University of
South-Eastern Norway

www.usn.no

FMH606 Master's Thesis 2022
Industrial IT and Automation Engineering

An approach to optimal control of snow melting systems

Tim Cato Lybekk

Faculty of Technology, Natural Sciences and Maritime Sciences
Campus Porsgrunn

University of
South-Eastern Norway

www.usn.no

Course: FMH606 Master's Thesis 2022
Title: *An approach to optimal control of snow melting systems*
Pages: *118*
Keywords: *MPC, Snow melting, Simulink, Optimization, Model Fitting, First order, Weather Predictions*

Student: *Tim Cato Lybekk*
Supervisor: *Carlos Pfeiffer*
External partner: *Schneider Electric , Drammen Eiendom*

Summary:
Snow melting systems are getting more and more common for keeping pavements and parking lots free of snow and ice, and are usually installed in relation to Heating, Ventilation, and Air Conditioning(HVAC) systems in commercial buildings. These systems are highly power consuming compared to other HVAC system and usually only active for shorter periods of time. The snow melting system used as a basis for the research of this project is located at Åskollen, Drammen, and is managed by Drammen Eiendom. The system is currently controlled using traditional PID controllers organised in a Cascade manner and in combination with logical control and weather predictions. Due to the high power consumption it is desirable to optimize the control of the snow melting system. A feasibility study of the possibility to develop a Model Predictive Control(MPC) for controlling the snow melting system is performed in this project. Several mathematical models for describing the behavior of the snow melting system is presented, and one is chosen for use in the development of the MPC. The optimization problem is defined to give the desired output, with the energy usage as a minimization objective and the pavement surface temperature as a lower constraint. The MPC is developed using MathWorks Matlab Simulink, and tested in the same environment. In total 12 test cases have been simulated, with variation on weather prediction and initial temperatures of the system. It has been concluded that the MPC results in overall better control than the existing PID control, when considering energy usage and overshoot of temperature. The implementation cost is found to be too high when considering only using the developed MPC for one site, but when considered for multiple sites the implementation cost is justifiable.

Preface

This thesis is the work of Tim Cato Lybekk, student at the University of South-Eastern Norway. The thesis presents the work from a Masters project performed in collaboration with Drammen Eiendom. This thesis is written for readers familiar with automation and computer science. It is also expected for the reader to have some familiarity with dynamic processes. The computer software used during the course of the project is Matlab Simulink for development and testing, Schneider Electric EcoStruxure for obtaining data, MS Excel for handling and analysing data and Overleaf for text editing.

The task description that forms the grounds for the project is attached as Appendix A

A special thanks to: Rune Simensen, chief engineer at Drammen Eiendom for making data on the snow melting system available, and for allowing testing on the system. Henrik Guneriussen and Sara Saade, colleagues at Schneider Electric, for throwing ideas back and forth. Walter Johansson, fellow student, for helping refresh on former common subjects relevant to the project. Finally a thanks to Carlos Pfeiffer, professor at USN, for supervision during the project.

Drammen, 16th May 2022

Tim Cato Lybekk

Contents

List of Figures

List of Tables

Nomenclature

Symbol	Explanation
API	Application Programming Interface
AS	Automation Server
COV	Change Of Value
ES	Enterprise Serve
FOPDT	First Order Plus Dead Time
HVAC	Heating, Ventilation, and Air Conditioning
I/O	Input/Output
MET	Meteorological Institute
MOOP	Multi Objective Optimization Problem
MPC	Model Predictive Control
MQTT	Message Queuing Telemetry Transport
NTC	Negative Temperature Coefficient
OF	Objective Function
OP	Optimizing Problem
PID	Proportional–Integral–Derivative
PLC	Programmable Logical Controller
QP	Quadratic Programming
SOPDT	Second Order Plus Dead Time
SQPF	Standard Quadratic Programming Formulation
.csv	File Extension for Comma Separated Values
.xls	File Extension for Microsoft Excel Spreadsheet

1 Introduction

An introduction to the background of this project is given in this chapter. First it is argued why a new approach for control of snow melting systems should be considered, then a description of the system in question is presented, furthermore the existing control system is described, finally the objectives and project goals are presented, at last the structure of the report is given.

1.1 Background

In Heating Ventilation and Air Conditioning(HVAC) systems water-to-water exchangers is a common and central component. Water-to-water exchangers is used in systems ranging from sanitary equipment, where it can be used to separate heating medium from drinking water, to cooling systems for comfort temperature control. The number of water-to-water exchangers present in a HVAC system is dependent on the design of the system. Simpler systems may have no exchangers, for example a water heating system with an electric boiler and a single circuit. For more complex systems there might be a need for one or more water-to-water exchangers, for example a water heating system consisting of a district heating supply and two heating circuits. In this example one water-to-water exchanger can be used to separate the district heating from the local system, and another to separate the main circuit from one of the heating circuits due to different temperature operating points.

The wide use of water-to-water exchangers in HVAC systems raises the potential of a smoother operation and potential economical saving with improved control.

Water-to-water exchangers are a common component in snow melting systems that are liquid based. More accurately, water-to-alcohol exchangers, as the liquid on the melting side of the exchange needs to be frost proof. These snow melting system consume a lot of power, and therefore controlling them in an optimal way is desirable. Snow melting works by circulating heated alcohol in the ground beneath the pavement to melt the snow/ice on the surface. The need of a snow melting system being active is dependent on the weather, but not only by whether it snows but also on the temperature and air humidity. As the cost of running a snow melting system is high, the time such a system is active is desired to be kept at a minimum, and to run it in an optimal manner.

According to a study done on the energy consumption for snow melting system conducted on locations in Oslo, Trondheim and Drammen, the power needed is between $175W/m^2$ and $350W/m^2$[1]. When the system in question was designed, the needed power was assumed to be about $250W/m^2$. The snow melting system in question is designed to deliver $27,5kW$ of power over a section of about $110m^2$. Compared to other heating installations in commercial buildings the power needed per area is high. For apartment buildings the power needed for indoor heating lies between $20-40W/m^2$ [2]

In addition, the accumulative effect in snow melting is low, and when the system is done melting the snow, little to none of the energy is stored in the pavement until the next period of snowfall. The system is subject to periodical use with rapid change in parameters, compared to room heating which is in constant operation with relatively small changes in parameters.

1.2 Why consider new control methods

Given the high power consumption and the fact that the system is subject to rapidly changing parameters compared to other temperature based HVAC systems, there is a potential of saving energy and improving the performance in optimizing the control of the system. By optimizing the control of the snow melting system, the amount of energy transferred to the pavement can be minimized while still delivering the required amount, and thus minimizing the cost of operation. If the time the system needs to be active can be reduced, this could lead to a reduction in energy lost to the environment while the system is not melting snow.

1.3 The system in question

The system that will be used for observation and testing is a snow melting system located at a retirement home at Åskollen in Drammen, covering the pavement surrounding the buildings with a cross section of about $110m^2$. The snow melting system consists of a water-to-water heat exchanger, a pump on the secondary side of the water-to-water exchanger, and a three-way valve. There are temperature sensors on the inlet and outlet of the water-to-water heat exchanger, on both the primary and secondary side. In addition, there is a set of sensors installed in and on the pavement. The temperature sensors on the primary side are related to a flow sensor, and in combination make out an energy meter. An illustration of the system can be seen in Figure 1.1. The primary side consists of everything left of the water-to-water heat exchanger, and the secondary side consists of everything to the right of it. The source of heated water available at the primary side of the water-to-water heat exchanger is a water-to-water heat pump. This heat pump draws

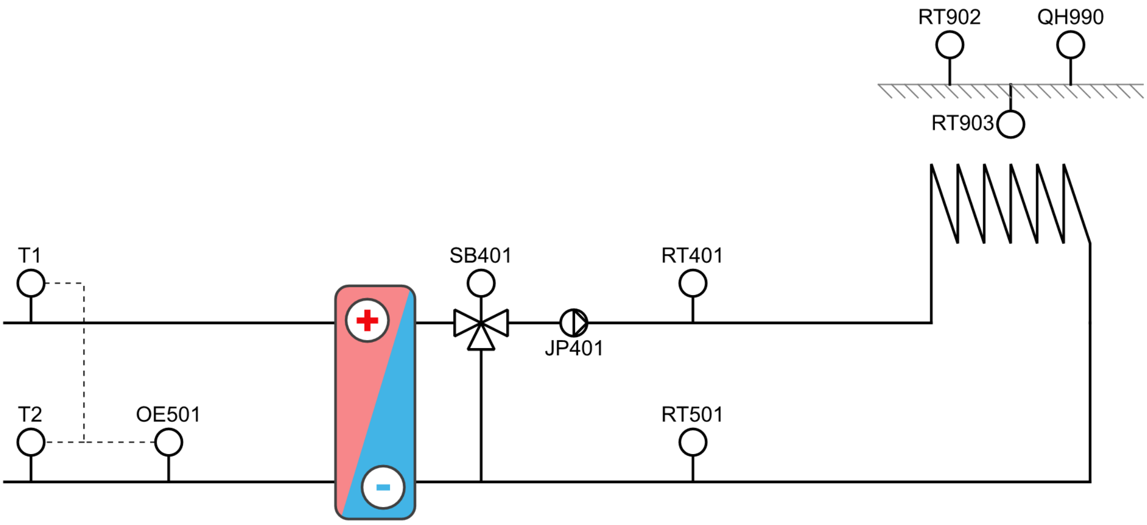


Figure 1.1: Sketch of the snow melting system with instrumentation.

energy from the ground via energy wells. This solution is an economical option compared to using a plain electrical source, with the drawback that the maximum temperature delivered is limited to $55°C$.

As a backup the water can be heated using an electrical boiler. It can be used in cases where the heat pump is faulty or to handle energy usage peaks. Energy usage peaks can be correlated to the snow melting system as it is a high energy consuming process, and is active for limited time intervals.

On the secondary side of the water-to-water heat exchanger the medium is a mixture of water and alcohol, this is to ensure that the pipes do not freeze. In ideal conditions the temperature of the heating medium do not get below $0°C$, but there is a risk, hence the need for a liquid that has a lower freezing point than water.

The available measurements are related to the earlier mentioned instrumentation. A list of the measurements available can be seen in Table 1.1. On the primary side there are two temperature measurements, supply(T1) and return(T2), and one flow meter(OE501) on the return. These sensors combined give the energy extracted in the water-to-water heat exchanger. On the secondary side there are two temperature measurements, supply(RT401) and return(RT501). The pump(JP401) on the secondary side has a number of internal sensors and gives measurements for the supply temperature, flow, pressure and electrical power consumption. The three sensors located in and on the pavement are two temperature sensors, one for the ground temperature(RT903) and one for the surface temperature(RT902). The last sensor is an humidity detector(QH990) located on the surface of the pavement. These three sensors are part of a component called *"Snøostat"*.

Table 1.1: List of measured variables from the snow melting system.

Variable	**Tag**	**Unit**	**Accuracy**
Primary side supply temperature	T1	$°C$	$\pm 0,4\%$
Primary side return temperature	T2	$°C$	$\pm 0,4\%$
Secondary side supply temperature	RT401	$°C$	$\pm 0,3°C$
Secondary side supply temperature	RT501	$°C$	$\pm 0,3°C$
Pavement surface temperature	RT902	$°C$	$\pm 0,3°C$
Pavement ground temperature	RT903	$°C$	$\pm 0,3°C$
Outdoor air temperature	RT901	$°C$	$\pm 0,3°C$
Pavement surface humidity	QH991	On/Off	
Primary side flow	OE501	m^3/h	$\pm 0,15\%$
Secondary side flow	JP401	m^3/h	$-$

1.4 Existing control philosophy

The current control system for the snow melting system is based on Proportional-Integral-Derivative(PID) controllers and setpoint determination based on weather predictions. The PID controllers are organized in a cascade manner with a master controller and a slave controller. The slave controller controls the three way valve(SB401) to give the desired temperature return temperature on the secondary side, measured by the temperature sensor RT501. The setpoint for the slave controller is given by the master controller. The master controller controls the setpoint for the slave to give the desired temperature at the surface of the pavement, measured by the temperature sensor RT902. The master controller gets its setpoint from a logical control block using weather predictions and the humidity detector to determine the setpoint. An illustration of the system can be seen in Figure 1.2

The snow melting system has five operating modes determined by the logical control. These modes are dependent on predicted rain/snowfall, predicted temperature and live humidity detection on the pavement.

The modes are as follows:

- Not active: When there is no need for snow melting, and the weather prediction do not call for rain/snowfall. In this mode the PID controllers are disabled and the valve is forced in closed position. The pump is stopped. For all other modes the pump is running and supplying constant flow.
- Preheating low: When the weather prediction indicates that there will be rain/snowfall within the next 12 hours, and the temperature is predicted to be below a certain level for the same time period. In this mode the PID controllers are activated and the pump is started. The setpoint for the pavement surface temperature,

measured by RT902, is set to $X°C$. This mode is meant for earlier preheating of the pavement.

- Preheating high: The weather prediction indicates that there will be rain/snowfall within the next 6 hours, and the temperature is predicted to be below $X°C$ for the same time period. In this mode the PID controllers are active, the pump is running and the setpoint for the pavement surface temperature, measured by RT902, is set to $X°C$. This mode is meant for late preheating of the pavement

- Melting: The outdoor temperature is below $X°C$ and the humidity detector QH990 detects snow/rain. In this mode the PID controllers are active, the pump is running and the setpoint for the pavement surface temperature, measured by RT902, is set to $X°C$.

- Dew frost protection: The temperature is predicted to fall below the dew point in the coming hours, and the temperature is so low that the dew will freeze on the ground and create a layer of ice.

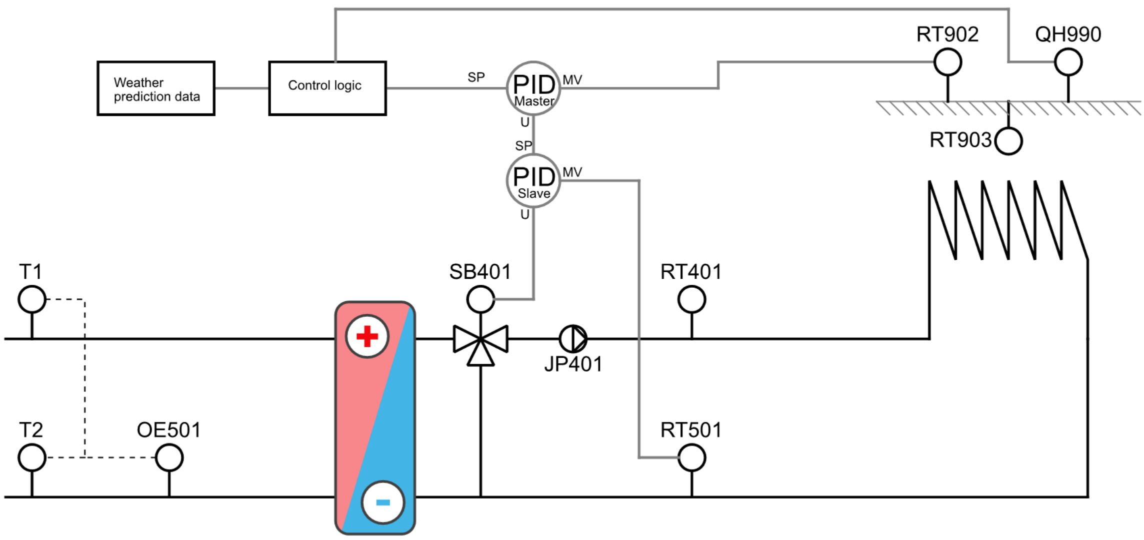


Figure 1.2: Sketch of the snow melting system with existing control system.

The pump JP401 is controlled on/off externally(by the control system) and has internal speed control, meaning that the pump delivers a constant flow of water alcohol mixture.

This control philosophy is based on experience and previous research[1], and is usually not tuned for the specific system. The preheating activation times, 12 hours for low and 6 hours for high, are an educated guess on how early the heating needs to start to preheat the pavement for it to be hot enough to melt the snow as it hits the ground, and not let it accumulate.

The weather prediction data is collected from an Application Programming Interface(API) solution delivered by The Norwegian Meteorological Institute. The prediction data is given for the following intervals: next 1-6 hours, next 6-12 hours, next 12-18 hours, next 18-24 hours, next 24-30 hours, next 30-36 hours, next 36-42 hours and next 42-48 hours. For the current control system only the next 1-6 and 6-12 hours are in use to determine the modes.

1.5 Objectives

The first objective is to obtain a mathematical model of the water-to-water heat exchanger system that is suitable for predictive control. This will be none by researching existing work on water-to-water heat exchanger models that can represent the system in question. Since the system is installed and running, the model needs to be based on variables obtainable from existing instrumentation or from assumptions.

The second objective is to verify the water-to-water exchanger system model behavior by fitting the model to data obtained from the snow melting system in question. The data can be obtained by running test sequences on the snow melting system located at Åskollen retirement home.

The third objective is to analyze the water-to-water heat exchanger snow melting system to determine what variables to optimize to improve the control. A feasibility study on implementability of the optimal control problem based on available measurements and control signals will be performed.

The fourth objective is to first use the fitted model of the water-to-water heat exchanger system to create a Model Predictive Control(MPC). This MPC must control the system in an optimal way, with respect to the variable found to optimize. Then the performance of the controller relative to the existing implemented control philosophy can be determined.

The fifth and final objective is to analyze the economical benefit of implementing a model based controller. It will be determined if the implementation cost of MPC can be justified by the saved energy usage. The expected lifetime of the system is to be taken into consideration. The adjustments needed on the MPC in the case of a change in control equipment is to be considered, and the cost of making these adjustment is to be taken into account when analysing the feasibility.

1.6 Project goals

- Give an insight into different uses and setups of water-to-water heat exchangers in HVAC installations.
- Perform a study of which variables can be optimized in a snow melting system.
- Formulate a mathematical model of a heat exchanger with the goal of simulating its behaviour. The model can be formulated using non-linear differential equations, ordinary differential equations or First Order Plus Dead Time(FOPDT). The possibility to use system identification to obtain a mathematical model is to be investigated.
- The model(s) is to be fitted to data gathered from the chosen system. The data shall come from test runs where a step response is performed.
- Develop a Model Predictive controller (MPC) to control the heat exchanger. What variables to optimize shall reflect the finds during the literature study.
- Test the MPC on simulations and compare the results with simulations with traditional control methods (PI/PID). Use weather forecasting to predict changes in the system and implement the changes in the optimal control problem.
- If time allows it, and it is feasible with the software used to control the system, the MPC is to be tested on the physical system
- Perform an economical evaluation of the possible savings.

1.7 Methods

During the course of this project the methods used are as follows. First research on existing models for describing processes with the behavior similar to the one of a water-to-water exchanger will be conducted. Then research on the instrumentation present in such a system and the relevant measurement methods is conducted. Further a feasibility study on the possibility of model predictive control of snow melting systems is conducted. Then data is gathered from historical logging data and by execution of step/response testing. Then formulation of optimization problems are conducted. Software is developed using Simulink and the same software is used for testing of the model. Plots of data is used to compare the results from the tests and determine its performance.

1.8 Report structure

Chapter 1 gives an introduction to the background and scope of the project.

Chapter 2 describes the relevant theory for the project.

Chapter 3 describes the existing control system.

Chapter 4 describes the collection of measurement data.

Chapter 6 discusses the feasibility of MPC for snow melting systems.

Chapter 7 describes the formulation of a model for the MPC.

Chapter 8 describes the formulation of the optimization problem.

Chapter 9 describes the implementation of the MPC on Simulink.

Chapter 10 describes the process of fitting the model to the gathered measurement data.

Chapter 11 describes the testing of the MPC.

Chapter 12 presents an economical evaluation of implementation of MPC for the snow melting system.

Chapter 13 presents the a summary of the main results, detailed descriptions of results are given in Chapters 11 and 12. Where Chapter 11 discusses and presents the performance of the MPC, and Chapter 12 discusses the economical aspect of implementation of MPC.

Chapter 14 discusses the project.

Chapter 15 gives the conclusion.

2 Theory

In this chapter some theory of the relevant concepts and equipment is given. Water-to-water exchangers are briefly discussed, some concepts for mathematical modeling is presented, theory on model fitting given, system identification is discussed, common control methods are addressed, and finally some theory on MPC is given.

2.1 Water-to-water heat exchanger

A water-to-water heat exchanger is a device for transferring energy from a liquid to another, without the liquids mixing.

There are several types of water-to-water heat exchangers, but the focus here will be on plate-design exchangers. In these exchangers the liquids are passed through chambers separated by thin metal plates. The energy is passed from one liquid through the metal plate and absorbed by the other liquid.

2.2 Mathematical models

A dynamic system can be described by several types of mathematical models. The focus here will be on the two approaches, *First principles* and *First Order Plus Dead Time*.

