017.

[60] Y. Liu and J. Zhang, "Self-adapting j-type air-based battery thermal management system via model predictive control," *Applied Energy*, vol. 263, p. 114640, 2020.

[61] S. Park and C. Ahn, "Model predictive control with stochastically approximated cost-to-go for battery cooling system of electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 5, pp. 4312–4323, 2021.

[62] M. R. Amini, I. Kolmanovsky, and J. Sun, "Hierarchical mpc for robust eco-cooling of connected and automated vehicles and its application to electric vehicle battery thermal management," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 1, pp. 316–328, 2021.

[63] M. Hermann and M. Saravi, *Nonlinear ordinary differential equations*. Springer, 2016.

[64] J. C. Butcher, *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2016.

[65] J. Nocedal and S. J. Wright, *Numerical optimization*. Springer, 1999.

[66] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.

[67] Q. Hu, M. R. Amini, I. Kolmanovsky, J. Sun, A. Wiese, and J. B. Seeds, "Multihorizon model predictive control: An application to integrated power and thermal management of connected hybrid electric vehicles," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 3, pp. 1052–1064, 2022.

[68] H. Wang, Y. Meng, Q. Zhang, M. R. Amini, I. Kolmanovsky, J. Sun, and M. Jennings, "Mpc-based precision cooling strategy (pcs) for efficient thermal management of automotive air conditioning system," in *2019 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 573–578, 2019.

[69] A. Zanelli, A. Domahidi, J. Jerez, and M. Morari, "Forces nlp: an efficient implementation of interior-point methods for multistage nonlinear nonconvex programs," *International Journal of Control*, vol. 93, no. 1, pp. 13–29, 2020.

[70] "Long range solar electric vehicle - lightyear 0."

Glossary

List of Acronyms

EV	Electric Vehicle
TMS	Thermal Management System
HVAC	Heating, Ventilation and Air Conditioning
HVB	High Voltage Battery
FHU	Front Hex Unit
MPC	Model Predictive control
NN	Neural Network
DP	Dynamic Programming
SoC	State of Charge
SoH	State of Health
COP	Coefficient Of Performance
PLR	Partial Load Ratio
OCP	Optimal Control Problem
NTU	Number of Transfer Units
LMTD	Log-Mean Temperature Difference
BDF	Backward Differentiation Formula
RK	Runge-Kutta
SEI	Solid Electrolyte Interface
MLD	Mixed Logical Dynamical
ODE	Ordinary Differential Equation
IVP	Initial Value Problem
SQP	Sequential Quadratic Programming
NLP	Nonlinear Programming

QP	Quadratic Programming
PSO	Particle Swarm Optimization
LUT	Look-Up Table
BVP	Boundary Value Problem
LTE	Local Truncation Error
GE	Global Error

List of Symbols

Abbreviations

α_{road}	Road slope	
$\dot{m}$	Mass flow rate	kg s^{-1}
$\dot{Q}$	Heat flow rate	W
$\dot{V}$	Volumetric flow rate	$m^3 s^{-1}$
ϵ	Heat transfer efficiency coefficient	-
η_{dt}	Drivetrain efficiency	p.u.
μ	Dynamic Viscosity	Pa $\cdot s$
ω	Rotational speed	rad s^{-1}
ρ	Density	kg m^{-3}
A	Area	m^2
a	Acceleration	m s^{-2}
C	Thermal capacity	J K^{-1}
c	Specific heat capacity	J $K^{-1} kg^{-1}$
C_{d}	Aerodynamic drag coefficient	-
C_{lost}	Battery capacity fade	Wh
C_{original}	Original unused battery capacity	Wh
c_{rr}	Rolling resistance coefficient	-
C_{used}	Remaining attery capacity after usage	Wh
C_r	Heat capacity rate	W K^{-1}
D_H	Hydraulic Diameter	m
E	Energy	J
F_{accel}	Inertial force	N
F_{aero}	Aerodynamic drag force	N
F_{grav}	Gravitational force	N
F_{roll}	Rolling drag force	N
F_{tr}	Total traction force	N
g	Gravitational acceleration	m/s^2
H	Enthalpy	J

h	Step size	s
I	Electric Current	A
k_b	Boltzmann constant	J K^{-1}
m	Mass	kg
n	Speed	s^{-1}
Nu	Nusselt number	-
P	Power consumption	W
p	Absolute pressure	Pa
P_{pr}	Vehicle propulsion power	W
R_0	Resistance at reference temperature	Ω
R_{aged}	Aged battery cell resistance	Ω
R_{bat}	Battery pack resistance	Ω
R_{cell}	Battery cell resistance	Ω
R_{gas}	Specific gas constant	J $K^{-1}mol^{-1}$
Re	Reynolds number	-
T	Temperature	°C
T_0	Reference temperature	°C
U	Heat transfer coefficient	W $m^{-2}K^{-1}$
v	Velocity	m s^{-1}
w	Valve position	-
τ	Torque	Nm
Q	Heat	J
t	Time	s
t_s	Sample time	s
V_{oc}	Battery open circuit voltage	V
COP	Coefficient of performance	-
NTU	Number of Transfer Units	-
PLR	Partial Load Ratio	-
SoC	State of Charge	%
SoH	State of Health	%