2.2.1 First principles model

First principles models are based on the fundamental physics that can be used to describe a system or process. First principles are concepts describing the behaviors of a system, such as energy balance, mass balance, and other laws of physics to derive mathematical equations. [3]

One such approach for a water-to-water heat exchanger model has been presented in the thesis *'Temperature control and power monitoring of liquid cooling systems for mechanical manufacturing'* by Johansson, Lybekk and Moe. [4]. Here energy balance is used as a base to develop a mathematical model of the water-to-water heat exchanger, and the model can be presented as the following energy balance:

$$\frac{dE}{dt} = \sum Q_{\text{in}} - \sum Q_{\text{out}} \tag{2.1}$$

Where the energy is represented by E, Q_{in} represents the flow of energy into the system and Q_{out} represents the energy extracted from the system. Dividing the water-to-water heat exchanger chambers into smaller sections and defining separate energy balances describing the exchange of energy between them. A set of ten differential equations, five for the primary side and five for the secondary side of the exchanger can be derived and gives the following equations:

$$\dot{T}_{p_1} = \frac{1}{c_p\rho_p(V_p/5)}(c_p w_p(t)(T_{p_{\text{in}}} - T_{p_1}) + U(A/5)(T_{s_{\text{out}}} - T_{p_1})) \tag{2.2}$$

$$\dot{T}_{s_{\text{out}}} = \frac{1}{c_s\rho_s(V_s/5)}(c_s w_s(t)(T_{s_2} - T_{s_{\text{out}}}) - U(A/5)(T_{s_{\text{out}}} - T_{p_1})) \tag{2.3}$$

$$\dot{T}_{p_2} = \frac{1}{c_p\rho_p(V_p/5)}(c_p w_p(t)(T_{p_1} - T_{p_2}) + U(A/5)(T_{s_2} - T_{p_2})) \tag{2.4}$$

$$\dot{T}_{s_2} = \frac{1}{c_s\rho_s(V_s/5)}(c_s w_s(t)(T_{s_3} - T_{s_2}) - U(A/5)(T_{s_2} - T_{p_2})) \tag{2.5}$$

$$\dot{T}_{p_3} = \frac{1}{c_p\rho_p(V_p/5)}(c_p w_p(t)(T_{p_2} - T_{p_3}) + U(A/5)(T_{s_3} - T_{p_3})) \tag{2.6}$$

$$\dot{T}_{s_3} = \frac{1}{c_s\rho_s(V_s/5)}(c_s w_s(t)(T_{s_4} - T_{s_3}) - U(A/5)(T_{s_3} - T_{p_3})) \tag{2.7}$$

$$\dot{T}_{p_4} = \frac{1}{c_p\rho_p(V_p/5)}(c_p w_p(t)(T_{p_3} - T_{p_4}) + U(A/5)(T_{s_4} - T_{p_4})) \tag{2.8}$$

$$\dot{T}_{s_4} = \frac{1}{c_s\rho_s(V_s/5)}(c_s w_s(t)(T_{s_5} - T_{s_4}) - U(A/5)(T_{s_4} - T_{p_4})) \tag{2.9}$$

$$\dot{T}_{p_{\text{out}}} = \frac{1}{c_p\rho_p(V_p/5)}(c_p w_p(t)(T_{p_4} - T_{p_{\text{out}}}) + U(A/5)(T_{s_5} - T_{p_{\text{out}}})) \tag{2.10}$$

$$\dot{T}_{s_5} = \frac{1}{c_s\rho_s(V_s/5)}(c_s w_s(t)(T_{s_{\text{in}}} - T_{s_5}) - U(A/5)(T_{s_5} - T_{p_{\text{out}}})) \tag{2.11}$$

These are to be seen in relation with the illustration in Figure 2.1

In eq. 2.2 to eq. 2.11 c_p and c_s represents the heat capacities of the liquids, w_p and w_s represents the flows of mass, U is the heat transfer coefficient of the specific heat exchanger, A is the area of contact between the chambers, ρ_p and ρ_s represents the density of the fluids.

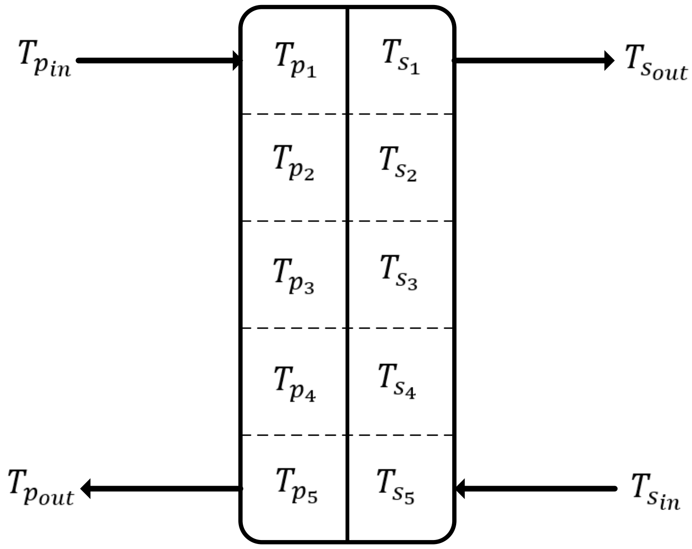


Figure 2.1: Figure illustrating the simplified layout and interaction between the energy balances forming the water-to-water heat exchanger model.[4]

2.2.2 First Order Plus Dead Time

A First Order Plus Dead Time(FOPDT) model can describe many dynamic processes. Temperature processes are commonly of such a character that they can be described by a FOPDT model, which is defined as follows[5]:

$$\tau_p \frac{dy(t)}{dt} = -y(t) + K_p u(t - \theta_p) \tag{2.12}$$

Where K_p is the process gain, τ_p is the process time constant and θ_p is the process dead time. [5]

The process gain is based on the effect a change in input $u(t)$, has on the output $y(t)$. The process gain is given by:

$$K_p = \frac{\Delta y}{\Delta u} \tag{2.13}$$

The dead time, also known as the transport delay, is the time it takes from when a change is made in the input to a response can be seen in the output.

The time constant describes how fast the system is when not counting dead time, and is measured by the time it takes from a change in output is detected to when the output reaches 63.2% of the difference between the starting and final steady state.

The transfer function for a FOPDT model can be given by [2]:

$$G(s) = \frac{K_p e^{-t_0}}{dt} \tag{2.14}$$

The FOPDT model can be derived from First principles, but further information on this is left to the reader.

2.2.3 Model fitting

For both First principles models and First Order Plus Dead Time models there is a need for model fitting. For First principles models the basic behavior of the system is described by the physics and parameters used to develop the model, but there is still a need to fit the model to remove error in the parameters, and in disturbances not implemented in the model. First Order Plus Dead Time models are as described based on three parameters, process gain, time constant and dead time. These parameters are not in the same way as the First principles model partially given by the parameters in the model, and need to be fitted entirely. There are several approaches to model fitting both analytically and mathematically. A common method is regression, which can be used for both linear and non-linear systems. Another common method is to manually adjust models based on plots[6].

2.3 System identification

System identification is a method for obtaining models of a system, commonly physical or economical, based on measurement data obtained from the system. System identification uses statistical methods to create a mathematical model. There is a variation of ways to approach system identification, but the most common method is called *Black Box* and is when what happens inside the system is unknown and the mathematical models is constructed only based on the behavior of the inputs and outputs of the system. Another approach is called *Gray Box*, and unlike the Black Box methods, what happens inside the system is known to some degree. The FOPDT model could be used for such an approach, where the relationship is somewhat defined but the parameters process gain, time constant and dead time are free variables to be determined by system identification[7].

2.4 Common control methods (PID and cascade)

The currently implemented control philosophy at the test location is based on PID controllers. Using PID is common for these types of control systems and are considered a reliable solution. A PID controller is an error based controller that calculates the control signal using three components, the proportional part, the integrating part, and the derivative part.

As described in Chapter 1.4 the PID controllers are organized in a cascade manner. This is a way of controlling a system with two different control loops with two sets of PID-parameters. The cascade control consists of one outer loop that is commonly slower and controlling the main process parameter, and one inner loop that is commonly faster and controlling an internal process variable that has an effect on the main process parameter.

2.5 Model predictive control

Model Predictive Control(MPC) is a concept for controlling dynamic processes. It uses knowledge about the behavior of the system, in the form of a mathematical model, to determine the appropriate control outputs to get the desired future behavior of the system. The MPC simulates the behavior of the system for a finite future time horizon, based on the current states and future known disturbances, and finds the appropriate control outputs(inputs to the system) for obtained the desired system states. MPC can be used to control both physical system and abstract systems such as economical systems. There are many approaches to MPC, and the degree of complexity can vary. One concept to address in relation to using MPC for control of a physical system is the *Moving Horizon*[8], where the control problem is re-calculated at every time step in a control loop, and the prediction horizon is moved forward with time. As mentioned, the Moving Horizon concept is fit for direct control of dynamic system, but cases where it is not applicable might be when using MPC for planning long term systems such as investments[9].

3 Existing Hardware

The installed control system is delivered by Schneider Electric, the controllers are of type SmartX controllers and is a type of Automation Server(AS). Automation servers are a type of Programmable Logical Controller(PLC), but for use in analog control systems. Some of the differences between PLC and AS is that the AS is designed to handle processing of analog data and has a longer cycle time. In addition, an AS has a greater storage capacity for data and user interfaces. The instrumentation on the snow melting system consists of mostly well known sensor types such as thermistors for temperature measurements and ultrasound for flow measurements. The sensor for detecting snow, called a ”Snøostat” is an uncommon component that is only used in snow melting systems.

3.1 Controller

The existing control philosophy at the test location, described in Chapter 1.4, is implemented on an Automation Server delivered by Schneider Electric. This system is, in addition to running the software for implementing the control, hosting the sensors and actuators of the system. It can implement numerous types of I/O as well as the most common bus systems such as Modbus and BACnet. The temperature sensors, as well as the Snøostat and the valve, is implemented using I/O, while the pump and the flow meter is implemented using Modbus.

The AS used for control of the snow melting system is only one of many servers in the system that make up the control system for HVAC for the building in question. Along side the other servers in the building, the AS for the snow melting system is hosted by a Supervisory system called the Enterprise Server(ES). The ES is where the user interface is handled, as well as control across the Automation Servers, alarm handling, long time logging and administration . The communication between the Enterprise Server and the Automation Servers is on a proprietary protocol. The AS has the ability to integrate an interface for obtaining the weather predictions, but for implementation reasons this is done in the ES. A simplified topology of the system can be seen in Figure 3.1.

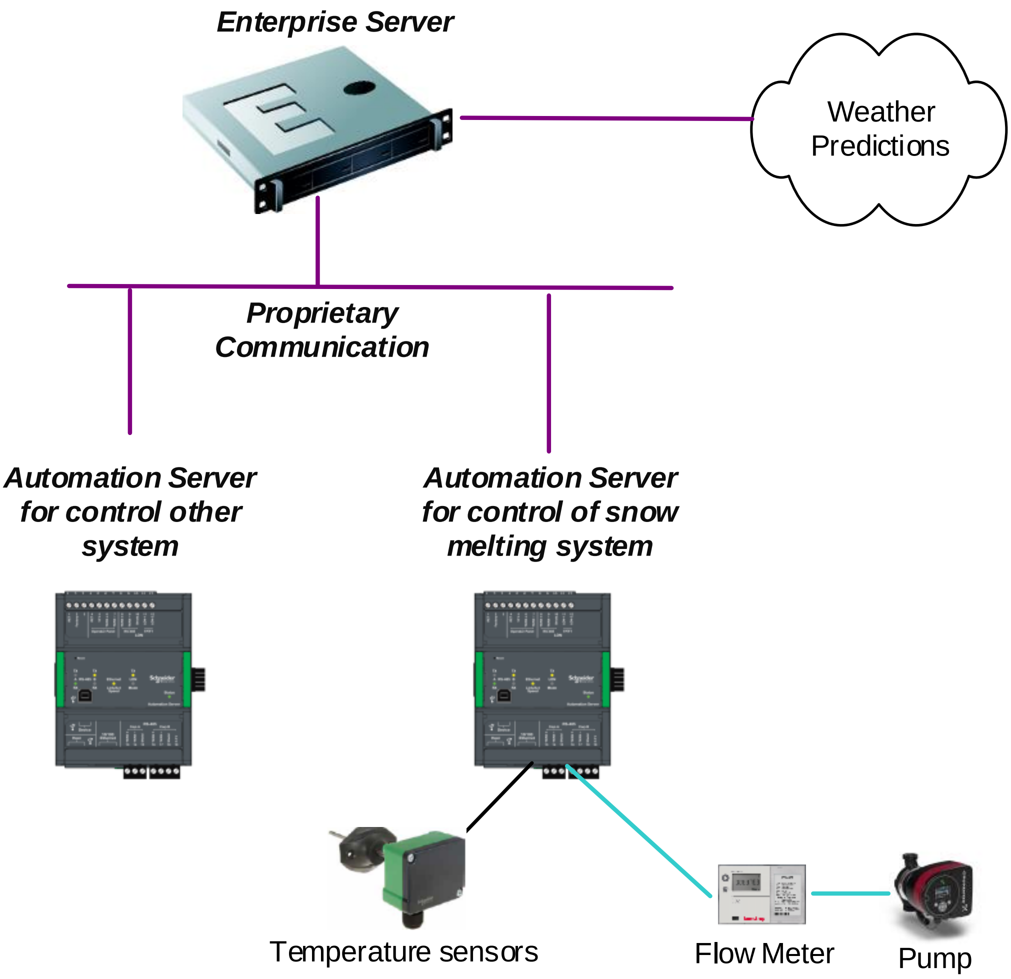


Figure 3.1: Simplified topology of the server structure.

3.2 Field equipment

The snow melting system is equipped with components that are common in the HVAC field. The temperature sensors are of NTC20 type thermistor, giving $10k\Omega$ at $25°C$. NTC stands for Negative Temperature Coefficient and indicates that the resistance in the sensor element is inverse proportional to the temperature.[10][11]

The temperature sensors used in this system are *Schneider Electric STP100*, an example of the temperature sensor can be seen in Figure 3.2.

The pump installed on the secondary side of the snow melting system is a *Grundfos Magna 3*. This is a combined pump and frequency converter. The pump is set to deliver

Figure 3.2: Illustration photo of the temperature sensors used in the snow melting system.[11]

a constant flow, and is equipped with internal sensors for monitoring the flow. From the pump integration various operational data such as speed and energy consumption can be read. In addition, data from the internal sensors are available.[12]

The energy meter used on the primary side of the snow melting system is a *Kampstrup Multical 602*. This meter consist of one flow sensors and two temperature sensors. The temperature sensors are *PT100* type. PT100 temperature sensors are Platinum thermistor that have 100Ω resistance at $0°$, and is the most common temperature sensor element used in industry. The Flow sensor is called ULTRAFLOW and is an ultrasound sensor. Ultra sound for flow metering is done by sending a sound pulse from one node to another, and measuring the time it took for the sound pulse to travel between the nodes. As sound travels at different speeds in different mediums, the medium needs to be known. For the snow melting system primary side the medium is water, and sound travels at $1481 m/s$ in water. If the medium in the meter is moving the time it takes from the sound pulse is emitted from one node until it is detected by the other node will change. If the medium flows from the emitting node towards the detecting node the time the sound pulse uses to travel decreases, If the medium flows in the opposite direction the time increases. From this deviation in time, together with physical dimensions of the sensors and other physical constants the flow can be calculated.[13][14][15]

Snøostat

The snow sensor, called "Snøostat" is a product delivered by *Jan Grosh AS* designed for the sole purpose of use in snow melting systems. The unit consists of two temperature sensors, one mounted at the surface facing upwards at the same level as the pavement surface, and one mounted downwards in the ground for measuring ground temperature. The temperature elements are of the same type as for the STO100. There is also a humidity sensor that uses change in resistance to detect water/snow on the surface. A heating element is installed on the surface of the sensor to evaporate the snow and water on the sensor. The heating element is needed as the snow melting system itself doesn't cover the area covered by the Snøostat. The layout of the Snøostat can be seen in Figure 3.3 [1]

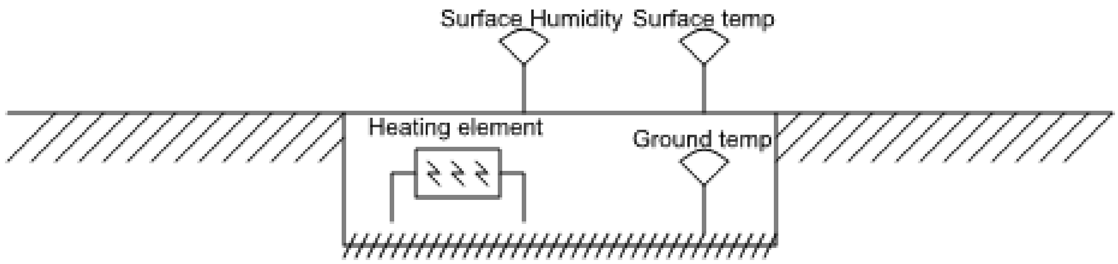


Figure 3.3: Sketch showing the layout of the Snøostat sensor.

4 Data acquisition

In this chapter the logging of measurement data in the control system is discussed, and the process of gathering weather data from Norwegian Meteorological Institute(MET) using API is addressed.

4.1 Logging measurement data

As the system is controlled and monitored in its entirety by the AS, information on the control signals and measurements are available through the same interface. The snow melting system and accomplishing infrastructure are operational at the beginning of the project, so some historical data from operation with the existing control system is available. The data is stored in trend logs and are available to display directly in the AS interface, or it can be exported on .xls/.csv format for use in other software. The data is logged either on Change Of Value(COV) or in set time intervals.

The quality of the data varies depending on the sensors. An overview of the measurements and corresponding accuracy can be seen in Table 1.1. In addition to the error in the sensor there is some error in the equipment recording the measurement. The AS is a digital recorder and the increments in measurement change recorded is finite, thus creating some error in the measurement.

As mentioned the logging can be triggered to record a log sample based on time interval or COV. The COV limit defines how much the measurements need to deviate relative to the previous sample for a new sample to be stored. For the variable types relevant for the snow melting system; temperature and flow, the COV is usually set to $0.5°C/0.5m^3/h$. This will potentially add an error of $1.0°C/1.0m^3/h$ to the log points, as the measurement is allowed to deviate $\pm0.5°C/\pm0.5m^3/h$ before a new sample is recorded. An illustration of this case can be seen in Figure 4.1.

4.2 Weather data

The weather predictions are gathered from the Norwegian Meteorological Institute using an API solution. The ES handles the interface between the control system and the weather

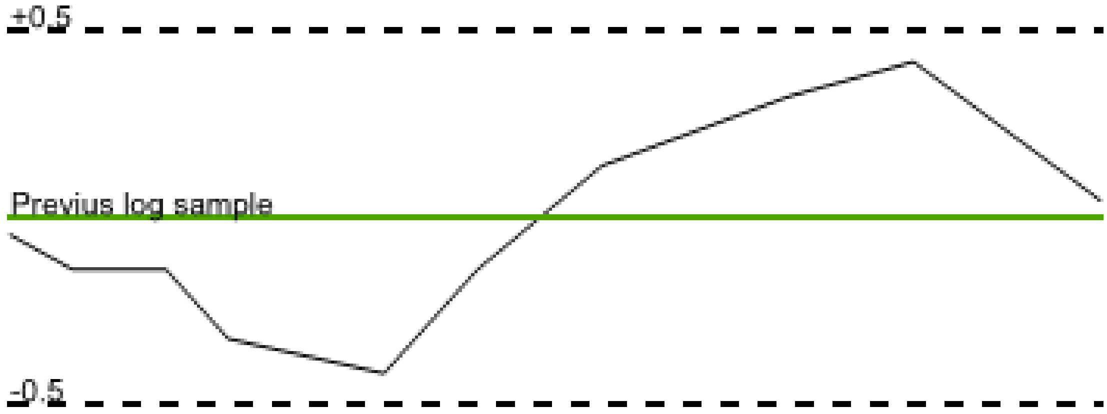


Figure 4.1: Illustration of COV triggered log, and potential unlogged measurement behavior.

prediction data source. As earlier described, the predictions are given in intervals of 6 hours into the future. The data is updated every 15 minutes so that eventual changes in predictions are intercepted.

The reliability of the weather predictions is dependent on the models used and the measurement data available. Forecasted weather data is said to have an accuracy of 80% accuracy for a 7-day forecast, and up to 90% accuracy for 5-day forecasts. In 'Bruk av værprognoser for optimal styring av snøsmelteanlegg' by Jonsson[1] it is argued that introducing predicted weather data in control systems for snow melting has an positive effect on the economical aspect.[16]

5 Measurement data

Ideally, obtaining data for fitting the model of the water-to-water heat exchanger snow melting system to the behavior of the real system would be done by trial runs in ideal conditions. There is data available for normal operation where the system is being controlled by the established control system. This can be used to identify the dynamics of the system, but the data is colored by the feedback in the control system. The parameters in the FOPDT model that need to be identified is the gain, dead time and time constant. The trial run desired to conduct is a step/response, where the system is manually controlled to a steady state before the controllable system input is given a step change and the system is monitored until it reaches the new steady state. How big a change to make on the system input, in this case the valve control system, needs to be evaluated for the specific system for the data to show changes of observable amplitude. Typically two trials are performed, one where the valve control signal is increased by 10% and one where the control signal is decreased by 10%.

Unfortunately the opportunity to perform trial runs under ideal conditions was not present during the time of research. trial runs was dependent on weather, by the time the appropriate preparation for doing the trial runs were complete, proper research and definitions, the temperature and snow conditions were not present.

Fortunately there is available historical data for normal operation for the snow melting system at Åskollen, as well as for other locations. This historical data is not as ”clean” as data from a trial run would be, in regards to obtaining the dead time, time delay and gain, but the data will be sufficient to estimate the parameters for the purpose of further research. Historical data from normal operation does not have the clear action and reaction behavior as a step/response would have, but as the implemented control philosophy has modes it switches between based on weather conditions and humidity measurements the control has behavior resembling a step behavior in the desired setpoints. As an example when transitioning between the mode *Not active* to *Preheating Low*. Systems with only one operational mode that is always active and that uses PID control will not have this step response like behavior that can be found in the historical data, unless the setpoint is suddenly changed with a magnitude that causes the control signal to change rapidly from a steady state to another. If the PID controller for such systems is only adjusting for errors in the system and not changes in setpoint, there would not be possible to determine the dead time, time delay and gain for the system.

The historical data is available in the ES, but for a limited time period determined by available storage capacity. As described in Chapter 4 the measurements are logged at a fixed time interval or based on COV, and since the log size is of a finite number of samples the loges will have varying length of time. The challenge is then to find a period in the logs containing a transition between the mode Not active and Preheating Low and that has all the necessary data available.

One flaw to the historical data is that it only contains information on measurements and control signals. Weather prediction data and control modes are not logged. Determining when the control system transitions between Not active and Preheating Low therefore needs to be assumed based on system behavior. In Figure 5.1 the control signal to SB401 is shown. Assuming that the periods where the control signal is zero represents the mode Not Active and that the peaks in control signal following periods of zero values represents the start of a preheating period, a step/response equivalent type of behavior is located.

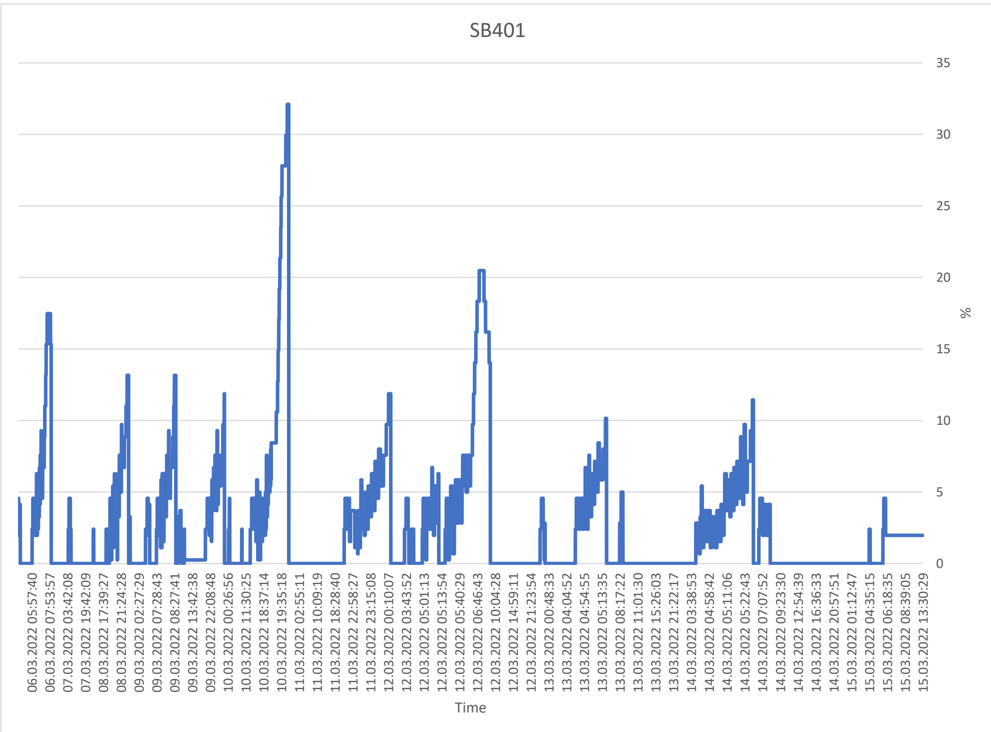


Figure 5.1: Historical data of the control signal to SB401.

Taking a closer look at one of the periods as described, from about timestamp 12.03.2022 05:10:00 to 12.03.2022 23:59:00(shown in Figure 5.2) the behavior of the control signal to SB401 can be seen in greater detail. This signal is subject to control by a PID controller in the existing control system and is therefore somewhat oscillating. Ideally, to analyze the behavior of the control signal would be a step but as the data is from normal operation it is not available. However, when looking at the control signal for SB401 and the pavement surface temperature measured by RT902 together the relationship can be analysed approximately. The two variables can be seen plotted together in Figure 5.3.

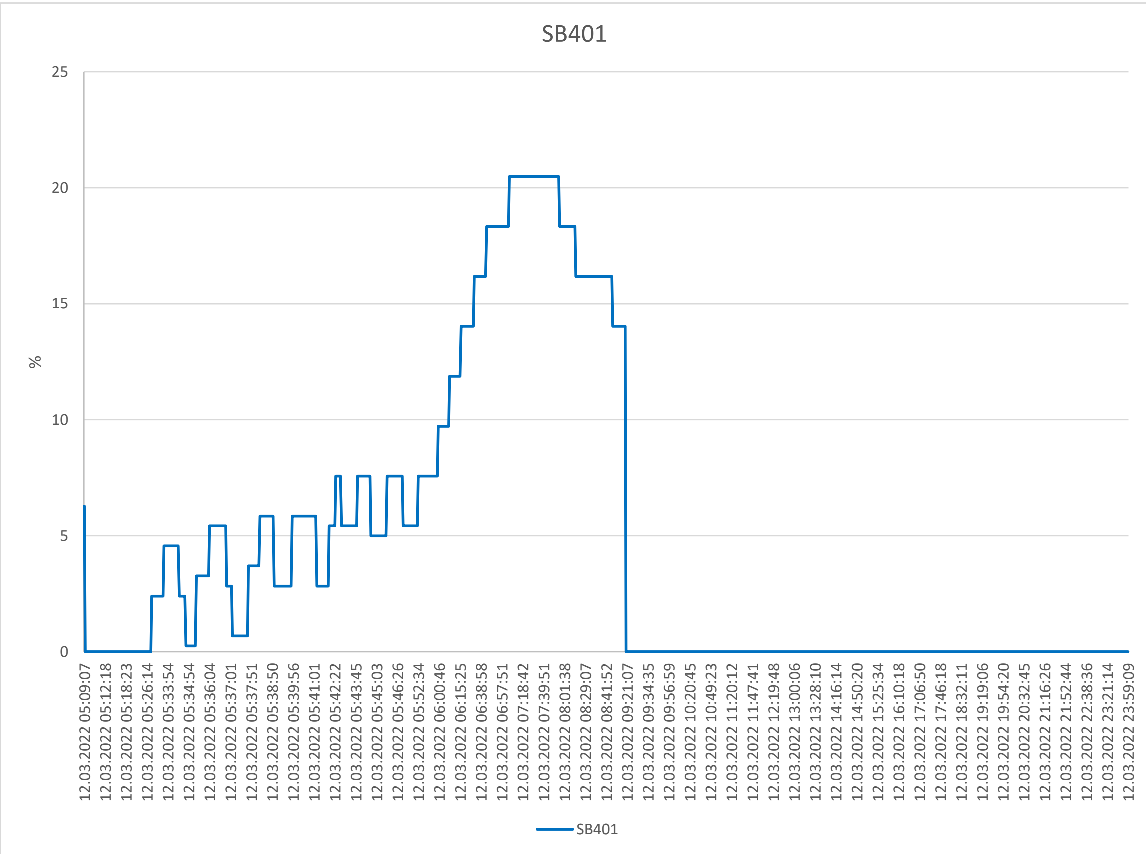


Figure 5.2: Historical data of the control signal to SB401 for a period representing change in operating mode.

Historical measurement data is not sufficient for describing the behavior of the snow melting system primary side. Therefore a step/response test was conducted. The test was performed under conditions where snowfall was not plausible considering air temperature, but the results from the test is sufficient to describe the behavior of the system not obtainable from historical data.

The data discussed here will be addressed again in greater detail when fitting the model to the behavior of the snow melting system.

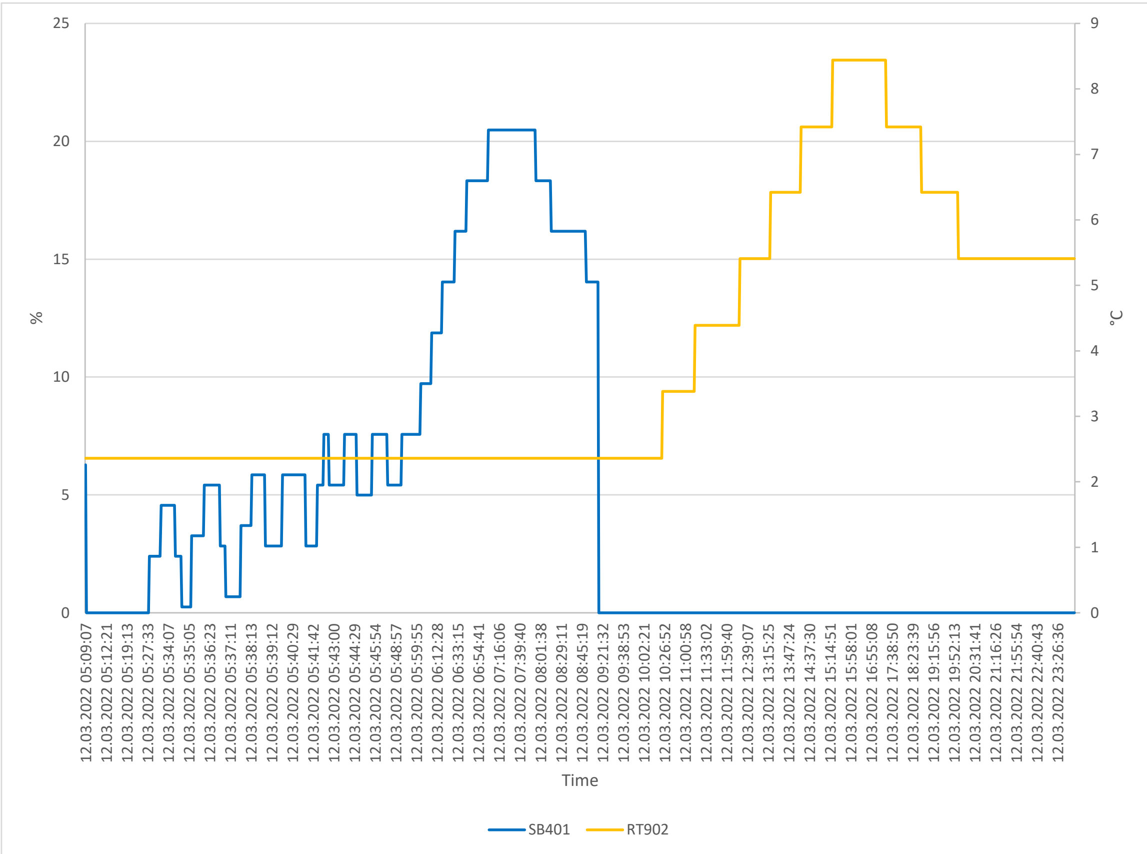


Figure 5.3: Historical data of the control signal to SB401 and temperatures T2 and RT902 for a period representing change in operating mode.

6 MPC Feasibility

To determine the needed complexity of the MPC controller, and then again the complexity of the model, an analysis of inputs, states and outputs of the system is required. In Chapter 1 an overview of the existing instrumentation of the snow melting system is given, but it is not certain that the currently available measurements are sufficient for MPC.

Firstly it is important to determine what variables are desired to optimize. In the case of snow melting systems for commercial buildings the main factors are cost and comfort. Comfort in this case is to keep the pavement free for snow and ice, and this can be maintained by heating the pavement to keep a certain temperature when it is about to snow or the conditions for water to condensate from the air is present. The comfort can say to be maximized by minimizing the time the pavement surface is covered by snow or ice. In terms of available measurements this is when there is detected humidity on the pavement surface, and the surface temperature is below freezing. To clarify, the temperature can be below freezing as long as the pavement is dry. For the MPC to be developed, the focus is on snowfall and not condensation, therefore the measure of comfort will be to ensure that the pavement is heated at the time snowfall is predicted.

The cost aspect is mainly the cost of the energy used for heating the pavement. There is some cost in running the pump and actuators of the system, but these are trivial and can be neglected for this analysis. The cost can then be minimized by minimizing the energy consumption. The energy consumption of the snow melting system is in the form of thermal energy supplied by the heatpump, as described in Chapter 1. This thermal energy can be calculated using the measurements of primary side supply temperature, return temperature and flow. As the supply flow and temperature is delivered from another part of the system, that is designed to delivered the required energy, these parameters can if desired be assumed to be constant. This means that the energy consumption only varies with the return temperature on the primary side.

To emphasize, the variables in which to include in the control system can be the pavement surface temperature, the pavement ground temperature, the pavement humidity detection for comfort monitoring, and the primary side return temperature to monitor the energy. The choice of variables are dependent on the model chosen.

For this project, the MPC is to be considered for replace the modes of the existing control system that is called ”Not active” ”Preheating low”, ”Preheating high” and ”Melting”.

The mode ”Dew frost protection” are left to the original control system, or further investigation at a later time. If necessary the mode ”Melting” can be considered left to the original control system as well. The reason for this is that the ”Melting” mode is determined by detection of humidity on the ”Snøostat” and temperature directly, regardless of the weather forecast. This means that there is no future data available to use in the MPC to optimize when to start and stop the ”Melting” mode. A way to add some information to the system that might help predict when to activate the ”Melting” mode could be to add a sensor that detects snowfall and rainfall separate from the pavement. This sensor would need to detect snow/rainfall without the delay that the sensor in the Snøostat is subject to. In this way it can be detected when the snow/rainfall stops, and how long the ”Melting” mode needs to be active can be predicted. The mode ”Dew frost protection” is also left to be implemented in MPC for another time. As this mode is only determined by predictions and no local sensors, the existing control depends only on already predicted values.

7 Formulate model for MPC

In this Chapter the formulation of a model of the snow melting system for use in MPC will be discussed. Four different alternatives will be presented, describing the system in different manners. The presented models vary in which and how many variables they represent, and how they can be used to develop a MPC. The appropriate alternative is chosen for further use in development of the MPC.

7.1 Alternative one

The system can be described by a model consisting of FOPDT equations for each of the monitored variables. Defining the FOPDT equations with the control signal as the input and the variables as outputs. This would give the equations seen in eq 7.3 to eq 7.10. For a MPC controller substituting the entire existing control system described in Chapter 1.4, this model formulation would be a feasible alternative as the inputs and outputs of the model reflects the variables controlled by the PID controllers in the existing control system.

The model consisting of FOPDT equations is derived as follows. The general form of the FOPDT

$$\tau_p \frac{dy(t)}{dt} = -y(t) + K_p u(t-\theta_p) \tag{7.1}$$

Isolating the derivative and writing a FOPDT model for each variable

$$\frac{dy(t)}{dt} = \frac{-y(t) + K_p u(t-\theta)}{\tau_p} \tag{7.2}$$

$$\frac{dy_{T1}(t)}{dt} = \frac{-y_{T1}(t) + K_{T1} u_{SB401}(t-\theta_{T1})}{\tau_{T1}} \tag{7.3}$$

$$\frac{dy_{T2}(t)}{dt} = \frac{-y_{T2}(t) + K_{T2} u_{SB401}(t-\theta_{T2})}{\tau_{T2}} \tag{7.4}$$

$$\frac{dy_{RT401}(t)}{dt} = \frac{-y_{RT401}(t) + K_{RT401} u_{SB401}(t-\theta_{RT401})}{\tau_{RT401}} \tag{7.5}$$

Table 7.1: List of parameters for the FOPDT model without simplifications

	Gain	Deadtime	Timeconstant
T1	K_{T1}	θ_{T1}	τ_{T1}
T2	K_{T2}	θ_{T2}	τ_{T2}
RT401	K_{RT401}	θ_{RT401}	τ_{RT401}
RT501	K_{RT501}	θ_{RT501}	τ_{RT501}
RT901	K_{RT901}	θ_{RT901}	τ_{RT901}
RT902	K_{RT902}	θ_{RT902}	τ_{RT902}
RT903	K_{RT903}	θ_{RT903}	τ_{RT903}
RT904	K_{RT904}	θ_{RT904}	τ_{RT904}
OE501	K_{OE501}	θ_{OE501}	τ_{OE501}
JP401	K_{JP401}	θ_{JP401}	τ_{JP401}

$$\frac{dy_{RT501}(t)}{dt} = \frac{-y_{RT501}(t) + K_{RT501}u_{SB401}(t-\theta_{RT501})}{\tau_{RT501}} \tag{7.6}$$

$$\frac{dy_{RT902}(t)}{dt} = \frac{-y_{RT902}(t) + K_{RT902}u_{SB401}(t-\theta_{RT902})}{\tau_{RT902}} \tag{7.7}$$

$$\frac{dy_{RT903}(t)}{dt} = \frac{-y_{RT903}(t) + K_{RT903}u_{SB401}(t-\theta_{RT903})}{\tau_{RT903}} \tag{7.8}$$

$$\frac{dy_{OE501}(t)}{dt} = \frac{-y_{OE501}(t) + K_{OE501}u_{SB401}(t-\theta_{OE501})}{\tau_{OE501}} \tag{7.9}$$

$$\frac{dy_{JP401}(t)}{dt} = \frac{-y_{JP401}(t) + K_{JP401}u_{SB401}(t-\theta_{JP401})}{\tau_{JP401}} \tag{7.10}$$

The found model describes all the monitored variables in the snow melting system, making the model complex. The model consist of variables of the same dynamic behavior, but with large difference in how fast the processes are. For example the temperature on the return pipe on the secondary side, RT501, will change faster and with a greater amplitude than the temperature in the pavement, RT903.

Table 7.2: List of parameters for the FOPDT model with some simplifications

	Gain	Dead time	Time constant
T2	K_{T2}	θ_{T2}	τ_{T2}
RT401	K_{RT401}	θ_{RT401}	τ_{RT401}
RT501	K_{RT501}	θ_{RT501}	τ_{RT501}
RT902	K_{RT902}	θ_{RT902}	τ_{RT902}

7.2 Alternative two

To simplify the model and MPC the variables either not relevant for the control or redundant can be excluded. The flow on the primary side, given by JP401 can be excluded as it is set to be constant. There is small variations in the flow due to noise and control error but for the purpose of MPC it can be assumed to be constant and the error to be ignored. The flow on the primary side given by OE501 and the supply temperature on the primary side can be excluded from the model and assumed to be constant, as the source of heated water is designed to deliver the requested temperature and flow. It is important to note that there will be some disturbances in the temperature when the snow melting system is turned on or off, but not so much that it affects the snow melting system performance. As the surface temperature is the important factor in melting of snow and ice, the ground temperature can be excluded from the model.

These simplification lead to the equations describing the model eq 7.11 to eq 7.14 and the variables in the Table 7.2. A sketch of the described MPC structure can be seen in Figure 7.1.

$$\frac{dy_{T2}(t)}{dt} = \frac{-y_{T2}(t) + K_{T2}u_{SB401}(t-\theta_{T2})}{\tau_{T2}} \tag{7.11}$$

$$\frac{dy_{RT401}(t)}{dt} = \frac{-y_{RT401}(t) + K_{RT401}u_{SB401}(t-\theta_{RT401})}{\tau_{RT401}} \tag{7.12}$$

$$\frac{dy_{RT501}(t)}{dt} = \frac{-y_{RT501}(t) + K_{RT501}u_{SB401}(t-\theta_{RT501})}{\tau_{RT501}} \tag{7.13}$$

$$\frac{dy_{RT902}(t)}{dt} = \frac{-y_{RT902}(t) + K_{RT902}u_{SB401}(t-\theta_{RT902})}{\tau_{RT902}} \tag{7.14}$$

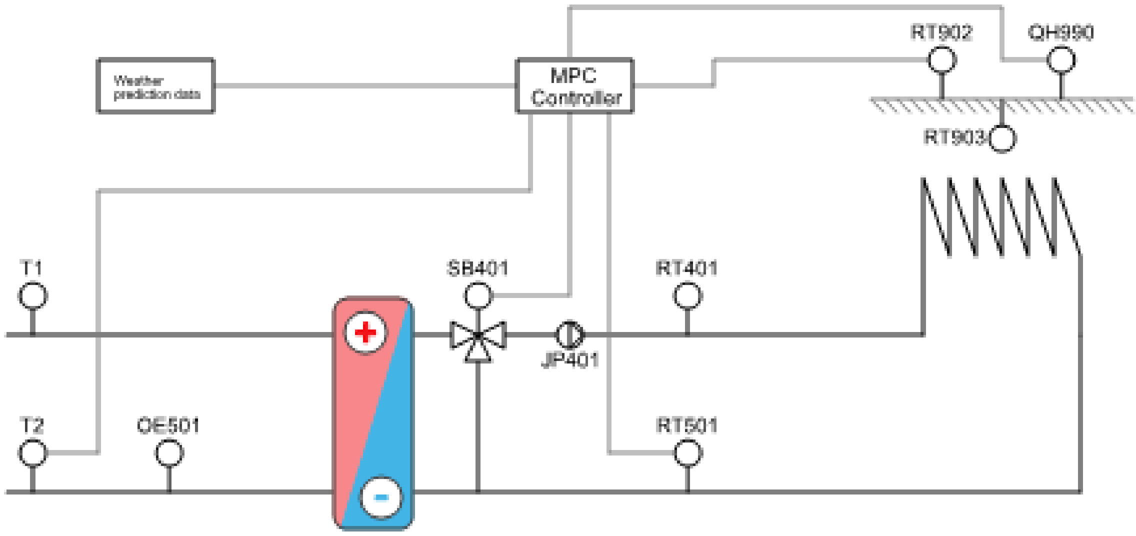


Figure 7.1: Sketch of the snow melting system full MPC control

7.3 Alternative three

An additional way of simplifying the model and the following MPC is to limit what the MPC is to control. Instead of replacing the entire existing control system, consisting of logic for choosing the master PID reference value, the master PID and the slave PID, the MPC can replace only the logic and master PID controller. The disturbances from weather changes that affect the system and is to be monitored for predicted control mainly affect the pavement surface temperature. This temperature is subject to control by the master controller and the reference for the same temperature is defined by the logic. The return temperature on the primary side, controlled by the slave PID, is as described a relatively fast process. PID control performs well on controlling this temperature. By defining the MPC controller to replace the logic and master PID, and keeping the slave PID, the complexity of the model can be reduced. The slave PID controller that control the temperature then needs to be defined as a Second Order Plus Dead Time(SOPDT) process, as the controller adds another dimension to the process, thus re-introducing some complexity. A sketch of the described MPC structure can be seen in Figure 7.2.

7.4 Alternative four

An alternative formulation of the MPC is to look into the modes of the existing control philosophy and evaluate what modes to replace with MPC. As described in Chapter

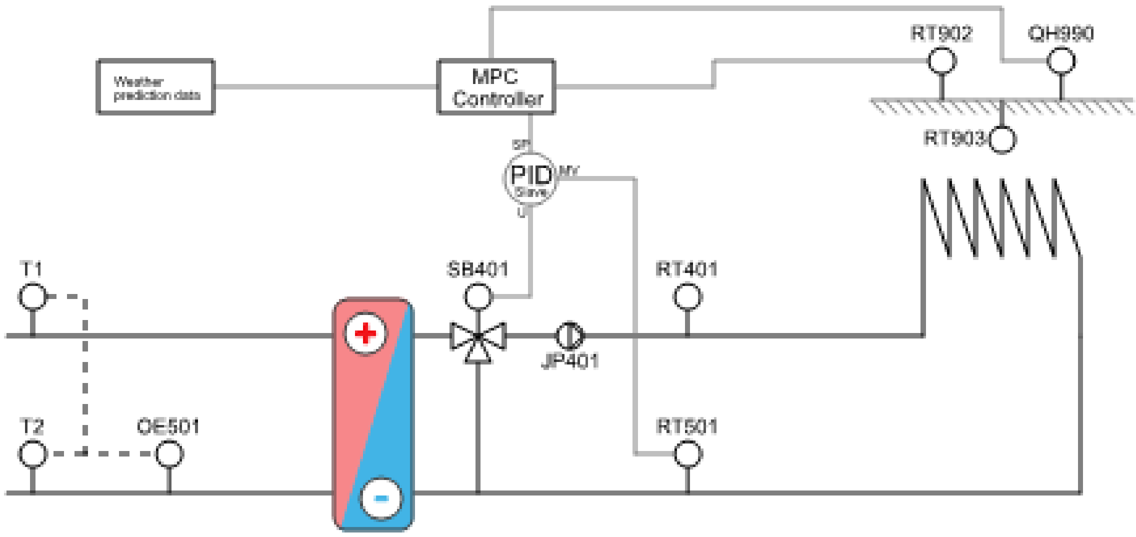


Figure 7.2: Sketch of the snow melting system with partial MPC control.

1.4, the prediction data from the weather forecast is used to determine when to start *Preheating Low* and *Preheating High*. *Frost Protection* is also subject to weather data but is not to be included in the MPC.

The mode *Melting* is dependent on only current outdoor air temperature and the detection of humidity on the *Snøostat*, and is not dependent on weather data. This mode can be excluded from the MPC on the grounds that there is no advantage to gain from predicted data in a feed forward manner. PID control is efficient in temperature control when the error is relatively low, so when the pavement has been preheated using the Preheating Low and Preheating High modes PID control is sufficient for keeping the temperature in the pavement stable. Some consequences of using PID control with sudden change in reference value is discussed in a later chapter.

The discussed simplifications in scope for the MPC reduces the model to only include two variables, T2 and RT902. The model is then given by eq. 7.15 and 7.16, the free variables for tuning is presented in Table 7.3 and a sketch of the control system can be seen in Figure 7.3.

$$\frac{dy_{T2}(t)}{dt} = \frac{-y_{T2}(t) + K_{T2}u_{SB401}(t-\theta_{T2})}{\tau_{T2}} \tag{7.15}$$

$$\frac{dy_{RT902}(t)}{dt} = \frac{-y_{RT902}(t) + K_{RT902}u_{SB401}(t-\theta_{RT902})}{\tau_{RT902}} \tag{7.16}$$

Table 7.3: List of parameters for the FOPDT model with some simplifications

	Gain	Dead time	Time constant
T2	K_{T2}	θ_{T2}	τ_{T2}
RT902	K_{RT902}	θ_{RT902}	τ_{RT902}

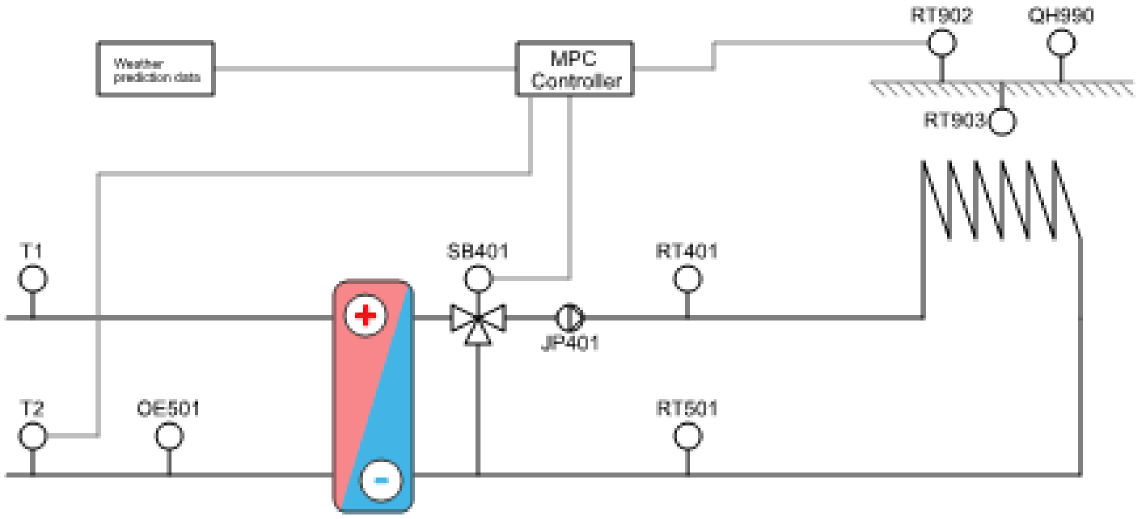


Figure 7.3: Sketch of the snow melting system with MPC for control of Preheating.

7.5 Discussion

As discussed in Chapter 6 the variables to be optimized are pavement surface temperature and primary side return temperature. In all the presented models these variables are available, and therefore feasible for use in the MPC. The question is which model is sufficient to create an MPC for controlling the snow melting system, while keeping the development and potential implementation complexity as low as possible.

The model presented in Alternative one gives an detailed description of the system, modeling almost all of the variables measured in the control system. The model is also mathematical simple as it represents all the variable as linear in relations to the input. The downside being that the computational power needed to simulate the model increases with the number of variables.

The model presented in Alternative two is a simplification of Alternative one in the sense that variables either redundant, not relevant or assumed to be constant are excluded, while still covering the needed scope for substituting the entire existing control system.

The model presented in Alternative three limits the scope of the model by redefining the input to the system as a controller setpoint rather than a direct actuator to the system. This limits the variables to be modeled, but introduces a non-linear relationship between inputs and outputs of the system. This makes the mathematical implementation of the model more complex.

Lastly the model presented in Alternative four is a further simplification of Alternative two, with additional adjustment is scope for the MPC, reducing the complexity further. This model contains sufficient information to control the system as desired, while only containing linear relations.

All the models has flaws regards to representation of the system in that assumptions on the behavior of the system has been made, such as the external variables not included in the models. As will be later discussed in further detail the number of variables in the model highly affects the computational power needed to simulate the system, and this is a big factor in the execution of a MPC. Therefore the number of variables in the model has made an impact on the choice of model for further work. This, along with the wish to only represent the system with linear relation and the fact that the use of predicted data is the focus of the MPC the model to further use in development of the MPC is the one presented in Alternative four.

8 Formulating the optimizing problem

The formulation of the Optimizing Problem(OP) can be viewed as describing the optimal control output, as described in chapter 6, in mathematical terms. Two factors are considered for this OP, the cost of running the system and the comfort of keeping the pavement free of Ice and Snow.

Starting by looking at the cost and comfort objectives separately, to determine how to handle them as OP. The cost of operating the snow melting system is defined in Chapter 6 as the error between the return temperature on the primary side T2 and the supply temperature on the primary side T1. The OP for cost is then minimizing problem where the sum of the error over the prediction horizon is minimized.

The comfort is defined as keeping the pavement surface temperature RT902 above a certain limit at the time of the predicted snowfall. This can be viewed as a setpoint tracking problem, where the OP is to minimize the error between the reference value and the measurement value for RT902. However, this would likely result in a control sequence where the error in temperature for RT902 is negative for some time after the reference value changes, resulting in lag in desired temperature on the pavement surface. An illustration of this assumed behavior is shown in Figure 8.1. This behavior is fine for a setpoint tracking problem, but for the snow melting system the reference for RT902 is a lower limit, and error on the negative side of this limit is undesirable. Therefore the reference value for RT902 can be viewed as a lower limit for the state RT902 in the OP formulation for T2, instead of a separate OP with setpoint tracking for RT902 as the main goal. In this way the two objectives are handled in one OP.

There are ways of solving Multi Objective Optimization Problems(MOOP) without defining one of the references as an limitation, but for this case that is not required. Further research into MOOP is left to the reader and is not to be further addressed here.

The detailed derivation of the objective function and the transition of it into Standard Quadratic Programming Formulation(SQPF) can be found in Appendix B. Only the main factors of the formulation is presented here.

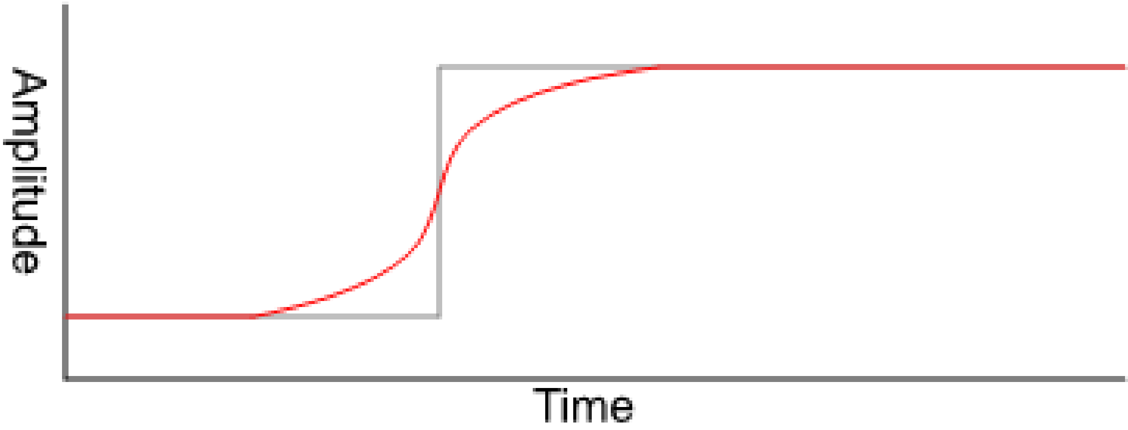


Figure 8.1: Plot showing the expected behavior of an OP for setpoint tracking. The red line illustrate the optimal solution while the gray line is the reference value.

8.1 Objective Function

An Objective Function(OF) is a representation of the desired function to optimize. It can be a minimizing or maximizing function of a goal that is desired to reach.[8] The general representation of the OF is expressed as

$$\underset{(\mathrm{x})}{min/max} \quad J = f(x) \tag{8.1}$$

s.t.

$$\begin{aligned} h_i(x) = 0, \quad i = 1,2,...,m \\ g_i(x) \leq 0, \quad i = 1,2,...,r \\ x_L \leq x \leq x_U \end{aligned} \tag{8.2}$$

Where $h_i(x)$ is Equality constraints, $g_i(x)$ is Inequality constraints, and x_L and x_H is the lower and upper bounds.

The OF derived for the described problem is expressed as

$$\underset{(\mathrm{u})}{min} J = \frac{1}{2}\sum_{k=1}^{N}(e^T Q_k e_k + u_{k-1}^T P_{k-1} u_{k-1}) \tag{8.3}$$

s.t.

$$\begin{aligned} e_k = T2_k - y_{T1} \\ x_{k+1} = Ax_k + Bu_k \\ y_k = Cx_k \\ u_L \leq u_k \leq u_U \\ x_L \leq x_k \leq x_U \end{aligned} \tag{8.4}$$

where Q_k is the weighting matrix for the error and P_k is the weighting matrix for the control signal. u_L lower limit for the inputs and u_U upper limit for the inputs. x_L lower limit for the states and x_U upper limit for the states. The main objective is here represented by $e_k = T2_k - y_{T1}$, describing the cost of running the system.

8.2 Standard Quadratic Programming Formulation

The solver desired to use to solve the OP requires the OF to be on Standard Quadratic Programming Formulation(SQPF), therefore the OF presented needs to be reformulated. The SQPF is a way of expressing the problem using well constructed matrices obtained using the Kronecker product.

The SQPF of the OF derived in Appendix B is

$$\overset{min}{(\mathrm{z})} J = \frac{1}{2} z^T H z + c^T z \tag{8.5}$$

s.t

$$\begin{aligned} A_e z &= b_e \\ A_i z &\leq b_i \\ z_L \leq z &\leq z_U \end{aligned} \tag{8.6}$$

where

$$z = \begin{bmatrix} u \\ x \\ e \\ y \end{bmatrix} \quad u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \quad e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{8.7}$$

$$H_{11} = \begin{bmatrix} P & 0 & \dots & 0 \\ 0 & P & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & P \end{bmatrix} \quad H_{22} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{8.8}$$

$$H_{33} = \begin{bmatrix} Q & 0 & \dots & 0 \\ 0 & Q & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q \end{bmatrix} \quad H_{44} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{8.9}$$

$$H = \begin{bmatrix} H_{11} & 0 & 0 & 0 \\ 0 & H_{22} & 0 & 0 \\ 0 & 0 & H_{33} & 0 \\ 0 & 0 & 0 & H_{44} \end{bmatrix} \tag{8.10}$$

$$c^T = \begin{bmatrix} 0_{N.n_u} \\ 0_{N.n_x} \\ 0_{N.n_e} \\ 0_{N.n_y} \end{bmatrix} \tag{8.11}$$

$$A_\varepsilon = \begin{bmatrix} -I_N \otimes B & I_{N\cdot n_x} - (I_{N-1} \otimes A) & 0_{(N\cdot n_x \times N\cdot n_y)} & 0_{(N\cdot n_x \times N\cdot n_y)} \\ 0_{(N\cdot n_y \times N\cdot n_u)} & -I_N \otimes C & 0_{(N\cdot n_y \times N\cdot n_y)} & I_{N\cdot n_y} \\ 0_{(N\cdot n_y \times N\cdot n_u)} & 0_{(N\cdot n_y \times N\cdot n_x)} & I_{N\cdot n_y} & I_{N\cdot n_y} \end{bmatrix} \quad B_\varepsilon = \begin{bmatrix} Ax_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \tag{8.12}$$

$$x_L = \begin{bmatrix} -\infty \\ -\infty \\ \vdots \\ -\infty \\ RT902_{k_L} \\ -\infty \\ \vdots \\ -\infty \end{bmatrix} \quad x_U = \begin{bmatrix} \infty \\ \infty \\ \vdots \\ \infty \\ \infty \\ \infty \\ \vdots \\ \infty \end{bmatrix} \quad u_L = 0\% \quad u_U = 100\% \quad z_L = \begin{pmatrix} 1_{N\times 1} \otimes u_L \\ 1_{N\times 1} \otimes x_L \\ -\infty_{(n_e)\times 1} \\ -\infty_{(n_y)\times 1} \end{pmatrix} \quad z_U = \begin{pmatrix} 1_{N\times 1} \otimes u_U \\ 1_{N\times 1} \otimes x_U \\ \infty_{(n_e)\times 1} \\ \infty_{(n_y)\times 1} \end{pmatrix} \tag{8.13}$$

The impotent aspects to address here, for understanding how the objectives are represented in the SQPF is $r_1, r_2, ..., r_k$ and $RT902_{k_L}$. $r_1, r_2, ..., r_k$ represents the reference value for error between T1 and T2, which as described earlier represents the cost of running the snow melting system. The reference is defined to be zero for every time step k, as the objective is to minimize the cost, thus keeping the cost as close to zero as possible. N is a variable in relation to the Prediction Horizon and will be discussed in greater detail later.

$RT902_{k_L}$ represents the earlier discussed lower limit for the pavement surface temperature for every time k.

As the lower limit for the surface temperature of the pavement(RT902) is dependent on the predicted snowfall, we can say that when it is not predicted snow the lower limit is equal negative infinity, and if there is predicted snowfall the setpoint is equal to $4°C$. If there is predicted snowfall within the next 6 hours the lower limit for the surface temperature of the pavement is set to $4°C$.

8.3 Prediction horizon

To understand how the $RT902_{k_L}$ is to be defined, the Prediction Horizon needs to be addressed. The Prediction Horizon is how far in the the future the system is to be simulated. The optimal control is predicted for N time steps forward in time relative to the current time. N then defines the prediction horizon. $RT902_{k_L}$ is then defined for $k = 0$ to $k = N$ based on the predicted snowfall. As an example it can be said that the prediction horizon is set to $N = 24$, and the time step is set to $1h$. The weather prediction calls for snow between 6 and 12 hours in the future. The $RT902_{k_L}$ will then be $4°C$ for $k = 6-12$ and $-\inf°C$ for $k = 0-6$ and $k = 12-24$. We here assume that the weather prediction will not change through the prediction horizon. An illustration of this example can be seen in Figure 8.2. Now imagine 6 hours have passed, the weather prediction has stayed the same so there is predicted snowfall within the next 6 hours. The $RT902_{k_L}$ will then be $4°C$ for $k = 0-6$and $-\inf°C$ for $k = 6-24$. An illustration of this example can be seen in Figure 8.3.

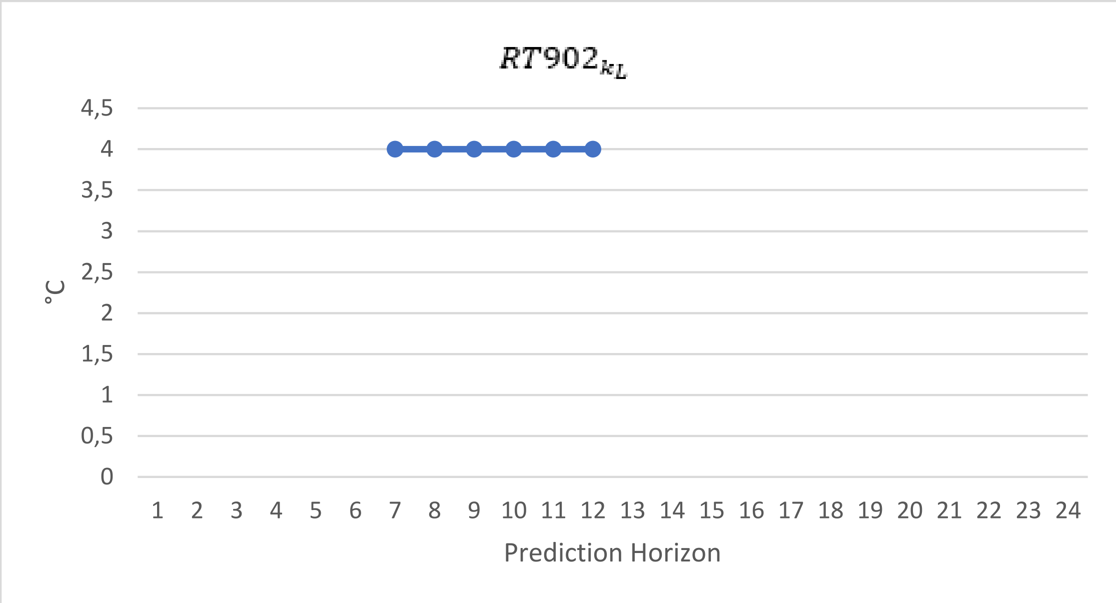


Figure 8.2: Example of $RT902_{k_L}$ for snow predicted between 6 and 12 hours in the future.

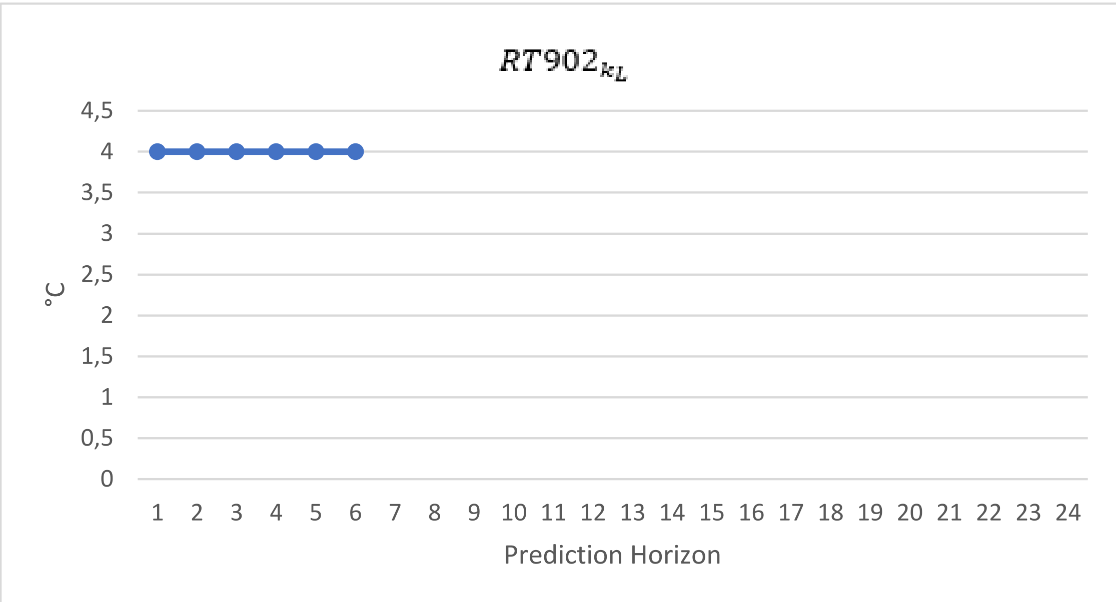


Figure 8.3: Example of $RT902_{k_L}$ for snow predicted between 0 and 6 hours in the future.

9 Implementing the model and development of MPC in Simulink

Now that the optimizing problem is defined and expressed on a form compatible with a solver it can be implemented in a suitable software. In this chapter the Simulink program developed for solving the optimizing problem is presented. The solver used in the program is presented. How the programs developed from the MPC is addressed and how this solution could be implemented for continues control of the snow melting system is discussed. The Moving Horizon concept is addressed in relation to continues control.

9.1 Simulink Program

The software for solving the optimization problem is developed using Simulink, a program delivered by MathWorks and is developed specially for Model-Based Design and is therefore optimal for developing the MPC. In addition the solver desired to use in this MPC, the `qpOASES` solver, is available for use with Simulink.

The developed MPC consist of four main parts. The Block program where the parts are connected and that handles plotting. A script for initialising the model and prediction horizon, as well as handling some PID control simulation for use in testing. The solver script, and finally a script for extracting the results from the solver.

9.1.1 Block Program

The Block Program can be seen displayed in Figure 9.1.

The right part, displayed in greater detail in Figure 9.2, handles the inputs from the initiation script, the solver script and the script for extracting the results.

The middle part of the Block Program handles offsets to correct values for the operating point of the simulation and calculating of energy based on simulated values and constants.

The left part of the Block Program handles plotting and storing of data. The data can be analysed using the plots in Simulink or be opened using Microsoft Excel

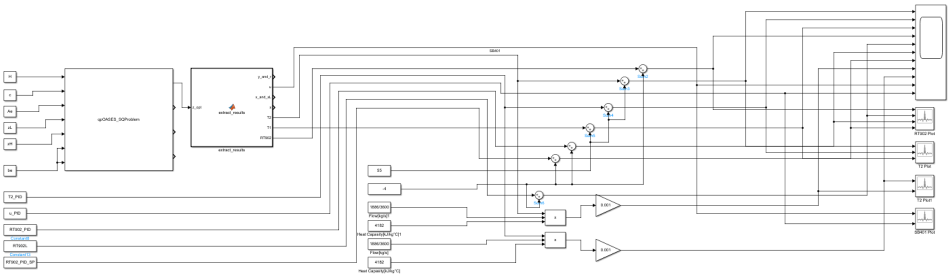


Figure 9.1: Simulink Block program for implementation of MPC for control of snow melting system

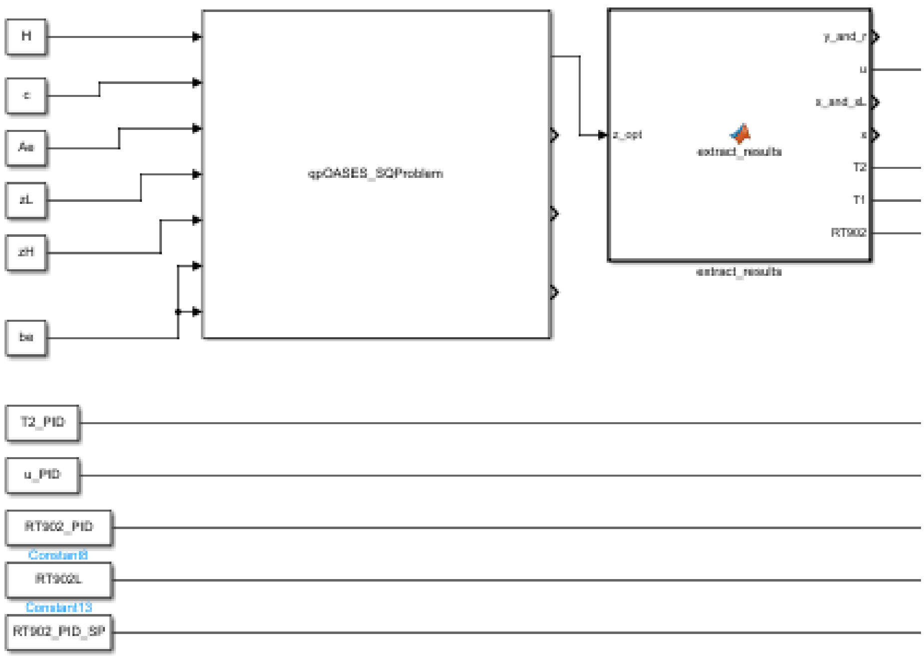


Figure 9.2: Part of Simulink block program handling inputs from initiation script, running the solver and extracting the results.

9.1.2 Initiation Script

The script for initializing the MPC can be seen in Appendix C. This script needs to be executed before the simulating Block Program can be started. The script handles the setup of the model parameters, defining the state space model matrices, setting up the standard quadratic programming problem, defining reference values and bounds and simulating control of the system using PID control for comparison with MPC.

9.1.3 Solving for the optimal solution

The Script `qpOASES_SQProblem` is the solver used to find the optimal solution to the objective function. The qpOASES is an open source C++ implementation of the Online Active Set Strategy to solve optimization problems on Quadratic Programming(QP) formulation. [17]

9.1.4 Extracting the results

The script for extracting the results from the solver in the MPC can be seen in Listing 9.1. This script extracts the resulting optimal solution from the solver and organizes the data for plotting and storing. This is needed as the data from the solver is in one long vector containing all inputs, outputs and states.

Listing 9.1: Code extracting the results from the Solver

```
function [y_and_r,u, x_and_xL, x, T2, T1,RT902]...
    = extract_results(z_opt, RT902L, r)
N = 96; %prediction horizon length
nx = 14; nu = 1; ny = 1; %no. of states, inputs and outputs
%extract results
Ua = z_opt(1+N*(0) :N*(nu),1); %control inputs
Xa = z_opt(1+N*(nu) :N*(nu+nx),:); %states
Ea = z_opt(1+N*(nu+nx) :N*(nu+nx+ny),:); %error in tracking
Ya = z_opt(1+N*(nu+nx+ny) :N*(nu+nx+ny+ny),:); %outputs
%we use the reshape function to rearrange the data
y_temp = reshape(Ya,ny,N);%arranged outputs(rows as
                                %signals, columns as data)
x_temp = reshape(Xa,nx,N);
x = x_temp';

%r = zeros(N,1);
T1=r';
%RT902L = [zeros(N/2,1); 1*ones(N/4,1); zeros(N/4,1)];
```

```
%RT902L = zeros(N,1);
T2=y_temp';
RT902 = x_temp(4,:)';
%Put it as the third column (for r1) and fourth column (for r2).
y_and_r = [T2,T1];
x_and_xL = [RT902,RT902L'];
u_temp = reshape(Ua,nu,N); %arranged input(row as signals, columns as data)
u = u_temp';
```

9.2 Implementation on existing control hardware

The implementation of the developed MPC can be approached in several different ways. As described in Chapter 3 the control system consists of several layers of hardware. The two alternatives for implementation is the Enterprise Server and the Automation Server. For optimal autonomy of the system the implementation should be implemented on the AS, to have the MPC closest to the sensors and actuators, and to eliminate the potential risk of the controller not working if the network between the AS and ES is down. As the controller uses external weather predictions to calculate the control signal, there is a potential risk of the controller not working because of network issues between the AS and the ES. The implementation of the MPC on the AS is challenging as the programming tools available are limited to proprietary solutions delivered by Schneider Electric, these tools are not initially designed for handling calculations of the form used in the MPC.

The alternative solution is to implement the MPC on the ES. The flaw of this solution is the potential risk of network issues between the ES and the AS, but as discussed this issue would also limit the weather prediction data being transferred to the AS. The ES has the same programming tools as the AS. As the ES is software installed on a server, it is possible to install other software such as Simulink on the same server. The MPC could then be implemented using Simulink, and gather and transfer data to the ES using some communication protocol such as MQTT or API. The control information is communicated between the ES and the AS on the existing proprietary solution. So for further development of the MPC, implementation on the ES is recommended.

9.3 Moving Horizon

The prediction horizon is earlier mentioned to be how far in the future the output of the system is modeled and the optimal solution is found. For continues control of a system this prediction horizon needs to be moving along with the real time, here the moving horizon is introduced. If the time the MPC is first initialised is denoted t_0 then the prediction horizon becomes from t_0 to t_{0+N} where N is the number of time steps in the prediction horizon. The optimal solution is now calculated for the solution by the MPC for the found prediction horizon. Now, to implement the moving of the horizon, only the first optimal input to the system, found by the MPC, is applied. Instead of considering the problem relative to the initial time t_0, the next time step is considered the initial time t_1. Thus moving the horizon forward by one time step so that the prediction horizon becomes from t_1 to t_{1+N}. Now the MPC calculates an entirely new optimal solution for the system, with new measurement for the initial states of the system.

This concept is not implemented in the developed MPC but needs to be considered for implementation when it is to be used for continues control. For testing of the MPC feasibility this functionality is not essential, as there is no feedback from the system available. By no feedback it is meant that there is no real or simulated system that can give a reaction to the action of the control signal found by the MPC, and therefore there is no new information about the states of the system that can be used after moving the horizon one step forward. This feedback could come from a secondary model of the system for testing purposes, but this option is not considered here. Implementing Moving Horizon is needed to further develop the MPC and is required before considering implementation on the real system. One potential flaw with MPC that comes with the Moving Horizon is static error is the system. Static error can be handled by implementing a integrator that shifts the contro, signal to remove the error. The static error is caused by the error in the model describing the system that is used to develop the MPC, as well as measurement error and resolution.

9.4 Discussion

The development of the MPC using Simulink is a good solution for testing the concept and comparison against PID control. The solution is partially transferable to the exciting control system hardware, so future implementation is possible to base on the developed programs. One aspect of the MPC that has potential for improvement is the choice of solver, as alternative solvers might give room for more memory efficiency and by that give room for adding more states. The Moving Horizon concept would need to be introduced into the MPC for continues control, but for testing the developed MPC is sufficient.

10 Fitting of the model simulation to data from trial runs

In this chapter the fitting of model parameters are presented. As discussed in Chapter 4 the data that is to be used is based on historical measurements from the snow melting system and trial runs. The model parameters to be found are the free variables of the FOPDT equations, process gain, process time constant and the process time delay for each of the two outputs, $T2$ and $RT902$, of the chosen model constant. Firstly the parameters describing the relationship between the control signal to the valve SB401 and the pavement surface temperature $RT902$ is defined, then the parameters for describing the relationship between the control signal to the valve SB401 and the primary side return temperature T2 is defined. Finally the quality of the found model parameters are discussed.

10.1 Valve control signal VS pavement surface temperature

As the opportunity to executing trial runs on the snow melting system while the weather conditions called for snowfall did not rise, the ideal data for obtaining the parameters process gain, process time constant and the process time delay is not available. Therefore the parameters is to be estimated based on historical data from normal operation of the system. This data has it flaws as it do not contain the step/response behavior as data from a trial run could have given.

In Figure 10.1 the data for SB401 and RT902 from the period discussed in Chapter 4 is displayed. In addition to the data some guiding lines are added to help read off the parameters.

From the plot we can read the process gain K_{T2} indirectly. The process gain K_{T2} can be calculated by dividing the step in input, here the control signal to the valve SB401, with the step in output, here the pavement supply temperature T2. The step in input needs to be approximated, as there is no clear step in the data available. The step is defined to happen at about 12.03.2022 06:30:00 and to be of an amplitude of 21%. This is at about the middle of the steepest rising curve of the control signal, and is the best approximation of a step available. The response in the output can first be seen at 12.03.2022 10:30:00, but do not reach a stable state until about 12.03.2022 15:30:00. Here the steady state is

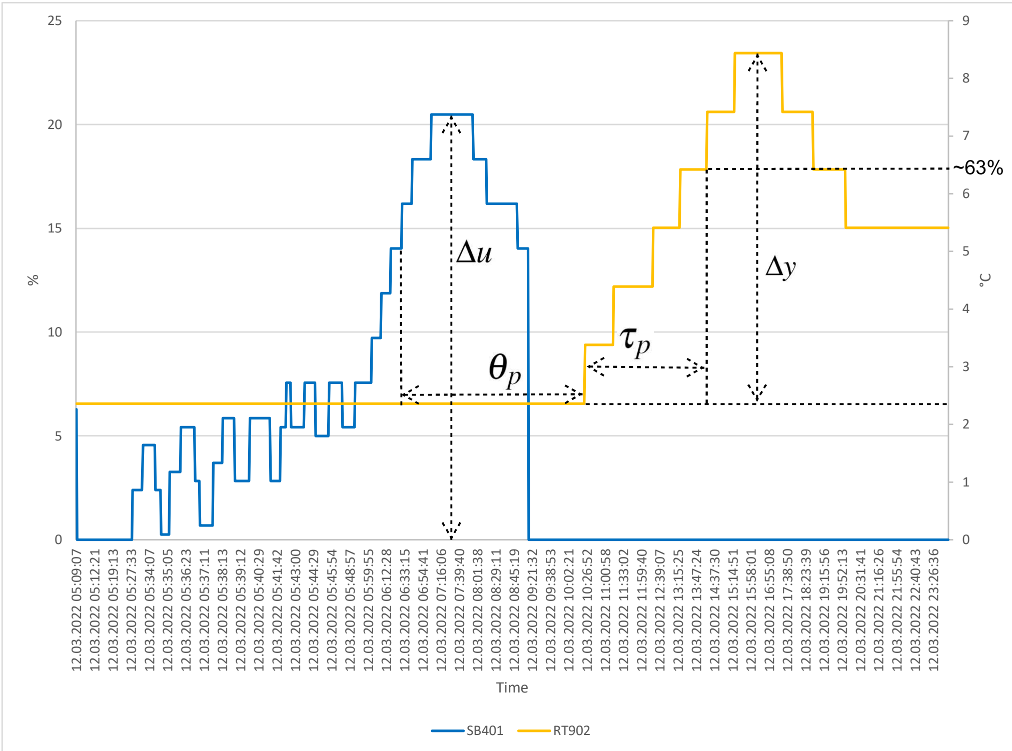


Figure 10.1: Plot of data gathered from SB401 and RT902 during normal operation when the system changes from not active mode. The parameters for describing the process in FOPDT terms are highlighted.

defined to be at the top of the $RT902$ curve, this because the decrees in temperature is assumed to come from the second negative step in input. The amplitude of the $RT902$ response is therefore approximately $6.3°C$. The process gain K_{T2} can then be calculated to be

$$K_{RT902} = \frac{6.3°C}{21\%} = 0,3\%/°C \tag{10.1}$$

Further the process time delay can be estimated using the same plot. The time delay is the time from step in input to a change in output can be seen in the output. The input step has been defined to occur at about 12.03.2022 06:30:00 and the following response can be seen at about 12.03.2022 10:30:00. The process time delay is then 4 hours.

$$\theta_{RT902} = 12.03.202210:30:00 - 12.03.202206:30:00 = 4h \tag{10.2}$$

Finely the time constant can be estimated. The process time constant is the time it takes from a change in the outputs happens to the output reaches 63% of the total change in amplitude relative to the new steady state. The value of the output $RT902$ is $2.3°C$ and the response change is earlier found to be $6.3°C$. The output reaches 63% of this change at about 12.03.2022 14:30:00. The time constant can then be calculated as

$$\tau_{RT902} = 12.03.202214:30:00 - 12.03.202210:30:00 = 4h \tag{10.3}$$

10.2 Valve control signal VS primary side return temperature

The historical data did not contain the sufficient information to determine the parameters process gain, process time constant and the process time delay to the relationship between process input $SB401$ and the output $T2$. Therefore a step/response test was preformed on the system to obtain this information. The test was not conducted under ideal conditions considering it was conducted in a warm spring day, but the ground was cool so the response in the system was as expected and sufficient to obtain the desired information. The reason this same data could not be used to analyze the behavior between the process input $SB401$ and the output $RT902$ is because the temperature at the pavement surface is highly affected by air temperature and the sun, so the response in this measurement is not comparable with whats expected at winter time. In Figure 10.2 the data from the step/response test can be seen plotted, along with some guiding indicators to help determine the parameters of the FOPDT process.

Following the same procedure as earlier, the step in process input is found to be 71% and the output response to have amplitude $-5°C$. The process gain is then given by

$$K_{T2} = \frac{-5°C}{71\%} = -0,07\%/°C \tag{10.4}$$

The time delay is found to be zero, as the response in output happens so close to the input change that it can be neglected. Therefore the time delay is given by

$$\theta_{T2} = 0h \tag{10.5}$$

The output reaches 63% of its amplitude change at about 19.04.2022 14:30:00. The change in output was first registered at about 19.04.2022 14:00:00. The time constant is then given by

$$\tau_{T2} = 19.04.202214:00:00 - 19.04.202214:30:00 = 0,5h \tag{10.6}$$

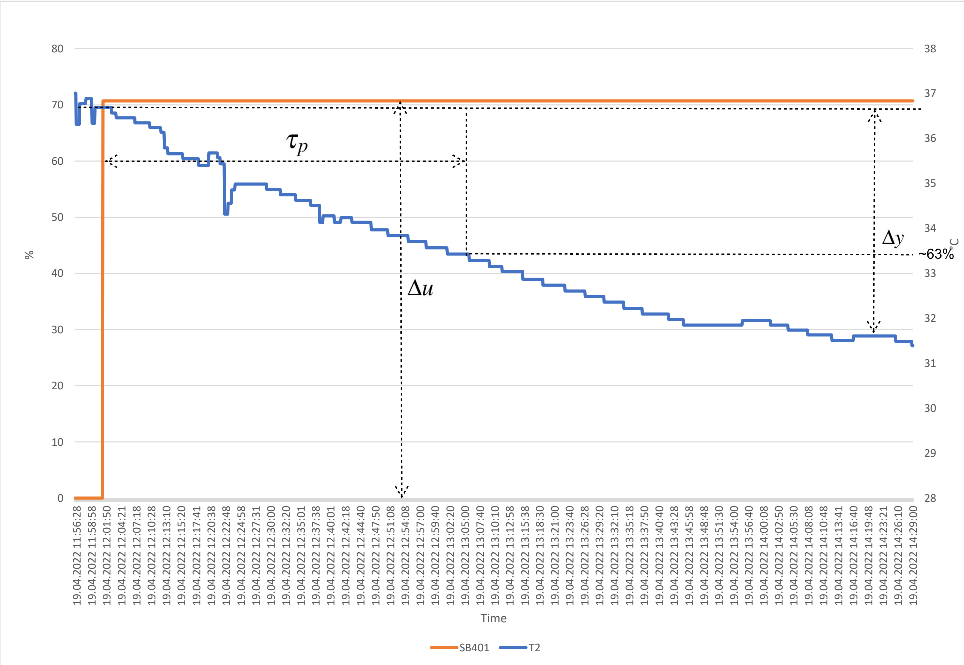


Figure 10.2: Plot of data gathered from SB401 and T2 during step/response test. The parameters for describing the process in FOPDT terms are highlighted.

10.3 Model tuning results

The resulting tuning parameters for the two FOPDT models describing that are used in the MPC for the snow melting system can be seen in Table 10.1.

Table 10.1: List of parameters for the FOPDT model with some simplifications

	Gain	Dead time	Time constant
T2	$-0,07\%/°C$	$0h$	$0,5h$
RT902	$0,3\%/°C$	$4h$	$4h$

10.4 Discussion

The process gain, process time constant and the process time delays found are sufficient for further use in testing the MPC. There is some uncertainty in the parameters as the data used to determine them are not gathered in an optimal manner. For the relationship between SB401 and RT902 the data is not from a step/response test but from historical data, and for the relationship between SB401 and T2 the data is from a step/response test but the test was not conducted under ideal conditions for snow melting. For further tuning of the parameters in the future it is recommended to perform one ore more step/response test under conditions where snowfall is plausible.

11 Testing the MPC by simulation

Testing of the MPC is performed on a simulated environment. The goal of the testing has been to determine if the MPC developed handles controlling the snow melting system in different states and when the snow is predicted at different times, and to determine if the MPC has the potential to outperform traditional PID in regards to both error handling and power efficiency. The key parameters for compering the outcome of the test cases as well as comparing MPC and PID for each case is total energy consummation, peak pavement surface temperature and lowest Primary side return temperature.

11.1 Test Cases

Test cases are defined by two parameters, the time interval of predicted snowfall and the initial pavement surface temperature. The time intervals for snowfall predictions are divided into the same intervals as the predictions are given in the existing control system, where weather data is gathered from MET. The time intervals are 6-12, 12-18, 18-24 and 24-30 hours from the time of control initialization. The initial pavement surface temperature chosen for the test cases is based on what temperature corresponds with plausible snowfall. The ideal air temperature for the formation of snow is just below zero. The conditions for snowfall commonly appear in the morning, when the air temperature is increasing from a minimum during the night. The pavement surface will in these condition commonly be somewhat lower than the air temperature. The chosen initial pavement surface temperatures for the test cases is $0°C$, $-4°C$ and $-8°C$, with the assumption that this correspond with an air temperature in the range of $-4°C$ to $4°C$.[18]

The test cases are named to reflect the parameters described. The name of the test case is structured as: `Snowfall interval:Initial temperature`. As an example the test case where snow is predicted between 6 and 12 hours from the control initiation and the initial pavement surface temperature is $-4°C$ the name of the test case is `6-12:-4`. a list of all the test case names and corresponding conditions can be seen in Figure 11.1.

In Table 11.2 a list of the test cases can be seen, with resulting key parameters listed.

In Figure 11.1 the resulting behavior of test case `12-18:-4` can be seen, this illustrates the data that will be discussed later for test cases of relevance. The results will mainly be discussed based on key parameters across test cases.

Table 11.1: Table of test cases and corresponding conditions

Test number	Initial temperature of pavement [°C]	Time period for snow prediction [h]
6-12:0	0	6 to 12
6-12:-4	-4	6 to 12
6-12:-8	-8	6 to 12
12-18:0	0	12 to 18
12-18:-4	-4	12 to 18
12-18:-8	-8	12 to 18
18-24:0	0	18 to 24
18-24:-4	-4	18 to 24
18-24:-8	-8	18 to 24
24-30:0	0	24 to 30
24-30:-4	-4	24 to 30
24-30:-8	-8	24 to 30

11.2 Energy

The total energy consumption for the test case gives information on the performance of the controller relative to each other. Higher energy usage is directly correlated to higher cost. In Figure 11.2 the total energy consumption for each test case is displayed. The blue bar is the total energy consumption for the test case controlled by MPC, and the orange bar is the consumption for the test case controlled by PID. The energy consumption is generally lower for the cases controlled by MPC, with only one exception being the test case where snow is predicted between 6 and 12 hours from control initiation, and the initial pavement surface temperature is $-8C$. How this relates to the peak in pavement surface temperature and lowest primary side return temperature will be discussed later. For the test cases where the snow is predicted between 12 and 18, 18 and 24, and 24 and 30 hours from the control initiation there is a trend of MPC control generating less energy usage than PID control. The deviation in energy usage cased by MPC and PID increases when the pavement surface temperature decreases.

This indicates that the MPC controller generally performs better with regards to energy usage, especially for cases where the temperature is closer to zero, but also for the cases where the temperature is lower as long as the snowfall is predicted earlier than 12 hours from the control initiation.

11.3 Pavement temperature

The peak pavement surface temperature is used to show how much the desired temperature is overshoot by, giving an indication of how quick the controller is to correct for the error, and how it handles the quick change in setpoint. In Figure 11.3 the maximum

Table 11.2: Table of key results from test runs executed on the snow melting system MPC

Test number	Initial temperature of pavement [°C]	Time period for snow prediction [h]	MPC Energy Consumtion [kWh]	PID Energy Consumtion [kWh]	MPC RT902 peak temperature [°C]	PID RT902 Peak temperature [°C]	MPC T2 peak temperature [°C]	PID T2 Peak temperature [°C]	PID reached desired surface temperature at time of snowfall	MPC reached desired surface temperature at time of snowfall
6-12:0	0	6 to 12	75,0	83,7	7,3	6,2	43,9	50,9	No	Yes
6-12:-4	-4	6 to 12	176,8	193,4	12,9	10,2	30,5	46,2	No	Yes
6-12:-8	-8	6 to 12	355,0	303,6	25,5	14,3	10,5	41,5	No	Yes
12-18:0	0	12 to 18	65,1	83,7	4,4	6,2	52,0	50,9	No	Yes
12-18:-4	-4	12 to 18	130,2	193,4	4,9	10,2	49,1	46,2	No	Yes
12-18:-8	-8	12 to 18	195,2	303,6	5,3	14,3	46,1	41,5	No	Yes
18-24:0	0	18 to 24	71,3	83,7	4,4	6,2	52,3	50,9	No	Yes
18-24:-4	-4	18 to 24	142,5	193,4	4,7	10,2	49,5	46,2	No	Yes
18-24:-8	-8	18 to 24	213,8	303,6	5,1	14,3	46,8	41,5	No	Yes
24-30:0	0	24 to 30	73,1	83,7	4,4	6,2	52,3	50,9	No	Yes
24-30:-4	-4	24 to 30	146,2	193,4	4,7	10,2	49,5	46,2	No	Yes
24-30:-8	-8	24 to 30	219,3	303,6	5,1	14,3	46,8	41,5	No	Yes

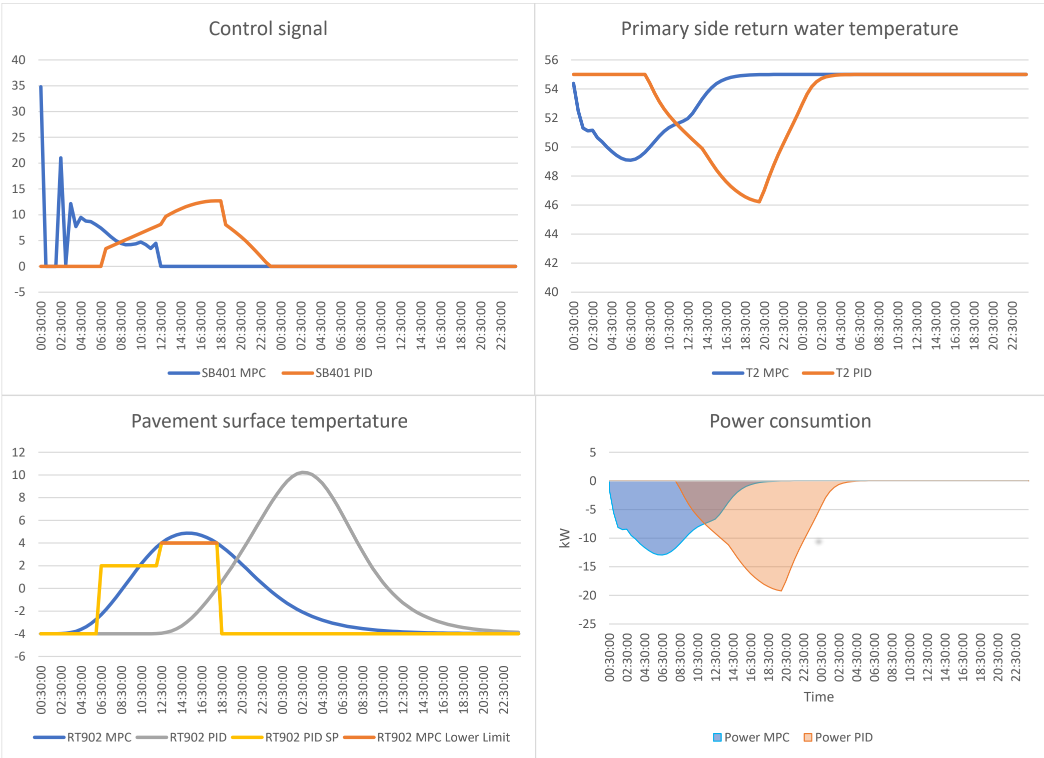
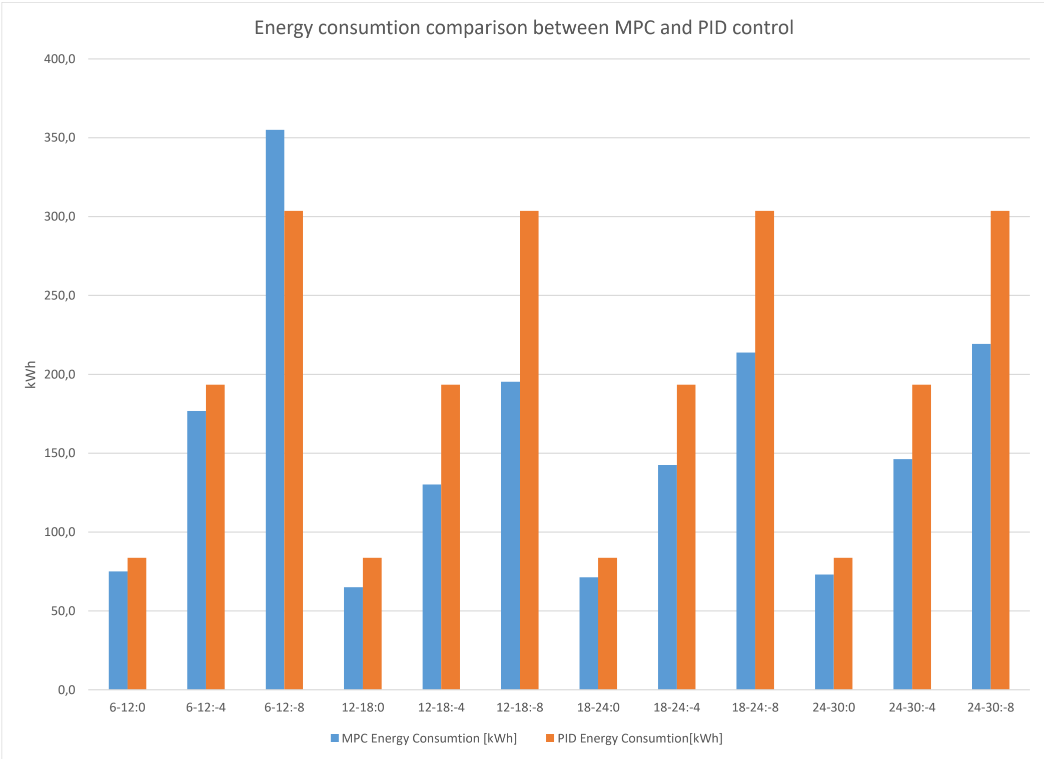


Figure 11.1: Plots showing the resulting snow melting system behavior of test case `12-18:-4`.

pavement surface temperate for each test case is plotted. The blue bar is the peak surface temperature for the test case controlled by MPC, and the orange bar is the temperature for the test case controlled by PID.

For the test cases where snow is predicted earlier than 12 hours from the control initiation the peak temperature is higher for PID control than for MPC, and the deviation increases the initial temperature decrease. While for the test cases where snow is predicted between 6 and 12 hours from the control initiation the MPC control results in a higher peak than PID control, especially for the test case `6-12:-8` which results in the overall highest peak. This is the same test case as the one where PID outperforms MPC regards to energy usage. The correlation here can be that the peak in temperature caused by the MPC requires a lot of energy.



Internal

Figure 11.2: Comparison of total energy usage for MPC and PID for each test case.

11.4 Primary side return temperature

The lowest primary side return temperature for each test case can be seen in Figure 11.4, where the orange rhombus represent the resulting lowest primary side return temperature from PID control while the blue triangle represent the temperature resulting from MPC control.

The grouping of behavior in temperature deviation relative to the primary side supply temperature($55°C$) is the same as for the peak pavement surface temperature. The three test cases where snow is predicted between 6 and 12 hours from the control initiation the MPC control results in a lower minimum temperature compared to the result from PID control for the same test case. For the other test cases PID has a lower minimum temperature than MPC. The most extreme case is, as for peak pavement surface temperature, the test case `6-12:-8`.

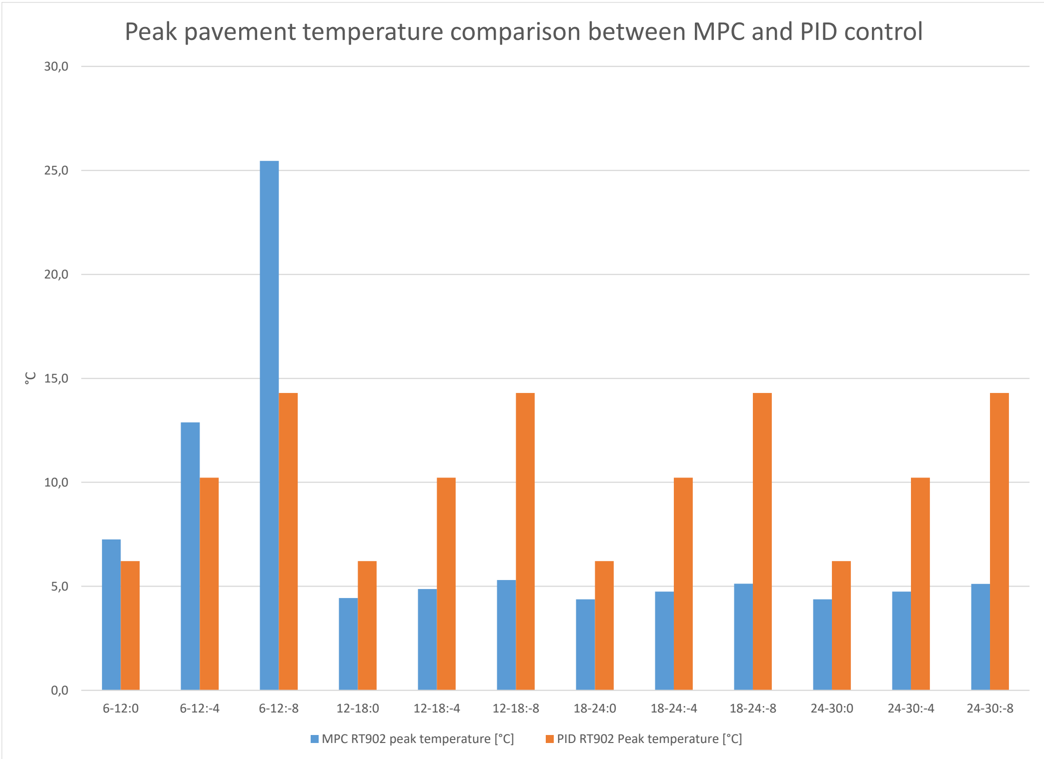


Figure 11.3: Comparison of maximum pavement surface temperature for MPC and PID for each test case.

11.5 Discussion

From the results discussed one can argue that MPC has an overall better performance than PID control in controlling the snow melting system regards to energy consumption. For only one of the test cases the MPC results in a higher energy consumption than PID control.

As can be seen in Table 11.2, when it comes to obtaining the desired temperature in the pavement surface PID control do not for any of the test cases reach the desired value, while MPC do this for every case. It is important to note that the simulation of the PID control has some flaws in imitating the existing control system. The logical components, as well as the cascade controller from the existing control system is not implemented in the simulation of PID control. However, the approximation of the existing control system imitating the system behavior in a sufficient for the purpose of testing MPC.

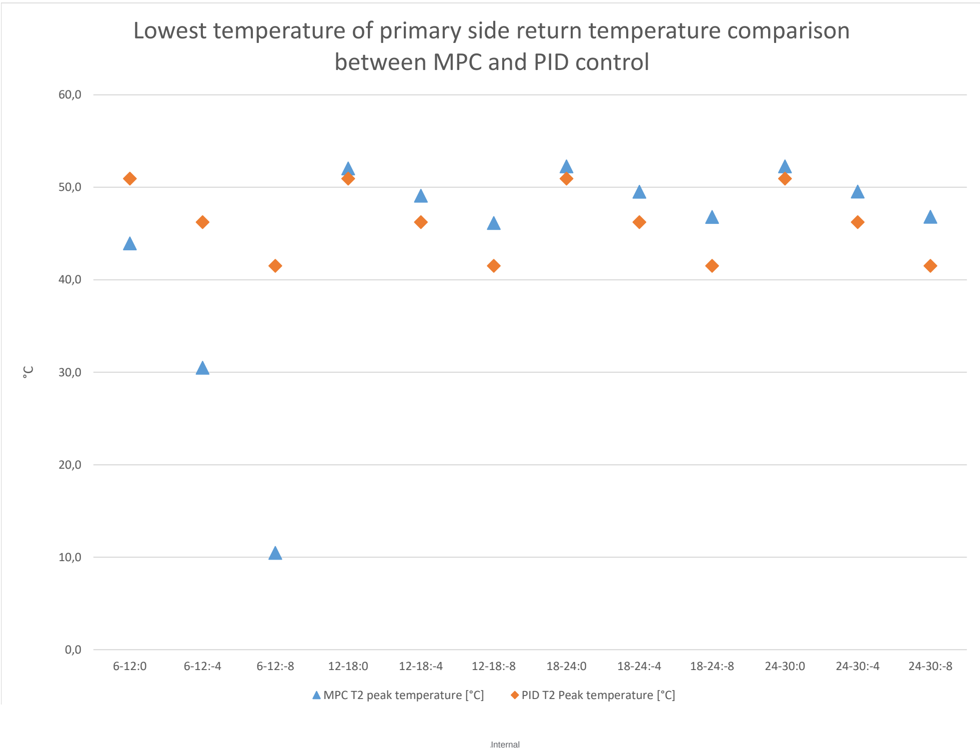


Figure 11.4: Comparison of lowest primary side return temperature for MPC and PID for each test case.

For looking closer on the difference in system behavior for PID control and MPC the results from the test cases `12-18:-4` and `6-12:-8` is further analysed. `12-18:-4` representing a test case where MPC allegedly performs better than PID, and `12-18:-4` representing a test case where PID allegedly performs better than MPC. In Figure 11.1 and 11.5 we can see four plots, one in the upper left corner showing the control signals, one in the upper right corner showing the resulting primary side return temperature, one in the lower left corner showing the resulting pavement surface temperature along with the reference value for the PID controller and the lower limit for the MPC controller, and one in the lower right corner showing the energy subtracted from the primary side of the snow melting system. What is not discussed earlier is the behavior of the control signal. As seen by looking at the plots for the control signal for both test case `12-18:-4` and `6-12:-8`, the resulting control signal caused by MPC changes rapidly while the result from PID control is a smother response. Rapid changes in control signal may cause more stress to the system and the field equipment, this is argues in favor of PID control.

The test case `6-12:-8` is the only test case that had poorer results for MPC compered to PID in regards to energy consumption. By looking at the results from test case `6-12:-8`, shown in Figure 11.5, it can be seen that the control signal from the MPC controller is 100% for approximately two hours. The continuous high control signal is what causes the large overshoot in pavement surface temperature and following high energy consumption. This high control signal is required for the pavement surface temperature to reach the lower limit within the time of the predicted snowfall. When implementing the Moving horizon concept this problem will not occur, unless the prediction of snow is not foreseen by MET before it is less than 12 hours until it is going to snow. When the snowfall is earlier predicted, the MPC will get the information earlier and the resulting solution will look somewhat closer to the test cases where snow is predicted from 12 hour or earlier.

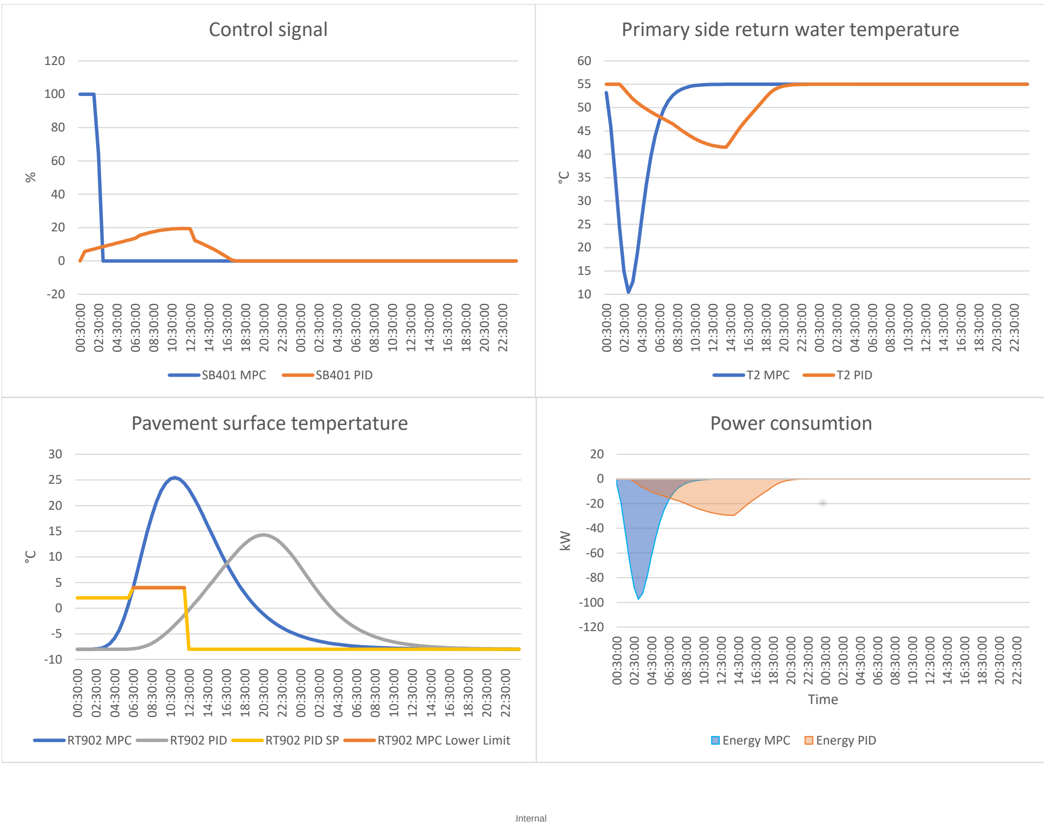


Figure 11.5: Plots showing the resulting snow melting system behavior of test case `6-12:-8`.

12 Economical Evaluation

This chapter investigates the cost of implementation of the developed MPC for control of a snow melting system. Firstly the implementation cost is evaluated, considering further development, installation on the discussed system and potential installation on additional systems. The potential benefit of implementing MPC is discussed and a Net Present Value analysis is preformed to determine the economical sustainability of the project.

12.1 Cost of implementation

The cost of implementing a MPC on the ES to control is challenging to estimate, as there are potential challenges that are hard to foresee. The known required work to be done for implementation is to further develop the MPC and implement the Moving Horizon strategy, establishing the working MPC to run on the sever hosting the ES, establish communication between Matlab Simulink and the ES, and implement the new control system on the AS for it to control the actuators and receive measurement data. There is also a cost in the license for Matlab. Fortunately there is no need for new hardware, unless additional is found to be required in later research.

A rough overview of the implementation cost can be seen in Table 12.1.

When considering the cost it is important to consider the potential of using the developed control system for other sites resembling the one at Åskollen. Drammen Eiendom has about seven snow melting systems that could be subject to control using MPC. Some of these snow melting system are hosted by the same ES as the one at Åskollen. For

Table 12.1: Table estimated cost of implementation of MPC for snow melting.

Task	Hours	Cost
Further development and implementation of MH		-
Install MPC on ES	8	9600
Establish communication between Matlab and ES	16	19200
Implementation of MPC on AS(actuators and measurements)	16	19200
Matlab License		20000[19]
Duplication for new site	8	9600

these system the implementation cost of MPC would be much lower as the initial system is already running. The needed work to duplicate the MPC to control a separate snow melting system would be to adjust the sources of information and tune the model to fit the relevant site.

12.2 Operational benefit

Based on the difference between the resulting energy usage for PID and MPC for each test case it is possible to do some estimates about the potential cost benefit of implementing MPC.

The cost of energy is dependent of source and varies with time. For the snow melting system at Åskollen the energy is supplied from a heat pump, and translating the cost of running the heat pump to delivered energy is complex. As many snow melting systems are supplied with district heating, using prices from this source is a good substitute. According to Fortum, a supplier of district heating, the average price of 1 kWh is 2.02NOK in March 2022. This correspond well with information given by Mats Akselsen at Entra ASA, estimating an average price of 2.01NOK/kWh for the period from the start of 2022 to the end of April 2022[20]. The average price for the entire year of 2021 is somewhat lower, but the pricing based on the winter months are more relevant as the prices vary with the seasons. There are some variations based on energy consumption peaks, as the price is higher over some kW limit, but this is not considered here.[21]

In Table 12.2 a list of the test cases and corresponding savings of using MPC compared to PID control is displayed. The potential savings are highest for the test cases where the initial temperature is lowest, except for the test case `6-12:-8` where the MPC results in higher energy usage than PID. The reason behind this has been discussed earlier, and this case would most likely not be relevant after implementing the Moving Horizon strategy.

The average savings for all the test cases is found to be 77NOK per case of predicted snowfall and the pavement is preheated. To get a yearly potential saving the number of days that have snowfall needs to be considered. Oslo has an average of 31 snow days per year, according to statistics gathered from 1937 to 2012.[22] Using the average savings over all the test cases, and multiplying this with 31 days of snowfall we get a yearly saving of about 2400NOK.

When the MPC is implemented on the ES, the additional work for duplicating the system to cover other locations is as discussed less than for the first site.

Drammen Eiendom has about 7 locations with snow melting, not all of them currently controlled by Schneider Electric an therefore not hosted by the ES, but the potential of future implementation is present. If including all 7 sites, the potential yearly saving is

Table 12.2: List of test cases and corresponding potential saving

Test number	Difference in energy consumtion for PID and MPC [kWh]	Difference in cost of operationfor PID and MPC [NOK]
6-12:0	-8,6	-17,4
6-12:-4	-16,7	-33,6
6-12:-8	51,4	103,8
12-18:0	-18,6	-37,5
12-18:-4	-63,3	-127,7
12-18:-8	-108,4	-218,9
18-24:0	-12,4	-25,0
18-24:-4	-50,9	-102,7
18-24:-8	-89,8	-181,4
24-30:0	-10,5	-21,3
24-30:-4	-47,2	-95,3
24-30:-8	-84,3	-170,2

about 16800NOk. In this calculation the variation in system size is not considered, so there is some uncertainty in varying potential savings.

12.3 Evaluation of cost VS benefit

To evaluate if the cost of implementing MPC control of the snow melting system can be justified by the potential savings, a Net Present Value(NPV) analysis is performed. Here the cost of implementation is compared against the savings over a 10 year period. A discount rate of 5% is assumed. Firstly a NPV analysis is performed considering implementation only at Åskollen. The resulting NPV can be seen in Table 12.3. Here the only cost is in the first period. This is a one time implementation cost as discussed earlier. The cost of operating the hardware, and maintenance of hardware and supporting software is not included in the analysis, as these costs will be present regardless of control methods. If the MPC is not implemented, and the existing control system is retained the same costs for maintenance are required, and can therefore be excluded from the analysis.

The savings is the saved cost of energy from implementing MPC compared with using the existing control system.

By looking at the column ”Net Present Value” in Table 12.3 it can be seen that the cost of implementation is not justified within the ten year time frame. Keep in mind that positive numbers are costs and negative numbers are savings.

Table 12.3: Net Present value analysis for implementation at Åskollen. Discount rate 5% and a time frame of 10 years.

Net Present value analysis for implementation at all snow melting sites managed by Drammen Eiendom. Discount rate 5%					
Period [year]	Cost [NOK]	Savings [NOK]	Cash Flow [NOK]	Discaunted cash flow [NOK]	Net Present Value [NOK]
0	68000	-2396	65604	65604	65604
1	0	-2396	-2396	-2282	63322
2	0	-2396	-2396	-2174	61148
3	0	-2396	-2396	-2070	59078
4	0	-2396	-2396	-1971	57107
5	0	-2396	-2396	-1878	55229
6	0	-2396	-2396	-1788	53441
7	0	-2396	-2396	-1703	51738
8	0	-2396	-2396	-1622	50116
9	0	-2396	-2396	-1545	48571
10	0	-2396	-2396	-1471	47100

Secondly a NPV analysis is performed considering implementation of MPC for controlling the snow melting systems for all seven sites managed by Drammen Eiendom. Still with a time frame of ten years and a discount rate of 5%. The cost in the first period is now somewhat higher due to the cost of duplicating the system for the 6 additional sites. The yearly savings is 7 times higher as the number of cites has increased, giving a yearly saving of 16800NOK. In Table 12.4 the NPV analysis can be seen. By looking at the column "Net Present Value" the cost is justified by year five, as the NPV turns negative, indicating that the cost of implementation is covered.

Table 12.4: Net Present value analysis for implementation at all snow melting sites managed by Drammen Eiendom. Discount rate 5% and a time frame of 10 years.

Net Present value analysis for implementation at all snow melting sites managed by Drammen Eiendom. Discount rate 5%					
Period [year]	Cost [NOK]	Savings [NOK]	Cash Flow [NOK]	Discounted cash flow [NOK]	Net Present Value [NOK]
0	77600	-16774	60825,95	60826	60826
1	0	-16774	-16774	-15975	44851
2	0	-16774	-16774	-15215	29636
3	0	-16774	-16774	-14490	15146
4	0	-16774	-16774	-13800	1346
5	0	-16774	-16774	-13143	-11797
6	0	-16774	-16774	-12517	-24314
7	0	-16774	-16774	-11921	-36235
8	0	-16774	-16774	-11353	-47588
9	0	-16774	-16774	-10813	-58401
10	0	-16774	-16774	-10298	-68699

13 Results Summary

It is found that the use of MPC for control of snow melting systems using heat exchangers and heated liquid to melt snow is feasible. Different models of the system have been suggested and one has been chosen to use as basis for development of a MPC. The MPC Has been developed using Simulink as the simulation environment giving flexibility in the test stage.

The MPC had been tuned to data from the snow melting system, gathered both from historical data and step/response tests do obtain information about the dynamics of the system. This data is not optimal for fitting of the model and the MPC, but sufficient for the purpose of testing.

The MPC has been tested in a simulated environment, where snow prediction is simulated. The MPC is compared with a simulation of a simplification version of the existing control system. In 11 of the 12 test cases, where the test cases different in time of predicted snowfall and initial temperature of the pavement surface, the MPC results in less energy usage compered to PID control. The difference in energy usage for the 11 cases where MPC outperforms PID range from $8,6kWh$ to $108,4kWh$, averaging at $38,3kWh$ for all 12 test cases.

The MPC manages to reach the desired pavement surface temperature for every test case, while the result from PID control is opposite. Here it is important to note that this trend, of the existing control system not managing to bring the pavement surface temperature to the desired temperature, is found in historical data from operation of the snow melting system, and is not caused by poor simulation or tuning.

The economical benefit of implementing MPC for controlling snow melting systems is analysed. Based on the found average saved energy from implementing MPC, an average saving in cost per day of predicted snow is found to be 77NOK. Using the average number of days with snowfall per year, 31 days, gives a yearly saving of 2400NOK/year for the system at Åskollen. It is found that the cost of implementing the MPC for only one system, in this case the system at Åskollen, is not economically justifiable within a time frame of ten years. However, the cost of implementation is found to be highest for the first system, and the cost of duplicating the system for control of additional sites is much lower. When considering implementation on the seven snow melting systems known to be managed by Drammen Eiendom, the cost of implementation is found to be covered by the saving of energy cost by the fifth year of operation.

14 Discussion

In regards to defining the scope of the developed MPC, some functionality is excluded from consideration when considering what part of the existing control system to include. The existing control system does not only cover the discussed modes for control of the temperature, but also some security functions for making sure the system do not take harm under normal operation. One such function is frost protection. This function is not considered to be implemented in the MPC, but is left to the existing control system. This solution is fine in regards to security, but some fail safe for the MPC if the system is to go in frost protection mode should be considered.

Peak energy usage is an unwanted symptom of fast control that is not considered by the MPC. If the needed peak of energy is to high the system delivering the energy can have problems keeping the temperature stable. This is not considered in the MPC, but can in the future be included as a limitation on the primary side return temperature.

The complexity of the model used to develop the MPC has been reduced by not including some of the variables. This lead to a fairly simple model only describing the behavior of two variables. The main reason for the reduction of model complexity has been the limitations in computing power needed to solve the optimization problem. The chosen solver requires the optimization problem to be structured in a way that is well organized in regards to readability, but that contains a lot of empty space in regards of data allocation. Other solvers exist that has a more compact structure and is more efficient in regard to memory usage.

One of the reductions of the model is justified by arguing that PID control is sufficient for controlling the pavement surface temperature once the pavement is preheated by the MPC. There are potential risk factors to consider in the transition between MPC and PID control. If the weather forecast misses when the snowfall occurs, the mode 'Melting' can be activated while the pavement is not sufficiently heated and the PID controller used in the Melting mode can be exposed to a large error. The PID controller needs to be sufficiently tuned to handle such a case and not result in unstable control.

The data gathered for tuning the model used in the MPC is from a mixture of historical data and test runs conducted under conditions where snowfall is not likely to occur. The data was sufficient for tuning the model for testing, but for further development and testing data from well executed stepresponse test runs should be used for tuning of the model.

The Moving Horizon concept is only discussed and is in the duration of this project not implemented or tested for the developed MPC. To further prove the concept of using MPC for control of snow melting systems, the Moving Horizon strategy should be implemented and tested on either a secondary model or on the real process. This to verify that the MPC is fitted for continous control.

To verify the performance of the developed MPC a simplified version of the existing control system is used as a benchmark. The simplified version of the existing control system is found to be sufficient, but do not describe all the aspects of the existing control system. Some logical control and the slave controller of the cascade setup for the existing control system is not included, but the overall behavior is well represented by the one PID controller used.

The performance of the MPC is only evaluated on the bases of results from simulations. For further evaluation experiments on a real system will need to be conducted. All results, both for performance and economical, is subject to the error and assumptions made when modeling the system behavior.

The economical evaluation is based on averaged pricing and assumption that the systems are equal in regards to potential savings and implementation costs. Variations in instrumentation for the different sites is not considered, and size of the sites are not factored into the calculations.

The price found to be the average savings for all the test cases is only based on the chosen variations in test criteria. The test cases might not be representative for the variation in conditions when the system is in use. Some test cases will probably be more representative for the conditions in which snowfall is probable, but this is not considered her.

15 Conclusion

Several models for describing the behavior of the snow melting system have been found. The models differ in complexity based on relationship between input and output, and in the number of variables included. A FOPDT model describing the relationship between the valve and the pavement temperature and the relationship between the valve and the primary side return temperature has been chosen to develop a MPC.

A MPC, using the found model tuned with data from the reals system, has been developed using MathWorks Matlab Simulink. The MPC is tested with simulations and the results show that using MPC for snow melting systems based on heated water/alcohol with water-to-water heat exchangers is feasible.

It is found that the developed MPC performs well compared to traditional PID control. For 11 out of 12 test cases conducted, the MPC results in less energy usage than PID. For 9 out of 12 test cases MPC results in less pavement surface temperature overshoot and less deviation in primary side return temperature, when compared to PID. The test cases where PID results in better results than MPC correlates in how far in the future the predicted snowfall is introduced. If the prediction is introduced less than 12 hours before the snowfall is to occur, the MPC results in poorer results compared to PID.

It is found that the MPC results in a potential average energy saving of $38kWh$ per period of predicted snowfall, corresponding to a saving of 77NOK per period of predicted snowfall. If the MPC is only considered for implementation on one site of the size as the one at Åskollen, it found to not be economically justifiable within a time frame of 10 years. The potential saved cost from less energy usage do not cover the estimated cost of implementing the system. However. if considered to be implemented on the seven sites known to be managed by Drammen Eiendom it will be economically beneficial within 5 years of implementation. Implementation on fewer sites will also be beneficial, but this is not analysed.

For future work it is recommended to implement the Moving Horizon strategy in the MPC for further testing. It is recommended to perform stepresponse tests to obtain better data for fitting of the model. The test run should be executed under conditions where snowfall is probable. The MPC can be implemented on the server hosting the ES and necessary software for communication between MathWorks Matlab Simulink and the ES can be developed. It is recommended to implement the developed MPC in the existing control system and integrate the existing solution for handling security functions.

Bibliography

[1] K. B. Jonsson, 'Bruk av værprognoser for optimal styring av snøsmelteanlegg,' nor, 2014.

[2] Photoon and Wichakool, 'System modelling and controller designed for thermoelectric generator using a first order plus dead time,' eng, 2016.

[3] S. Werner. 'The Navigance Blogchemical process models – from first principle to hybrid models.' (2020), [Online]. Available: https://blog.navigance.com/chemical-process-models (visited on 05/02/2022).

[4] L. Johansson and Moe, 'Temperature control and power monitoringof liquid cooling systems for mechanicalmanufacturing,' eng, 2019.

[5] M. LLC. 'Dynamics and Control first order plus dead time (fopdt).' (2020), [Online]. Available: https://apmonitor.com/pdc/index.php/Main/FirstOrderSystems (visited on 25/01/2022).

[6] 'Curve fitting.' (2022), [Online]. Available: https://en.wikipedia.org/wiki/Curve_fitting (visited on 25/01/2022).

[7] 'System identification.' (2022), [Online]. Available: https://en.wikipedia.org/wiki/System%5C_identification (visited on 25/04/2022).

[8] R. Sharma, 'Lecture notes for the course iia 4117: Model predictive control,' 2019.

[9] 'Understanding Model Predictive Control.' (2022), [Online]. Available: https://se.mathworks.com/videos/series/understanding-model-predictive-control.html (visited on 16/02/2022).

[10] 'NAROMtemperature sensor (ntc).' (2022), [Online]. Available: https://www.narom.no/undervisningsressurser/the-cansat-book/the-primary-mission/sensors-for-the-primary-mission/temperature-sensor-b-ntc/ (visited on 20/03/2022).

[11] *Stp100*, D-12-41, Schneider Electric, Aug. 2004.

[12] 'Grundfosmagna3.' (2022), [Online]. Available: https://product-selection.grundfos.com/no/products/magna/magna3?tab=explore (visited on 20/03/2022).

[13] *Multical 603*, DK-BEK 1178, Kamstrup, Nov. 2014.

[14] 'Wikipediaspeed of sound.' (2022), [Online]. Available: https://en.wikipedia.org/wiki/Speed%5C_of%5C_sound%5C#Dependence%5C_on%5C_the%5C_properties%5C_of%5C_the%5C_medium (visited on 20/03/2022).

[15] 'Peak Sensorswhat is a pt100 sensor?' (2022), [Online]. Available: https://peaksensors.co.uk/what-is/pt100-sensor/ (visited on 20/03/2022).

[16] 'SciJinkshow reliable are weather forecasts?' (), [Online]. Available: https://scijinks.gov/forecast-reliability/ (visited on 23/04/2022).

[17] H. Ferreau, C. Kirches, A. Potschka, H. Bock and M. Diehl, 'qpOASES: A parametric active-set algorithm for quadratic programming,' *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.

[18] 'Wikipediasnow.' (2022), [Online]. Available: https://en.wikipedia.org/wiki/Snow (visited on 25/04/2022).

[19] 'MathWorksplicing and licensing.' (2022), [Online]. Available: https://se.mathworks.com/pricing-licensing.html?prodcode=ML%5C&intendeduse=comm (visited on 28/04/2022).

[20] M. Akselsen, Private Communication, Oslo, Norway, 2022.

[21] 'Fortumpriser og vilkår.' (2022), [Online]. Available: https://www.fortum.no/bedrift/fjernvarme/priser-og-vilkar?vtab=accordion-item-5886 (visited on 26/04/2022).

[22] 'YR så mye snø kan du få i vinter.' (2013), [Online]. Available: https://www.yr.no/artikkel/sa-mye-sno-kan-det-komme-i-vinter-1.11312139 (visited on 28/04/2022).

[23] Q. Bi, W.-J. Cai, E.-L. Lee, Q.-G. Wang, C.-C. Hang and Y. Zhang, 'Robust identification of first-order plus dead-time model from step response,' eng, *Control engineering practice*, vol. 7, no. 1, pp. 71–77, 1999, ISSN: 0967-0661.

[24] M. S. Kim, D.-U. Jang, J.-S. Hong and T. Kim, 'Thermal modeling of railroad with installed snow melting system,' eng, *Cold regions science and technology*, vol. 109, pp. 18–27, 2015, ISSN: 0165-232X.

[25] T. WIGENSTAD, 'Prosjektveileder forenklet anlegg for vannbåren oppvarming av boliger,' nor, 2009.

Appendix A

Task Description

University of
South-Eastern Norway

Faculty of Technology, Natural Sciences and Maritime Sciences, Campus Porsgrunn

FMH606 Master's Thesis

Title: Analyzing heat exchangers for HVAC systems with respect to temperature, volume flow, and energy.

USN supervisor: Carlos F. Pfeiffer

External partner:

Task background:
All modern buildings for public and commercial use has a HVAC(Heating, ventilation, and air conditioning) system, these systems usually consist of one or more Water to water heat exchangers set up in different configurations and for different purposes. This can be to separate energy user from supplier, to have different operating points for temperature or to separate different mediums. The setup can often be dependent on the source of energy (central heating, heat pump, electrical boiler). For central heating it's usually important to have low flow and high delta temperature, while for heat pumps minimum run rime and few start/stops are desired. The setup can also vary dependent on the use, for example for room heating, air handling units or snow melting.

Specifically, a water-to-water heat exchanger used to control a snow melting system is of interest. The system uses heated water to heat a alcohol mixture that again is circulated in pavements to melt snow. The settings and desired temperatures for this system are dependent of the weather and weather forecasting. It is desirable to use the predicted weather to control the snow melting in an optimal way.
The melting of snow is an expensive operation, and it is desirable to minimise the cost(energy consumption).

Task description:

- Give an insight to different uses and setups of water-to-water heat exchangers in HVAC installations.
- Perform a literature study of which variables can be optimized in a snow melting system.
- Formulate a mathematical model of a heat exchanger with the means of simulating its behaviour. The model can be formulated using nonlinear differential equations, ordinary differential equations or First Order Plus Dead Time. The possibility to use system identification to obtain a mathematical model is to be investigated.
- The model(s) is to be fitted to data gathered from the chosen system. The data shall come from test runs where a step response is performed.
- Develop a Model Predictive controller (MPC) to control the heat exchanger. What variables to optimize shall reflect the finds during the literature study.
- Test the MPC on simulations and compare the results with simulations with traditional control methods (PI/PID). Use whether forecasting to predict changes in the system and implement the changes in the optimal control problem.
- If time allows it, and it is feasible withe the software used to control the system, the MPC is to be tested on the physical system

- Perform an economical evaluation of the possible savings.

Student category: IIA

Carlos f. Pfeiffer
Is the task suitable for online students (not present at the campus)? Yes

Practical arrangements:
zzz

Supervision:
As a general rule, the student is entitled to 15-20 hours of supervision. This includes necessary time for the supervisor to prepare for supervision meetings (reading material to be discussed, etc).

Signatures:

Supervisor (date and signature): 28.01.2022 Carlos f. Pfeiffer

Student (write clearly in all capitalized letters):

Student (date and signature): 28.01.22 Tim Cato Lybekk

Appendix B

Formulation of the Optimization Problem and transition to Standard Quadratic Programming Formulation

This document gives a detailed description of the Optimization Problem for use in the MPC for a snow melting system. The Optimization Problem is transformed into Standard Quadratic Programming Formulation to be feasible for use with the desired solver.

QP formulation of optimizing problem for control of snow melting system

Tim Cato Lybekk

May 16, 2022

0.0.1 Introduction

This document describes in greater detail the formulation of a optimization problem for a snow melting system. A objective function is formulated based on the desired optimal solution. Appropriate constraints are defined, based on a state space model and logical limitations of the system. The Optimization problem is formulated on QP(Quadratic Programming) form and required matrix structures are established. This document is considered to be an appendix to the thesis "An approach to optimal control of snow melting systems" by Tim Cato Lybekk, and needs to be seen in this context to understand the choices and assumptions made.

0.0.2 Objective function

Firstly the objective is defined. For this optimal control problem the objective is to minimize the error e_k, which is defined to be difference between the process output y_k and the reference values r_k(setpoint):

$$y_k = T2 \tag{1}$$

$$r_k = T2_{ref} \tag{2}$$

$$e_k = r_k - y_k = T2_{ref} - T2 \tag{3}$$

Further, the prediction horizon is defined as N. Using the discrete time linear model

$$x_{k+1} = Ax_k + Bu_k$$

$$y_k = Cx_k$$

The objective function is formulated as:

$$\overset{min}{\text{(u)}}\, J = \frac{1}{2}\sum_{k=1}^{N}(e_k^T Q_k e_k + u_{k-1}^T P_{k-1} u_{k-1}) \tag{4}$$

s.t.

$$\begin{aligned} e_k &= T2_k - y_{T1} \\ x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \\ u_L \leq\, &u_k \leq u_U \\ x_L \leq\, &x_k \leq x_U \end{aligned} \tag{5}$$

where Q_k is the weighting matrix for the error and P_k is the weighting matrix for the control signal. u_L lower limit for the inputs and u_U upper limit for the inputs. x_L lower limit for the states and x_U upper limit for the states.

0.0.3 Prediction horizon

Expanding the objective function from $k = 1$ to $k = N$, to eliminate the summation:

$$J = \frac{1}{2}[e_1^T P_1 e_1 + e_2^T P_2 e_2 + \ldots e_N^T P_N e_N + u_0^T Q_0 u_0 + u_1^T Q_1 u_1 + \ldots + u_{N-1}^T Q_{N-1} u_{N-1}] \tag{6}$$

0.0.4 Standard QP formulation

For using the *qpOASES* solver to solve the optimal control problem, the problem needs to be expressed as a standard quadratic programming problem:

$$\overset{min}{\text{(z)}}\ J = \frac{1}{2} z^T H z + c^T z \tag{7}$$

s.t

$$\begin{aligned} A_e z &= b_e \\ A_i z &\leq b_i \\ z_L \leq z &\leq z_U \end{aligned} \tag{8}$$

Where the z vector represents the unknowns to optimized, defined as:

$$z = \begin{bmatrix} u \\ x \\ e \\ y \end{bmatrix} \tag{9}$$

where

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \quad e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{10}$$

writing the standard quadratic programming problem formulation on matrix form:

$$\overset{min}{\text{(z)}}\ J = \frac{1}{2} \begin{bmatrix} u \\ x \\ e \\ y \end{bmatrix}^T \begin{bmatrix} H_{11} & 0 & 0 & 0 \\ 0 & H_{22} & 0 & 0 \\ 0 & 0 & H_{33} & 0 \\ 0 & 0 & 0 & H_{44} \end{bmatrix} \begin{bmatrix} u \\ x \\ e \\ y \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}^T \begin{bmatrix} u \\ x \\ e \\ y \end{bmatrix} \tag{11}$$

Multiplying the matrices gives

$$\overset{min}{\text{(z)}}\ J = \frac{1}{2}[u^T H_{11} u + x^T H_{22} x + e^T H_{33} e + y^T H_{44} y] + c_1^T u + c_2^T x + c_3^T e + c_4^T y \tag{12}$$

0.0.5 Number of unknowns

n_u defines the number of unknown control inputs, and is for the problem in question one. n_x defines the number of unknown states, and may vary based on the set dead-time of the system and step size. For further detail see original thesis. n_e is the number of unknown errors and the number of unknown outputs is defined as n_y.

$$n_u = 1 \tag{13}$$

$$n_x = \text{dependent on dead-time of process and step size} \tag{14}$$

$$n_e = n_y = 1 \tag{15}$$

the total number of unknowns is:

$$n_z = N \times (n_u + n_x + n_y + n_y) \tag{16}$$

0.0.6 Formulating Objective Function on Standard QP form

Starting with comparing 6 and 12 to define H_{11}

$$u^T H_{11} u = u_0^T P_0 u_0 + u_1^T P_1 u_1 + \ldots + u_{N-1}^T P_{N-1} u_{N-1} \tag{17}$$

on matrix form

$$u^T H_{11} u = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix}^T \begin{bmatrix} P_0 & 0 & \ldots & 0 \\ 0 & P_1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & P_{N-1} \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \tag{18}$$

$$H_{11} = \begin{bmatrix} P_0 & 0 & \ldots & 0 \\ 0 & P_1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & P_{N-1} \end{bmatrix} \tag{19}$$

Assuming that the weighting matrix P_k is equal for every step N, $P_0 = P_1 = \ldots = P_{N-1} = P$

$$H_{11} = \begin{bmatrix} P & 0 & \ldots & 0 \\ 0 & P & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & P \end{bmatrix} \tag{20}$$

Again comparing 6 and 12. As 6 does not contain any terms for x we get:

$$x^T H_{22} x = x_1^T 0 x_1 + x_2^T 0 x_2 + \ldots x_N^T 0 x_N \tag{21}$$

on matrix form

$$x^T H_{22} x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}^T \begin{bmatrix} 0 & 0 & \ldots & 0 \\ 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \tag{22}$$

this gives

$$H_{22} = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 \end{bmatrix} \tag{23}$$

Again comparing 6 and 12 to define H_{33}

$$e^T H_{33} e = e_1^T Q_1 e_1 + e_2^T Q_2 e_2 + \ldots e_N^T Q_N e_N \tag{24}$$

on matrix form

$$e^T H_{33} u = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}^T \begin{bmatrix} Q_1 & 0 & \ldots & 0 \\ 0 & Q_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & Q_N \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \tag{25}$$

$$H_{33} = \begin{bmatrix} Q_1 & 0 & \dots & 0 \\ 0 & Q_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q_N \end{bmatrix} \tag{26}$$

Assuming that the weighting matrix Q_k is equal for every step N, $Q_1 = Q_2 = \dots = Q_N = Q$

$$H_{33} = \begin{bmatrix} Q & 0 & \dots & 0 \\ 0 & Q & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q \end{bmatrix} \tag{27}$$

Again comparing 6 and 12. As 6 does not contain any terms for y we get:

$$y^T H_{44} y = y_1^T 0 y_1 + y_2^T 0 y_2 + \dots y_N^T 0 y_N \tag{28}$$

on matrix form

$$y^T H_{44} y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}^T \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{29}$$

this gives

$$H_{44} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{30}$$

Finally the H matrix can be formulated as:

$$H = \begin{bmatrix} H_{11} & 0 & 0 & 0 \\ 0 & H_{22} & 0 & 0 \\ 0 & 0 & H_{33} & 0 \\ 0 & 0 & 0 & H_{44} \end{bmatrix} \tag{31}$$

Now for the linear terms, it can be seen from 6 that there is no linear terms. Comparing 6 and 12 the following can be formulated:

$$c_1^T u = 0u_0 + 0u_1 + \dots + 0u_{N-1} \tag{32}$$

$$c_2^T x = 0x_1 + 0x_2 + \dots + 0x_N \tag{33}$$

$$c_3^T e = 0e_1 + 0e_2 + \dots + 0e_N \tag{34}$$

$$c_4^T y = 0y_1 + 0y_2 + \dots + 0y_N \tag{35}$$

on matrix form

$$c_1^T u = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \tag{36}$$

$$c_2^T x = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \tag{37}$$

$$c_3^T e = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \tag{38}$$

$$c_4^T y = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ u_N \end{bmatrix} \tag{39}$$

giving

$$c_1^T u = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \tag{40}$$

$$c_2^T x = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \tag{41}$$

$$c_3^T e = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \tag{42}$$

$$c_4^T y = \begin{bmatrix} 0 & 0 & \dots & 0 \end{bmatrix} \tag{43}$$

finally the linear terms can be expressed as:

$$c^T = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} 0_{N.n_u} \\ 0_{N.n_x} \\ 0_{N.n_e} \\ 0_{N.n_y} \end{bmatrix} \quad \text{Where } n_z \text{ is number of total unknowns} \tag{44}$$

Now using the equality constraints, defined for the LQ optimal problem seen in eq. 5, and formulating them on standard QP form:

$$A_\varepsilon z = b_\varepsilon \tag{45}$$

the objective function contains three equality constraints, therefore A_ε and b_ε will consist of three rows:

$$\begin{bmatrix} A_{\varepsilon,1u} & A_{\varepsilon,1x} & A_{\varepsilon,1e} & A_{\varepsilon,1y} \\ A_{\varepsilon,2u} & A_{\varepsilon,2x} & A_{\varepsilon,2e} & A_{\varepsilon,2y} \\ A_{\varepsilon,3u} & A_{\varepsilon,3x} & A_{\varepsilon,3e} & A_{\varepsilon,3y} \end{bmatrix} \begin{bmatrix} u \\ x \\ e \\ y \end{bmatrix} = \begin{bmatrix} b_{\varepsilon,1} \\ b_{\varepsilon,2} \\ b_{\varepsilon,3} \end{bmatrix} \tag{46}$$

starting with the equality constraint:

$$x_{k+1} = Ax_k + Bu_k \tag{47}$$

rearranging to be equal to zero

$$x_{k+1} - Ax_k - Bu_k = 0 \tag{48}$$

$$x_k - Ax_{k-1} - Bu_{k-1} = 0 \tag{49}$$

Then defining the constraint throughout the prediction horizon. x_0 is known and is therefore moved to the right hand side for the first term.

$$\begin{array}{cc} x_1 - Bu_0 = Ax_0 & \text{for } k = 1 \\ x_2 - Ax_1 - Bu_1 = 0 & \text{for } k = 2 \\ \vdots & \vdots \\ x_N - Ax_{N-1} - Bu_{N-1} = 0 & \text{for } k = N \end{array} \tag{50}$$

on matrix form

$$\left[\underbrace{\begin{matrix} -B & 0 & 0 & \dots & 0 \\ 0 & -B & 0 & \dots & 0 \\ 0 & 0 & -B & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & -B \end{matrix}}_{A_{\varepsilon,1u}} \vdots \underbrace{\begin{matrix} I & 0 & 0 & \dots & 0 \\ -A & I & 0 & \dots & 0 \\ 0 & -A & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -A & I \end{matrix}}_{A_{\varepsilon,1x}} \vdots \underbrace{\begin{matrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{matrix}}_{A_{\varepsilon,1e}} \vdots \underbrace{\begin{matrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{matrix}}_{A_{\varepsilon,1y}}\right]_{(N.n_x \times n_z)} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \\ x_1 \\ x_2 \\ \vdots \\ x_N \\ e_1 \\ e_2 \\ \vdots \\ e_N \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}_{(n_z \times 1)} = \underbrace{\begin{bmatrix} Ax_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{B_{\varepsilon,1}} \tag{51}$$

$$A_{\varepsilon,1u} = -I_N \otimes B \tag{52}$$

$$A_{\varepsilon,1x} = I_{N \cdot n_x} - (I_{N-1} \otimes A) \tag{53}$$

$$A_{\varepsilon,1e} = 0_{(N \cdot n_x \times N \cdot n_y)} \tag{54}$$

$$A_{\varepsilon,1y} = 0_{(N \cdot n_x \times N \cdot n_y)} \tag{55}$$

$$B_{\varepsilon,1} = \begin{bmatrix} Ax_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{56}$$

Then for the equality constraint:

$$y_k = Cx_k \tag{57}$$

rearranging to be equal to zero

$$y_k - Cx_k = 0 \tag{58}$$

Then defining the constraint throughout the prediction horizon.

$$\begin{array}{cc} y_1 - Cx_1 = 0 & \text{for } k = 1 \\ y_2 - Cx_2 = 0 & \text{for } k = 2 \\ \vdots & \vdots \\ y_N - Cx_N = 0 & \text{for } k = N \end{array} \tag{59}$$

on matrix form

$$\begin{bmatrix} \underbrace{\begin{matrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{matrix}}_{A_{\varepsilon,2u}} & \vdots & \underbrace{\begin{matrix} -C & 0 & 0 & \dots & 0 \\ 0 & -C & 0 & \dots & 0 \\ 0 & 0 & -C & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & -C \end{matrix}}_{A_{\varepsilon,2x}} & \vdots & \underbrace{\begin{matrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{matrix}}_{A_{\varepsilon,2e}} & \vdots & \underbrace{\begin{matrix} I & 0 & 0 & \dots & 0 \\ 0 & I & 0 & \dots & 0 \\ 0 & 0 & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & I \end{matrix}}_{A_{\varepsilon,2y}} \end{bmatrix}_{(N.n_x \times n_z)} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \\ x_1 \\ x_2 \\ \vdots \\ x_N \\ e_1 \\ e_2 \\ \vdots \\ e_N \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}_{(n_z \times 1)} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{B_{\varepsilon,2}} \tag{60}$$

$$A_{\varepsilon,2u} = 0_{(N \cdot n_y \times N \cdot n_u)} \tag{61}$$

$$A_{\varepsilon,2x} = -I_N \otimes C \tag{62}$$

$$A_{\varepsilon,2e} = 0_{(N \cdot n_y \times N \cdot n_y)} \tag{63}$$

$$A_{\varepsilon,2y} = I_{N \cdot n_y} \tag{64}$$

$$B_{\varepsilon,2} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{65}$$

Finally considering the equality constraint:

$$e_k = r_k - y_k \tag{66}$$

As the reference value r_k is known for the entire prediction horizon, the equation is not rearranged.

$$e_k + y_k = r_k \tag{67}$$

Then defining the constraint throughout the prediction horizon.

$$\begin{aligned} e_1 + y_1 &= r_1 \qquad \text{for } k = 1 \\ e_2 + y_2 &= r_2 \qquad \text{for } k = 2 \\ &\vdots \qquad\qquad \vdots \\ e_N + y_N &= r_N \qquad \text{for } k = N \end{aligned} \tag{68}$$

Arranging to matrix form

$$\begin{bmatrix} \underbrace{\begin{matrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{matrix}}_{A_{\varepsilon,3u}} & \vdots & \underbrace{\begin{matrix} 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{matrix}}_{A_{\varepsilon,3x}} & \vdots & \underbrace{\begin{matrix} I & 0 & 0 & \dots & 0 \\ 0 & I & 0 & \dots & 0 \\ 0 & 0 & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & I \end{matrix}}_{A_{\varepsilon,3e}} & \vdots & \underbrace{\begin{matrix} I & 0 & 0 & \dots & 0 \\ 0 & I & 0 & \dots & 0 \\ 0 & 0 & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & I \end{matrix}}_{A_{\varepsilon,3y}} \end{bmatrix}_{(N.n_x \times n_z)} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \\ x_1 \\ x_2 \\ \vdots \\ x_N \\ e_1 \\ e_2 \\ \vdots \\ e_N \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}_{(n_z \times 1)} = \underbrace{\begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix}}_{B_{\varepsilon,3}} \tag{69}$$

$$A_{\varepsilon,3u} = 0_{(N \cdot n_y \times N \cdot n_u)} \tag{70}$$

$$A_{\varepsilon,3x} = 0_{(N \cdot n_y \times N \cdot n_x)} \tag{71}$$

$$A_{\varepsilon,3e} = I_{N \cdot n_y} \tag{72}$$

$$A_{\varepsilon,3y} = I_{N \cdot n_y} \tag{73}$$

$$B_{\varepsilon,3} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \tag{74}$$

Finally the matrices A_ε and b_ε can be expressed as:

$$A_\varepsilon = \begin{bmatrix} -I_N \otimes B & I_{N \cdot n_x} - (I_{N-1} \otimes A) & 0_{(N \cdot n_x \times N \cdot n_y)} & 0_{(N \cdot n_x \times N \cdot n_y)} \\ 0_{(N \cdot n_y \times N \cdot n_u)} & -I_N \otimes C & 0_{(N \cdot n_y \times N \cdot n_y)} & I_{N \cdot n_y} \\ 0_{(N \cdot n_y \times N \cdot n_u)} & 0_{(N \cdot n_y \times N \cdot n_x)} & I_{N \cdot n_y} & I_{N \cdot n_y} \end{bmatrix} \tag{75}$$

$$B_\varepsilon = \begin{bmatrix} Ax_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \tag{76}$$

The bounds on the control inputs and the states, as seen in eq. 5 is formulated as follows:

$$x_L \leq x_k \leq x_U$$

$$\begin{bmatrix} -\infty \\ -\infty \\ \vdots \\ -\infty \\ RT902_{k_L} \\ -\infty \\ \vdots \\ -\infty \end{bmatrix} \leq \begin{bmatrix} T2 \\ T2_{\theta_1} \\ \vdots \\ T2_{\theta_q} \\ RT902 \\ RT902_{\theta_1} \\ \vdots \\ RT902_{\theta_p} \end{bmatrix} \leq \begin{bmatrix} \infty \\ \infty \\ \vdots \\ \infty \\ \infty \\ \infty \\ \vdots \\ \infty \end{bmatrix} \tag{77}$$

Where $RT902_{L_k}$ is the lower limit for the state $RT902_k$.

The outputs are to be limited to be between 0% and 100% , denoted as

$$u_L \leq u_k \leq u_U$$

$$0\% \leq SB401 \leq 100\% \tag{78}$$

As there is no constraints on the outputs and errors, the upper and lower limits for all outputs and errors are defined as ∞ and $-\infty$ for every step.

the constraints are stacked in the vectors z_L for the lower limits and z_U for the upper limits:

$$z_L = \begin{pmatrix} 1_{N\times 1} \otimes u_L \\ 1_{N\times 1} \otimes x_L \\ -\infty_{(n_e)\times 1} \\ -\infty_{(n_y)\times 1} \end{pmatrix} \tag{79}$$

$$z_U = \begin{pmatrix} 1_{N\times 1} \otimes u_U \\ 1_{N\times 1} \otimes x_U \\ \infty_{(n_e)\times 1} \\ \infty_{(n_y)\times 1} \end{pmatrix} \tag{80}$$

where

$$u_L = -50\% \qquad \text{and} \qquad u_U = 50\% \tag{81}$$

$$\begin{bmatrix} -\infty \\ -\infty \\ \vdots \\ -\infty \\ RT902_{k_L} \\ -\infty \\ \vdots \\ -\infty \end{bmatrix} \qquad \text{and} \qquad \begin{bmatrix} \infty \\ \infty \\ \vdots \\ \infty \\ \infty \\ \infty \\ \vdots \\ \infty \end{bmatrix} \tag{82}$$

0.0.7 Resulting QP problem formulation

$$z = \begin{bmatrix} u \\ x \\ e \\ y \end{bmatrix} \tag{83}$$

$$u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \quad e = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{84}$$

$$H_{11} = \begin{bmatrix} P & 0 & \dots & 0 \\ 0 & P & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & P \end{bmatrix} \tag{85}$$

$$H_{22} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{86}$$

$$H_{33} = \begin{bmatrix} Q & 0 & \dots & 0 \\ 0 & Q & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Q \end{bmatrix} \tag{87}$$

$$H_{44} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{88}$$

$$H = \begin{bmatrix} H_{11} & 0 & 0 & 0 \\ 0 & H_{22} & 0 & 0 \\ 0 & 0 & H_{33} & 0 \\ 0 & 0 & 0 & H_{44} \end{bmatrix} \tag{89}$$

$$c^T = \begin{bmatrix} 0_{N.n_u} \\ 0_{N.n_x} \\ 0_{N.n_e} \\ 0_{N.n_y} \end{bmatrix} \tag{90}$$

$$A_\varepsilon = \begin{bmatrix} -I_N \otimes B & I_{N\cdot n_x} - (I_{N-1} \otimes A) & 0_{(N\cdot n_x \times N\cdot n_y)} & 0_{(N\cdot n_x \times N\cdot n_y)} \\ 0_{(N\cdot n_y \times N\cdot n_u)} & -I_N \otimes C & 0_{(N\cdot n_y \times N\cdot n_y)} & I_{N\cdot n_y} \\ 0_{(N\cdot n_y \times N\cdot n_u)} & 0_{(N\cdot n_y \times N\cdot n_x)} & I_{N\cdot n_y} & I_{N\cdot n_y} \end{bmatrix} \tag{91}$$

$$B_\varepsilon = \begin{bmatrix} Ax_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \tag{92}$$

$$x_L = \begin{bmatrix} -\infty \\ -\infty \\ \vdots \\ -\infty \\ RT902_{k_L} \\ -\infty \\ \vdots \\ -\infty \end{bmatrix} \tag{93}$$

$$x_U = \begin{bmatrix} \infty \\ \infty \\ \vdots \\ \infty \\ \infty \\ \infty \\ \vdots \\ \infty \end{bmatrix} \tag{94}$$

$$u_L = 0\% \tag{95}$$

$$u_U = 100\% \tag{96}$$

$$z_L = \begin{pmatrix} 1_{N\times 1} \otimes u_L \\ 1_{N\times 1} \otimes x_L \\ -\infty_{(n_e)\times 1} \\ -\infty_{(n_y)\times 1} \end{pmatrix} \tag{97}$$

$$z_U = \begin{pmatrix} 1_{N\times 1} \otimes u_U \\ 1_{N\times 1} \otimes x_U \\ \infty_{(n_e)\times 1} \\ \infty_{(n_y)\times 1} \end{pmatrix} \tag{98}$$

Appendix C

Initiation Script

In this appendix the Matlab code for the script for initialization of the MPC is presented.

Listing C.1: Code for initialising the MPC

```
clc
clear


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% First Order Plus Dead Time model of snow melting system
%
%
%
%
%
% by Tim Cato Lybekk
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Index of variable
% T2 = 1;
% RT902 = 2;
TimeHorizon = 48; % 24h
delta = 0.5; % Timestep = hour
N=TimeHorizon/delta % N horizon

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Model parameters
%relationship between SB401 and T2
K_p_T2 = -0.7;                      %Gain
t_c_T2 = 0.5;                     %Time Constant h
t_d_T2 = 1;                       %Deadtime h
t_d_T2_n = round(t_d_T2/delta)+1        %Number of steps...
                                           %representing deadtime
%relationship between SB401 and RT902
K_p_RT902 = 1.5;                    %Gain
t_c_RT902 = 4;                    %Time Constant i*10min
t_d_RT902 = 4;                    %Deadtime i*10min
t_d_RT902_n = t_d_RT902/delta+1  %Number of steps...
                                    %representing deadtime
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Defining positions for variables in state space matrices
T2pos = 1;
RT902pos = t_d_T2_n+1;
n_tot = t_d_T2_n+t_d_RT902_n;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Creating A matrix
%Initializing epty A_T2 matrix
```

```
A_T2 = diag(ones(t_d_T2_n-abs(1),1),1) + eye(t_d_T2_n)*-1;
%Initializing A matrix for T2
A_T2_temp = [-1/t_c_T2, K_p_T2/t_c_T2; 0, -1];
%Initializing epty A_RT902 matrix
A_RT902 = diag(ones(t_d_RT902_n-abs(1),1),1) + eye(t_d_RT902_n)*-1;
%Initializing A matrix for RT902
A_RT902_temp = [-1/t_c_RT902, K_p_RT902/t_c_RT902; 0, -1];
A_T2(1:2,1:2) = A_T2_temp;
A_RT902(1:2,1:2) = A_RT902_temp;
Ac = zeros(n_tot);
Ac(1:t_d_T2_n,1:t_d_T2_n) = A_T2;
Ac(RT902pos:RT902pos+t_d_RT902_n-1,RT902pos:RT902pos+...
    t_d_RT902_n-1) = A_RT902;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Creating B matrix
Bc = [zeros(1,t_d_T2_n-1)';1;...
    zeros(1,t_d_RT902_n-1)';1;...
    ];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Creating C matrix
Cc = zeros(1,n_tot);
%C(1,T2pos) = 1;          % Dfining T2 as obserable
Cc(1,T2pos) = 1;   % Dfining RT902 as obserable
%Cc(1,RT902pos) = 1;   % Dfining RT902 as obserable
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%change to discrete time model
sys = ss(Ac,Bc,Cc,0); %there is no D matrix, so set it as 0
ds = c2d(sys,delta);
A = ds.a; B = ds.b; C = ds.c; D = ds.d;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%standard quadratic programming problem

%size of matrices
An = size(A);
nx = An(1)
ny = 1;                 %Numvber of controlled states?
nu = 1;
%size of the unknow vector z
nz = N*(nx + nu + 2*ny)
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Initial states
x0 = zeros(n_tot,1);
x0(T2pos,1) = 0;
x0(RT902pos,1) = 0;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Refrence value
%The refrence is the inlet temperature on the primary side and will...
%be 55°C for the entire prediction horizion
r = [zeros(1,N/2),ones(1,N/2)*0];
%r = zeros(1,N);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%weighting matrices
Q=100.0; %tuning weight for error: there is 1
P=1e-3; %tuning weight for inputs: there is 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% build matrices
H11 = kron(eye(N),P);
H22 = zeros(N*nx,N*nx);
H33 = kron(eye(N),Q);
H44 = zeros(N*ny,N*ny);
H_mat = blkdiag(H11,H22,H33,H44);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%qpOASES does not accept matrices, but only vectors
%we have to change H matrix to vector by stacking elements column wise
H = H_mat(:);
c = zeros(nz,1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%constraints (from process model)
% from eq 3.33: state equation
Ae1u = -kron(eye(N),B);
Ae1x = eye(N*nx)-kron(diag(ones(N-abs(-1),1),-1),A);
Ae1e = zeros(N*nx,N*ny);
Ae1y = zeros(N*nx,N*ny);
be1 = [A*x0;zeros((N-1)*nx,1)];
%from eq 3.34: measurement equation
Ae2u = zeros(N*ny,N*nu);
Ae2x = -kron(eye(N),C);
Ae2e = zeros(N*ny,N*ny);
Ae2y = eye(N*ny);
be2 = zeros(N*ny,1);
%from eq 3.35: error equation
Ae3u = zeros(N*ny,N*nu);
Ae3x = zeros(N*ny,N*nx);
Ae3e = eye(N*ny);
```

```
Ae3y = eye(N*ny);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% since be3 contains the reference vector in specific order,...
% we have to use the function “”reshape to put the reference...
% values in the right order
be3 = reshape(r,N*ny,1);
Ae_mat=[Ae1u Ae1x Ae1e Ae1y;...
     Ae2u Ae2x Ae2e Ae2y;...
     Ae3u Ae3x Ae3e Ae3y];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%qpOASES does not accept matrices, but only vectors
%we have to change Ae matrix to vector by stacking...
%elements column wise
Ae = Ae_mat(:); %stacking column wise
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% make the standard be vector
be=[be1;be2;be3];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%bounds

%RT902L = [-Inf*ones(N/2,1); 1*ones(N/4,1); -Inf*ones(N/4,1)];
%RT902L = zeros(N,1);
RT902L = [ones(1,24)*-inf ones(1,12)*8 ones(1,60)*-inf];
uL = ones(N,1)*0;
uH = ones(N,1)*100;
xL_mat = [ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     RT902L ;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf;...
     ones(1,N)*-inf];
xL = xL_mat(:);
%xL = [ones(N,1)*-1; ones(N,1)*-1; ones(N,1)*-1; ones(N,1)* -1;...
                      % ones(N,1)*-1; ones(N*(nx-5),1)*-2];
%xL = [(-Inf*ones(N*(t_d_T2_n),1)); RT902L;...
                      %(-Inf*ones(N*(t_d_RT902_n-1),1)) ];
%xL = (-Inf*ones(N*nx,1));
xH = (Inf*ones(N*(t_d_T2_n+t_d_RT902_n),1));
eL = (-Inf*ones(N*ny,1));
```

```
eH = (Inf*ones(N*ny,1));
yL = (-Inf*ones(N*ny,1));
yH = (Inf*ones(N*ny,1));



zL=[uL; xL; eL; yL];
zH=[uH; xH; eH; yH];
size(zL)







%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% State space model representation
%Initialising variable for storing outputs
y_ss = zeros(n_tot,N); %y = [T2 ; ]
%Initializing variable for derivative of states
x_dot = zeros(N,n_tot)'; % x_dot = [T1_dot;Z_1_dot; Z_2_dot; ...
                %; Z_(t_d_T2_n)_dot ]
%Initializing variable for states
x = zeros(n_tot,N);
%defining the system input u
u = ones(1, N)*0;
time = linspace(0, N*delta, N);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Running the state space model
sp = [ones(1, 12)*0,ones(1, 12)*6,ones(1, 12)*8,...
    ones(1, N-12-12-12)*0] ;
int_x = 0;
Kp_PID = 0.5;
Ti_PID =7;
for i =  1:N-1
    x_dot = Ac*x(:,i) + Bc.*u(i);
    y_ss(:,i) = Cc*x(:,i);
    x(:,i+1) = x(:,i) + x_dot*delta;
    e=sp(i)-x(RT902pos,i);
    int_x = int_x + (Kp_PID/Ti_PID)*e;
    if int_x > 100
        int_x = 100;
    end
    if int_x < 0
```

```
            int_x = 0;
        end
        if i < N
            u(i+1) = (Kp_PID*e+int_x);
        end
        if u(i+1) > 100
            u(i+1) = 100;
        end
        if u(i+1) < 0
            u(i+1) = 0;
        end

end

u_PID = u;
RT902_PID = x(RT902pos,:);
T2_PID = x(T2pos,:);
RT902_PID_SP = sp;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Plotting the result
plot(time, u(1,:),...
    time , x(T2pos,:), '--',...
    time , x(RT902pos,:), '--',...
    time , sp(:), '--'...
    )
legend({'SB401','T2','RT902','SP'},'Location','southwest')
```